We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter 7

Evolving Networks and Social Network Analysis Methods and Techniques

Mário Cordeiro, Rui P. Sarmento, Pavel Brazdil and João Gama

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.79041

Abstract

Evolving networks by definition are networks that change as a function of time. They are a natural extension of network science since almost all real-world networks evolve over time, either by adding or by removing nodes or links over time: elementary actor-level network measures like network centrality change as a function of time, popularity and influence of individuals grow or fade depending on processes, and events occur in networks during time intervals. Other problems such as network-level statistics computation, link prediction, community detection, and visualization gain additional research importance when applied to dynamic online social networks (OSNs). Due to their temporal dimension, rapid growth of users, velocity of changes in networks, and amount of data that these OSNs generate, effective and efficient methods and techniques for small static networks are now required to scale and deal with the temporal dimension in case of streaming settings. This chapter reviews the state of the art in selected aspects of evolving social networks presenting open research challenges related to OSNs. The challenges suggest that significant further research is required in evolving social networks, i.e., existent methods, techniques, and algorithms must be rethought and designed toward incremental and dynamic versions that allow the efficient analysis of evolving networks.

Keywords: evolving networks, social network analysis

1. Introduction

One of the consequences of today's information society is the rise of the digital network society [1]. Organizations and individuals are increasingly connected through a wide range of online

IntechOpen

© 2018 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. CC BY

and offline networks at different relational levels: social, professional, interaction, information flow, etc. The analysis and modeling of networks and also networked dynamical systems have been the subject of considerable interdisciplinary interest covering a wide range of areas from physics, mathematics, computer science, biology, economics and sociology, in the so-called "new" science of networks [2]. According to [3], media organizations, media content and audiences are no exception: news articles have hyperlinks to link other content; news organizations disseminate news via online social network (OSN) platforms like Twitter and Facebook; and users comment, share and react directly below online news. At the level of news events, recent observation proves that some events and news emerge and spread first using those media channels rather than other traditional media like the online news sites, blogs or even television and radio breaking news [4, 5]. Natural disasters, celebrity news, products announcements or mainstream event coverage show that people increasingly make use of those tools to be informed, discuss and exchange information [6]. Concerning the novelty and timely dissemination of news events, empirical studies show that the online social networking services like Twitter are often the first mediums to break critical natural events such as earthquakes often in a matter of seconds after they occur [4, 5]. Herewith, social networks' temporal dimension of information is of crucial importance and follows a time decay pattern, that is, posted messages in social media are exchanged, forwarded or commented in early stages and decrease in importance as time passes. The importance of information contained in those messages has their importance peak right after being posted or in the following hours or days [7]. Besides, the timing of many human activities, ranging from communication to entertainment and work patterns is characterized by bursts of rapidly occurring events separated by long periods of inactivity, following non-Poisson statistics [8]. The nature of the time decay pattern communication, bursts or peaks, and inactivity periods, enforces the importance of dynamic network analysis methods and techniques in a network analysis context as the best approaches to model these problems.

1.1. Solving problems with evolving networks

Typical tasks of social network analysis involve the identification of the most influential, prestigious or central actors, using statistical measures; the identification of hubs and authorities, using link analysis algorithms; the discovery of communities, using community detection techniques; the visualization the interactions between actors; or spreading of information. These tasks are instrumental in the process of extracting knowledge from networks and consequently in the process of problem-solving with network data. Specifically, in the areas of journalism and investigation the following topics have been discussed actively in the last years: ***Criminological research.*** In 2001, according to [9], social network analysis has the genuine potential to uncover the complexities of criminal networks. Ref. [10] introduced the intersection of terrorism studies and what was called the "networked criminology". In 2011, [9] concluded that the applications of social network analysis in criminology were still insufficient when compared to other research areas like sociology and public health [11]. Nevertheless, since then, social network analysis has become one of the major tools for criminal analysis with [12] identifying the three main areas of analysis and applications of social networks in criminological research. Other topics include the *influence of personal networks* on crime and,

more generally, on delinquent behavior [13, 11]; the analysis of *neighborhood networks* and their influence on crime [12]; and exploration and *modeling of the organization* of crime, that is, street gangs, terrorist groups and organized crime groups [12]. Concrete examples of criminological research using social network analysis are the Enron company dataset [14]; the Panama Papers analysis in connection with the economic network analysis of Portuguese companies; individuals with connections to offshore companies [15]; and the analysis of terrorist networks [16, 17]. Gill and Malamud [18] presented a broad overview, characterization and visualization of the interaction relationships between natural hazards using social network analysis. In addition to the previous contributions, essential and recent work regarding social network analysis and terrorism were published by Malm et al. [19], while Berlusconi [20] devoted an entire book to social network analysis and crime prevention. ***Research on journalism:*** Fu [21] presented an essay with the purpose of promoting social network analysis in the study of journalism. Starting with a communication network taxonomy [22, 23], their focus was on network relations in the study of journalism. The proposed framework presents the four types of communication relations that characterize different networked journalism phenomena [23]: *Affinity relations* describes the socially constructed relationships between two actors, such as alliances and friendships, with the valence of the relation being either positive or negative; *flow relations* refers to the exchange and transmission of data, resources and information; *representational relations* focuses on the symbolic affiliation between two entities; and *semantic networks* describes the associations, or semantic relations, among concepts, words or people's cognitive interpretations toward some shared objects in the network, aiming to build a knowledge base. Another relevant journalism research example is Ref. [24] that examines the temporal dynamics of reciprocity in the setting of legislative co-sponsorship in the 113th US Congress (2013–2015).

1.2. Related work

In the past years, several overviews of social network analysis can be found in the literature. Ref. [25] provide a general and succinct overview of the essentials of social network analysis, for static networks, with emphasis on simple statistical measures, link analysis, properties of real-world networks and community detection. Tabassum et al. [26] in an updated overview of social network analysis, based on Oliveira's and Gama's [25] work, included a full section devoted to evolving networks. Devoted explicitly to evolutionary network analysis, the Aggarwal and Subbian [27] survey provides an overview of the vast literature on graph evolution analysis. Although the scope of Aggarwal and Subbian [27] work was graph analysis in general, that is, it did not address specifically the social network analysis problem, many of the mentioned applications and particular contexts of applicability of the methods are social networks. The literature analyzed by Aggarwal and Subbian [27] covered both snapshot-based and streaming methods and algorithms, and critical applications of evolutionary network analysis on different domains such as the world wide web, telecommunication and communication networks, recommendation networks and social network events, among others, were given. Spiliopoulou [28] said that the advances on evolution is social networks into a common framework to model a network across the *time axis* and identified the four dimensions associated with knowledge discovery in social networks: dealing with time, objective of study,

definition of community, and evolution as an objective versus assumption. Spiliopoulou [28] enumerated the several challenges of the social network streams although this problem is generally conceived as a stream problem. An exciting application of temporal network theory and temporal networks to functional brain connectivity was presented by Thompson et al. [29]; in his work, the theory and methods that introduce the reader on how to add the temporal dimension to network analysis and precisely how many of well-known methods can be transposed from static networks to temporal networks were presented. Thompson et al. [29] included a complete list of network measures adapted or proposed for temporal networks.

2. Representation of social networks

A social network is a social structure consisting of a finite set of social actors, such as individuals or organizations, connected by interpersonal relationships. These relationships, also known as ties, can be of personal or professional nature and can range from casual friends, acquaintances or co-workers to the close family bonds. Besides the relations, social networks often represent flow of information, interactions and similarities, among the set of social actors. Social network analysis is the investigation of the relationship between actors. In network terminology, vertices—also known as nodes—refer to actors or subjects. Edges, also known as links or ties, describe the relationship between actors. Usually, this social structure, or network structure, is represented by graphs which are mathematical structures used to model pair-wise relations between objects. A graph in this context is made up of vertices, nodes or points which are connected by edges, arcs or lines. Therefore, a social network is a graph G composed of two fundamental components: a nonempty set of vertices V and a set of edges E. Formally it can be defined as $G = (V, E)$. Vertices represent objects, states, positions, placeholders and are represented by a set of unique vertices. No two vertices represent the same object or state where V can be represented by $\{v_1, v_2, v_3, \ldots, v_n\}$. For each graph edge $e \in E$, there is associated a pair of graph vertices u, v. Mathematically this can be formulated as $\forall_{e \in E} e \rightarrow (u, v)$ where $u, v \in V$. Edges can be directed or undirected and can be weighted (or labeled) or unweighted. An undirected edge $e = (v_i, v_j)$, with $v_i, v_j \in V$, indicates that the relationship or connection is bi-directional, that is, can go from v_i to v_j and vice versa. A directed edge $e = (v_i, v_j)$ specifies a one-directional relationship or connection, that is, can only go from v_i to v_j; this means that $(v_i, v_j) \neq (v_j, v_i)$. The total number of vertices n of graph G, mathematically $|V| = n$, is called the graph *order*. The total number of edges $|E| = m$ is known as the *size* of the graph G. The maximum number of edges in a undirected graph is $m_{max} = \frac{n(n-1)}{2}$, while for the directed ones, it is $m_{max} = n(n-1)$. The representation of graphs is done via two distinct types of graph-theoretic data structures: *list structures* and *matrix structures*. List structures, such as incidence lists and adjacency lists, reduce the required storage space for sparse graphs. Matrix structures such as incidence matrices, adjacency matrices, sociomatrices, Laplacian matrices and distance matrices are appropriated to represent the full matrices with dimension $n \times n$, where n is the total number of vertices of the graph. **Figure 1** shows several types of graphs that can be used to model different kinds of social networks. The classification of graphs is done according to

http://dx.doi.org/10.5772/intechopen.79041

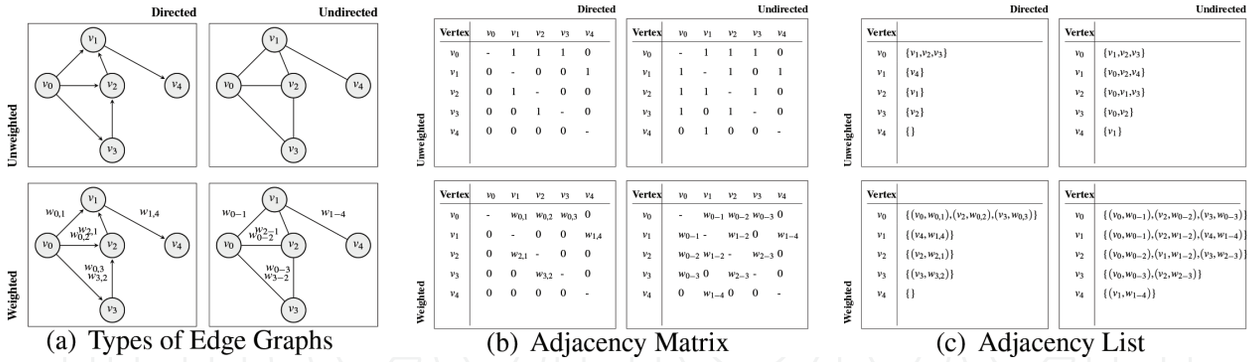
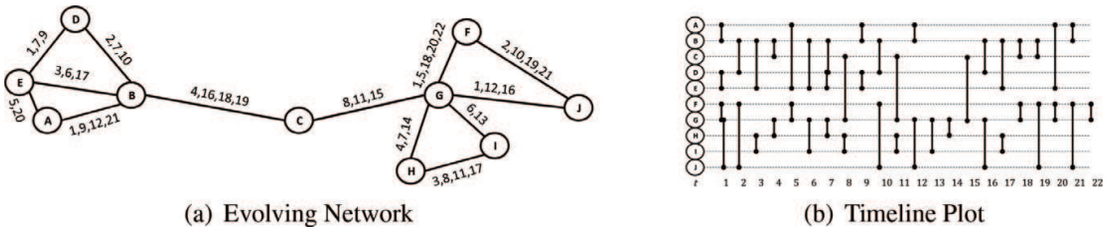


Figure 1. (a) Types of edge graphs and their representation according to an adjacency matrix (b) or an adjacency list (c).

the direction of their links and according to the values assigned to each link. Graphs whose edges, or arcs, connect unordered pairs of vertices or, in other words, each edge of the graph that connects simultaneously two vertices in both directions are called *undirected graphs* or undirected networks. On the other hand, graphs whose all edges, or arcs, have an orientation assigned are called *directed graphs* or directed networks. Formally, a directed graph D is an ordered pair (V, A) consisting of a nonempty set, V, of vertices, and a set A of arcs. These arcs are disjoint from V. If e_{12} is an arc and v_1 and v_2 are vertices such that $e_{12} = (v_1, v_2)$ then e_{12} is said to join v_1 and v_2, v_1 being called the *initial vertex* and v_2 called the *terminal vertex*. Depending on the presence of values assigned to the edges or arcs, the distinction between *unweighted* or *weighted* graphs or networks is made. Unweighted graphs or networks are binary by definition. This means that it is only represented by the presence or non-presence of an edge or arc between two vertices. Unless it is explicitly said, we always assume that graphs are unweighted. In weighted graphs, each edge has associated a weight $w \in \mathbb{R}_0^+$ providing more information about the relation between the two vertices (i.e., the strength of the relation). If e_{12} is an arc between the two vertices v_1 and v_2, w_{12} defines the strength of the connection. For undirected and unweighted graphs, adjacency matrices are binary as a consequence of being unweighted and symmetric as a consequence of being undirected. The edge between vertices v_1 and v_2 is the same, $e_{ij} = e_{ji}$, with $w_{ij} = w_{ji} = 1$. The absence of edges between vertices v_k and v_l is represented by $w_{kl} = w_{lk} = 0$. For directed and weighted graphs, the matrices are nonsymmetric and values from the interval are thus: $[0, \max(w)]$.

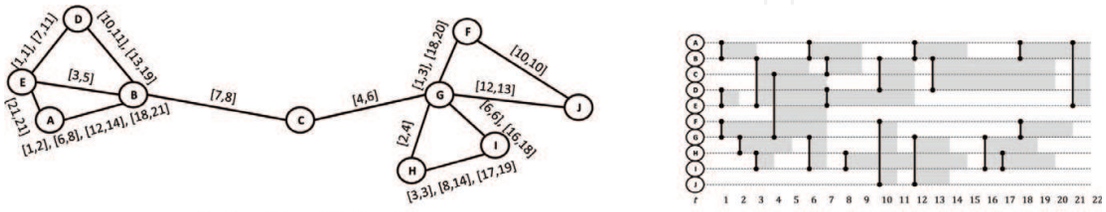
3. From static to evolving networks

Previously, the network types and their representations in a static context were described. In real life, however, many networks are dynamic. As time passes by, new nodes are added to the network, existing ones are removed and edges come and go too. Static networks lack one of the most critical dimensions, that is, the temporal dimension of a network. So by definition static networks are assumed not to change or evolve over time, ignoring the temporal dimension. In this section, we cope with the representation of *evolving networks*. Evolving networks arise in a wide variety of application domains such as the web, social networks and communication



(a) Evolving Network (b) Timeline Plot

Figure 2. Example of contact evolving network: (a) shows a labeled aggregate network where the labels denote the times of contact, and (b) shows a time-line plot, where each of the lines corresponds to one vertex and time runs from left to right.



(a) Evolving Network (b) Timeline Plot

Figure 3. Example of interval evolving network: (a) shows the labeled aggregate network where the labels denote the time interval of the relation, and (b) shows a time-line plot, where each of the lines corresponds to one vertex and gray zones the time duration between two edges.

networks. In recent years the interest in the area of dynamic social networks leads to new research and the need of analysis of evolving networks. The evolution analysis in graphs has applications in a number of scenarios like trend analysis in social networks and dynamic link prediction—to mention two typical examples. **Figure 2** shows the example of a contact evolving network with instantaneous interactions between vertices. When the interaction between network peers has a time duration, we are in the presence of interval evolving networks as shown in **Figure 3(b)**. Assuming that the time T during which a network is observed is finite we can consider the start point as $t_{start} = 0$ and the end time as $t_{end} = T$. A dynamic network graph $G^{D}_{0,T}(V, E_{0,T})$ on a time interval $[0, T[$ consists of a set of vertices or nodes V and a set of temporal edges $E_{0,T}$. The evolving network is a set of graphs across the time axis within discrete time points $t_1, t_2, \ldots, t_{n-1}, t_n$. At time point t_n a graph instance $G(V_n, E_n)$ is observed also denoted as G_n where E_n is the set of temporal edges; $(u, v)_{t_n} \in E_{0,T}$ at time point t_n with edges between vertices u and v on time interval as $t_n = \left[t_{n_{begin}}, t_{n_{end}}\right]$ such that $t_{n_{begin}} \leq T$ and $t_{n_{end}} \geq t_{n_{begin}} \geq 0$. Examples of network changes that may occur between two time points t_{n-1} and t_n are the addition of new edges, that is, $E_n \supset E_{n-1}$, and the appearance of additional nodes, that is, $V_n \supset V_{n-1}$.

3.1. Models of temporal representation

Several models for representing evolving networks are available in the literature. Kim and Anderson [30] introduced the concept of a *time-ordered graph*, and Thompson et al. [29] with a similar conceptual representation chose the *time graphlet* to represent time-varying graphs.

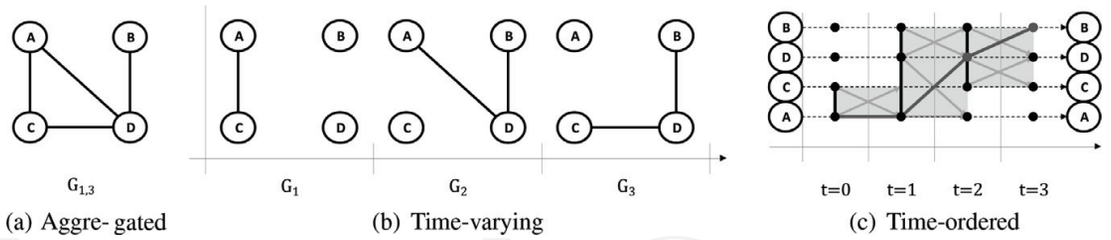


Figure 4. Comparison of aggregated representation (a) and time series representation (b). The corresponding time ordered (c) graph $\mathcal{G}$ is presented for the interval $[0,3]$.

Casteigts et al. [31] presented the *time-varying graph* formalism (TVG) with the concept of a journey to catch the temporal information on graphs. Santoro et al. [32] used this formalism to describe several network measures. Although there are differences in the representation and nomenclature, these models are conceptually equal. **Figure 4** presents the concept of a time-ordered graph for an example network for the time interval $[0,3]$. **Figure 4(a)** shows all the time intervals aggregated into a single graph $G_{1,3}$. The discretization of the network by converting the temporal information into a sequence of n snapshots is presented in **Figure 4(b)**. In this example the evolving network is represented as a series of static networks $G_1, G_2, \ldots, G_n$. The time-ordered graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of **Figure 4(c)** assumes that at each time step, a message can be delivered along a single edge. It is an asymmetric directed graph with a vertex v_t for each $v \in V$ and for each $t \in \{0, 1, \ldots, n\}$ for each edge $(u, v) \in G_t$; it has a directed edge from u_{t-1} to v and vice versa. Although it was not represented in the figure, the time-ordered graph also has edges from v_{t-1} to v_t for all $v \in V$ for all $t \in \{1, \ldots, n\}$. The time-ordered graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ constructed from n static networks of a dynamic network $G^D_{i,j} = \left(V, E_{i,j}\right)$ is a powerful tool to define network metrics and capture their temporal characteristics. In the example of **Figure 4 (c)**, the temporal shortest path from node $u = A$ to node $v = B$ is shown. The temporal shortest path from A to B in the interval $[0,3]$ is $A_0 \rightarrow A_1 \rightarrow D_2 \rightarrow B_3$. The time-ordered graph of Kim and Anderson [30] will be the model used during the course of the rest of the document.

3.2. Timescale of evolving networks

Regarding the evolution of the networks, not all networks evolve equally. Some networks evolve faster than others or have edges that are being added at different rates. Two distinct timescale examples of networks are email networks, where edges are added at the timescale of seconds, and bibliographic networks, where edges are added at the scale of weeks or months. Different time-evolving scenarios require different types of analysis [27]. ***Slowly evolving networks***: When networks evolve slowly over time, *snapshot* analysis can be used very effectively. In this case, dynamic networks are discretized in time by converting temporal information into a sequence of n static snapshots. All the analysis can be done in each snapshot of the network at different times t_1, t_2, …, t_n using static analysis methods. Usually, in slowly evolving networks, a discretization of the time axis into intervals of equal length occurs. For a time discretization in years, days or seconds, a time window size w for each snapshot are set to T/n with n being the number of snapshots. Another solution consists of record buckets of equal size for numerical discretization—a window size w, in this case, is set to a given number

of network updates. In both cases the dynamic network can be represented as a series of static graphs $G_1, G_2, \ldots, G_n$ with $0 \leq n \leq t_n$. A time point t_n is the moment where the network suffers a set of changes in the network represented by the addition or removal of sets of edges ($\{+e_1, +e_2, -e_3, -e_4, \ldots, (+/-)e_n\}$) and/or appearance or disappearance of vertices ($\{+v_1, +v_2, -v_3, -v_4, \ldots, (+/-)v_n\}$). Signs + and − represent the additions or removals, respectively. ***Streaming networks***: When networks are built by a never-ending flow of transient interactions, such as email or telecommunications networks, they should be modeled and represented as graph streams. Graph streams typically require real-time analytical methods. The scenario of graph streams is more challenging because of the computational requirements and the inability to hold complete graphs on memory or disc. Velocity is also an issue because common methods require dealing with graph updates at very high edge rates. Time point t_i is the moment in which a single change in the network occurs. Guha et al. [33] defined this to be the stream model of computation with a stream being a sequence of records $x_1, x_2, \ldots, x_n$ arriving in increasing order of the index i, where x_i may be a new vertex v_i or a new edge e_i.

3.3. Landmark versus sliding windows

When the temporal dimension is added to the analysis of networks, different methodologies regarding the strategy to cope with data that is being analyzed vary. **Figure 5** shows three types of graph data windowing strategies. ***Landmark windows*** by Gehrke et al. [34] encompass all the data from a specific point in time up to the current moment. In the landmark window, the model is initialized in a fixed time point, the so-called landmark that marks the beginning of the window. In successive snapshots, the data window grows to consider all the data seen so far after the landmark. ***Sliding windows*** are better suitable when we are not interested in computing statistics over all events of the past but only over the recent past [35]. Datar et al. [36] incorporate a forgetting mechanism by keeping only the latest information inside the window and disregarding all the data falling outside the window. Usually, the sliding windows are of fixed size. The time-based length sets the window length as a fixed time span. Sliding windows can be overlapping and non-overlapping depending on whether two consecutive windows share some data between them or not. From the several window models presented in the literature [37, 38], two basic types of sliding windows are commonly defined: sequence-based models, where the size of the window is determined regarding the number of observations, and timestamp-based models where the size of the window is defined concerning duration. A timestamp window of size t consists of all elements whose timestamp is within a time interval t of the current period.

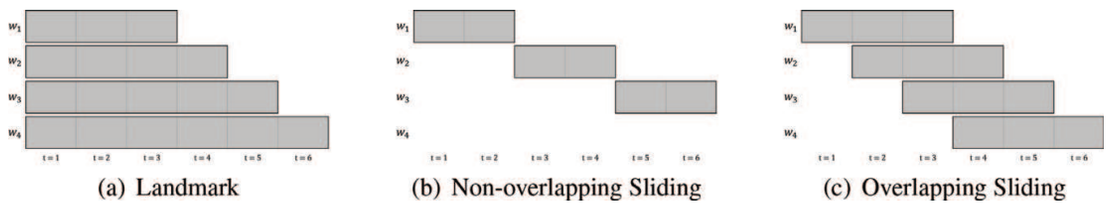


Figure 5. Types of data windows: Landmark window (a) non-overlapping sliding window (b) and overlapping sliding window (c).

3.4. Types of evolving network analysis

Depending on the timescale of the network and the chosen strategy to cope with network data, distinct evolving network analysis methods are available. These methods are divided into one of the following categories [27]. ***Maintenance methods***: In these methods it is desirable to *maintain* the results of the data mining process continuously over time. Examples of maintenance methods are classification and clustering. ***Analytical evolution methods***: In this case it is desirable to directly *quantify* and *understand* the changes that have occurred in the underlying network. Such models are focused on modeling change. ***Bridge methods***: From a methodological point of view, and in the context of a few key problems, an overlapping of maintenance and analytical evolution methods occurs. These *bridge* methods, such as community detection, fall into both categories.

4. Elementary network measures

In this section, elementary network measures and popular metric used in the analysis of social networks are presented.

4.1. Actor-level statistical measures

Actor-level or node-level statistical measures determine the importance of an actor or node within the network. These measures reveal the individuals in which the most important relationships are concentrated and give an idea about their social power within their peers.

4.1.1. Degree or valency

$$D(v) = \sum_{u=1}^{n} a_{u,v}, 0 < D_v < n \tag{1}$$

$$D(v) = |N_v|, 0 < D(v) < n \tag{2}$$

The degree of valency of a node v is usually denoted as $D(v)$ and measures the involvement of the node in the network. It is computed as the number of edges incident on a given node or as the number of neighbors of node v. The neighborhood N_v is defined as the set of nodes directly connected to v. The degree is an effective measure to access the importance and influence of an actor in a network despite some of its drawbacks like not taking into consideration the global structure of the network. In static networks, the degree can be computed via the adjacency matrix by (1) or using the neighborhood of a node (2). Depending on the type of the networks different degree calculation methods should be made for directed and undirected networks and weighted and unweighted networks.

$$D^{+}_{i,j}(v) = \sum_{t=i}^{j} \sum_{u=1}^{n} a_{u,v} \tag{3}$$

$$D^{-}_{i,j}(v) = \sum_{t=i}^{j} \sum_{u=1}^{n} a_{v,u} \tag{4}$$

$$D^{w}_{i,j}(v) = \sum_{t=i}^{j} \sum_{u=1}^{n} a^{w}_{u,v} \tag{5}$$

For dynamic networks, and generalizing to a directed and unweighted network, the temporal degree $D_{i,j}(v)$ is the total number of inbound edges and outbound edges from a node $v \in V$ on a time interval $[i, j]$ where $0 \leq i < j \leq n$. If we disregard the self-edges from v_{t-1} to v_t for all $t \in \{i+1, \ldots, j\}$, $D_{i,j}(v)$ is equal to $\sum_{t=i}^{j} 2.D_t(v)$ where $D_t(v)$ is the degree of v in G_t (i.e., the dynamic graph at time t). For directed networks, there are two variants of degree centrality: considering *in degree*, denoted by $D^{+}_{i,j}(v)$, (3) is the number of incoming edges to node v or edges that end at v and considering *out degree*, denoted by $D^{-}_{i,j}(v)$, (4) is the number of outgoing edges from node v or edges that start at v. For weighted networks, *strength* is the equivalent to degree but is computed as the sum of the weights of the edges adjacent to a given node (5).

4.1.2. Betweenness

$$\mathcal{B}(v) = \sum_{s,\, d \in V(G)\ v} \frac{\sigma_{sd}(v)}{\sigma_{sd}} \tag{6}$$

$$\mathcal{B}_{i,j}(v) = \sum_{i \leq t < j} \sum_{\substack{s \neq u \neq d \in V \\ \sigma_{t,j}(s,d) > 0}} \frac{\sigma_{t,j}(s,d,v)}{\sigma_{t,j}(s,d)} \tag{7}$$

Node betweenness $\mathcal{B}(v)$ measures the extent to which a node lies between the other nodes in the network. For static networks, (6) is used, where σ_{sd} denotes the number of shortest paths between vertices s and d (usually $\sigma_{sd} = 1$) and $\sigma_{sd}(v)$ expresses the number of shortest paths passing through node v. Nodes with high betweenness occupy critical roles in the network structure once their position allows them to work as an interface between different regions of the network. The temporal betweenness $\mathcal{B}_{i,j}(v)$ (7) for a node $v \in V$ on a time interval $[i, j]$ where $0 \leq i < j \leq n$ is the sum of the proportion between all the temporal shortest paths passing by the vertex v and the total number of temporal shortest paths passing over all pairs of nodes in each time interval $\{[t, j] : i < t \leq j\}$. The temporal betweenness for node v is given by (7). Examples of betweenness algorithms are the Brandes algorithm [39], the incremental algorithm proposed by Nasre et al. [40] and the algorithm proposed by Kas et al. [41] for evolving graphs.

4.1.3. Closeness

$$\mathcal{C}(v) = \frac{1-n}{\sum_{u \in V\ v} d(u,v)} \tag{8}$$

$$\mathcal{C}_{i,j}(v) = \sum_{i \le t < j} \sum_{u \in V\ v} \frac{1}{\Delta_{t,j}(u,v)} \tag{9}$$

Closeness measures the overall position of an actor in the network giving an idea of how long it will take, on average, to reach other nodes from a given starting node. It is represented by the average length of the shortest path between the node and all other nodes in the graph. Thus more central a node is, the closer it is to all other nodes. In general, it is only computed for nodes within the largest component of the network as shown in (8). The temporal closeness is defined by considering m intervals $\{[t,j] : i < t \le j\}$ where $m = j - i$ by varying the start time t of each time interval from i to $j-1$ instead of one time interval $[i,j]$ with the starting time as i. Formally the temporal closeness for a node v is given by (9) where $\Delta_{t,j}(u,v)$ is the temporal shortest path distance from u to v on a time interval $[t,j]$. If there is no temporal path from v to u on a time interval $[t,j]$, $\Delta_{t,j}(u,v)$ is defined as ∞. Since the time-ordered graph $\mathcal{G}$ is a directed graph, $\Delta_{t,j}(u,v)$ is different from $\Delta_{t,j}(v,u)$. Regarding the update of closeness centrality in evolving graphs, it was worked on by Kas et al. [42] and Sariyuce et al. [43]. Kas et al. [42] developed incremental closeness centrality algorithms for dynamic networks. An extension of the Ramalingam and Reps [44] algorithm computes the closeness values incrementally, using all-pairs shortest paths for streaming, dynamically changing social networks.

4.1.4. Eigenvector centrality

$$x_v = \frac{1}{\lambda} \sum_{t \in M(v)} x_t = \frac{1}{\lambda} \sum_{t \in G} a_{v,t} x_t \tag{10}$$

$$\mathbf{A}\mathbf{x} = \lambda \mathbf{x} \tag{11}$$

For a given graph $G = (V,E)$ with $|V|$ vertices, let $A = (a_{v,t})$ be the adjacency matrix of an unweighted network, that is, $a_{v,t} = 1$ if vertex v is linked to vertex t, and $a_{v,t} = 0$ otherwise. The relative centrality score of vertex v can be defined by (10), where $M(v)$ is a set of the neighbors of v and λ is a constant. This definition can be rewritten in vector notation as the eigenvector equation using small arrangements is shown in (11). The classical eigenvector centrality had improvements or variants developed to approach the evolving graphs' problem. Examples are Google's PageRank [45] and Katz centrality [46] as the possible variants of this measure as proposed by Society [47]. The concrete implementation of PageRank variants to evolving networks was developed by several researchers, for example, by Bahmani et al. [48], Desikan et al. [49] and Kim and Choi [50]. These improvements over the original PageRank measure show significantly faster results when compared with the original PageRank that, for being an

iterative process, did not scale well to large-scale graphs. This algorithm will be discussed in detail in Section 5.1 .

4.1.5. Laplacian centrality

$$E_L(G) = \sum_{i=1}^{n} \lambda_i^2 \tag{12}$$

$$E_L(G) = \sum_{i=1}^{n} x_i^2 + 2 \sum_{i<j} w_{i,j}^2 \tag{13}$$

The Laplacian centrality permits to consider *intermediate* environmental information around a vertex or node to compute its centrality measure. The centrality of some vertex v is then characterized as a function of the number of 2-walks that vertex v takes part in the network. To estimate the centrality of a vertex, we need to reflect not only the first-order connections but also the importance of their neighbors. These results and related calculations describe the so-called Laplacian energy of the network. Therefore this strategy is known as the Laplacian centrality. The Laplacian energy $E_L(G)$ for a weighted network $G = (V, E, W)$ with n vertices and $\lambda_1, \lambda_2, \ldots, \lambda_n$ eigenvalues of its Laplacian matrix is defined by (12). Considering that $x_1, x_2, \ldots, x_n$ are the vertex sum weights calculated by $x_i = \sum_{j=1}^{n} w_{i,j}$ where $w_{i,j}$ is the weight of the edge from vertex i to j, the Laplacian energy $E_L(G)$ can be computed by (13). The motivation for the incremental Laplacian centrality is supported by the fact that it is known to be a local measure [17, 51].

4.1.5.1. Locality of the Laplacian centrality

The Laplacian centrality metric is not a global measure, that is, it is a function of the local degree plus the degrees of the neighbors (with different weights for each). Qi et al. [17, 51] show that local degree and the 1-order neighbors' degree are all that are needed to calculate the metric for unweighted networks (**Figure 6**).

4.1.5.2. Dynamic Laplace centrality

Regarding the original Laplace centrality algorithm proposed by Qi et al. [51], despite being a static algorithm, it can be used to calculate centralities in changing networks. This is true by

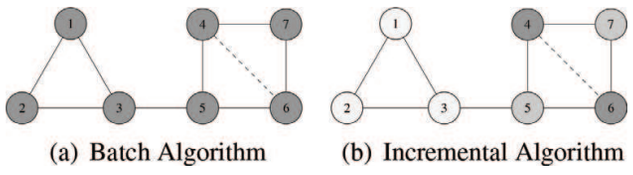


Figure 6. Calculated node centralities with edge {(4, 6)} added. Dark gray nodes affected by addition of edges. Light gray nodes centralities need to be calculated due to their neighborhood with affected nodes.

considering full calculations of the centralities for each network snapshot. In Sarmento et al. [52] proposal, Qi et al. [51] principles were adapted and resulted in two incremental algorithms. The incremental Laplace algorithm by Sarmento et al. [52] presents better computational efficiency, by performing careful Laplace centrality calculations only for the nodes affected by the addition and removal of edges in each one of the snapshots. Thus, it reuses information of the previous snapshot to perform the Laplace centrality calculations on the current snapshot, for unweighted networks only.

4.2. Network-level statistical measures

Before describing network-level statistical measures, it is important to describe three fundamental concepts that are common to static and dynamic networks. ***Path*** represents a sequence of nodes in which consecutive pairs of non-repeating nodes are linked by an edge. When adding the temporal dimension of dynamic networks, the concept of path slightly changes, because non-repeating nodes are now considered only within the same snapshot or time step in what is called a temporal path. Temporal paths can have repeating nodes in different time steps, for example, $A_0 \rightarrow B_0 \rightarrow C_1 \rightarrow B_1 \rightarrow A_2$. ***Geodesic distance***, or the shortest path, between nodes u and v is denoted as $\delta(u, v)$ and defines the length of the shortest path, or minimal path, between nodes u and v in a static graph. For a given time-ordered graph $\mathcal{G}$, a temporal path from node u to node v on time interval $[i, j]$ where $i \leq i < j \leq n$ is defined as any path, $p = < u_i, \ldots, v_i >$ where $i < k \leq j$, having the path length $|p| = \min_{i<l\leq j}\delta(u_i, v_l)$. $\delta(u, v)$ is the shortest path distance, in a static graph, from u to v. The temporal shortest path from node u to node v is defined as the temporal path connecting u to v which has minimum temporal length. In **Figure 4**, an example of a temporal shortest path in a time-ordered graph was shown. ***Eccentricity*** is the greatest geodesic distance between a given vertex v and any other in the graph, that is, $\varepsilon_v = \min_{i \in V(G)\setminus v} d(v, i)$.

4.2.1. Edge bursts

$$\mathcal{B}_{ij} = \frac{\sigma(\tau_{ij}) - \sigma\left(\mu_{ij}\right)}{\sigma(\tau_{ij}) + \sigma\left(\mu_{ij}\right)} \tag{14}$$

A hallmark of a bursty edge is the presence of multiple edges with short interconnect times, followed by longer and varying interconnect times [29]. One of the methods available to quantify bursts is the burstiness coefficient $\mathcal{B}$. Presented by Goh and Barabasi [53], it can be formulated for discrete graphs [54] where bursts are computed by edges using (14), with τ_{ij} being a vector of the intercontact times between nodes i and j though time, $\sigma(\tau)$ is the standard deviation and $\sigma(\mu)$ is the mean. For temporal connectivity being considered as bursty, that is, $\mathcal{B} > 0$, it occurs when the standard deviation $\sigma(\tau)$ is greater than the mean $\sigma(\mu)$.

4.2.2. Fluctuability

$$\mathcal{F} = \frac{\sum_i \sum_j U(A_{i,j})}{\sum_i \sum_j \sum_t A_{i,j}^t} \tag{15}$$

$$\mathcal{F}_i^N = \frac{\sum_j U(A_{i,j})}{\sum_j \sum_t A_{i,j}^t} \tag{16}$$

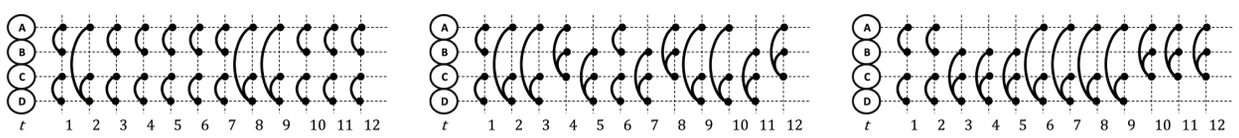
As discussed before, centrality measures provide information about the degree of temporal connectivity while bursts describe the distribution of the temporal patterns of connectivity at the node level. Also, *fluctuability* can be used to retrieve information about the global state of a temporal network, in this case, the quantification of the temporal variability of connectivity [29]. The fluctuability $\mathcal{F}$, as shown in (15), is the ratio of the number of edges present in A over the sum of A_t, with U being a function of the binary output: $U(A_{i,j})$ is set to 1 if at least one of the edges occurs between nodes i and j across time $t = 1, 2, \ldots, T$ and 0 if not. T is the number of time points. The maximum value of $\mathcal{F}$ is 1 and occurs only when every edge is unique and occurs only once. The definition of fluctuability $\mathcal{F}_i^N$ at the node level, when $U(A_{i,j}) > 0$, is defined using (16); when $U(A_{i,j}) = 0$, $\mathcal{F}_i^N$ is equal to 0 (**Figure 7**).

4.2.3. Volatility

$$\mathcal{V} = \frac{1}{T-1} \sum_{t=1}^{T-1} D(A^t, A^{t+1}) \tag{17}$$

$$\mathcal{V}_{i,j}^L = \frac{1}{T-1} \sum_{t=1}^{T-1} D\left(A_{i,j}^t, A_{i,j}^{t+1}\right) \tag{18}$$

The *volatility* $\mathcal{V}$ is a global measure of temporal order that represents how much, on an average, the connectivity between consecutive temporal time-ordered graphs changes [29]. This measure indicates how volatile the temporal network is over time and is computed by (17), where D is a distance function and T is the total number of time points. The distance function quantifies the difference between the temporal time-ordered graph G_t and the



(a) Network with the lowest fluctuability (F = 0.125) and volatility (V = 0.73).

(b) Network with highest volatility (V = 2.55) and fluctuability (F = 0.25) equal to the network C

(c) Lower volatility than B (V = 1.27) but equal fluctuability (F = 0.25) equal to B

Figure 7. Variation of Fluctuability and volatility measures over three different evolving contact networks.

temporal time-ordered graph G_{t+1}. One example of a distance function for volatility can be the Hamming distance. Volatility can be defined at the local level, for example, a per-edge volatility can be computed using (18). An estimate of the volatility centrality of node i can be computed by taking the mean $V^L_{i,j}$ over j (**Figure 7**).

4.2.4. Reachability latency

$$\mathcal{R}_r = \frac{1}{TN} \sum_t \sum_i d^t_{i,k} \tag{19}$$

$$\mathcal{R}_1 = \frac{1}{TN} \sum_t \sum_i \max_j d^t_{i,j} \tag{20}$$

Reachability measures, like *reachability ratio* and *reachability time*, focus on estimating the time taken to reach the nodes in a temporal network [55]. While the reachability ratio is the percentage of edges that have a temporal path connecting them, the reachability time is defined as the average length of all temporal paths. When applying these reachability measures to most real-world networks, if we consider a sufficient time interval, any vertex or node of the networks can reach all the others within that time span. With this assumption in mind, Thompson et al. [29] defined the *reachability latency*, which quantifies the average time it takes for a temporal network to reach an a-priori-defined reachability ratio as defined in (19), where d^t_i is an ordered vector of length N of the shortest temporal paths for node i at time point t. Value k represents the $[rN]$th element in the vector. In case $r = 1$, that is, all nodes are reachable, the former formula can be simplified to (20), which is also known as the *temporal diameter* of the network [56].

4.2.5. Temporal efficiency

$$\mathcal{E} = \frac{1}{T\left(N^2 - N\right)} \sum_{i,j,t} \frac{1}{d^t_{i,j}}, i \neq j \tag{21}$$

For static networks, efficiency is computed as the inverse of the average shortest path for all nodes [29]. *Temporal efficiency*, at first, is calculated at each time point as the inverse of the average shortest path length of all nodes; subsequently, these values are averaged across time points to obtain an estimate of global temporal efficiency as shown in (21).

4.2.6. Diameter and radius

The diameter D is given by the maximum eccentricity of a set of vertices $D = \max\{\varepsilon_v : v \in V\}$ and, analogously, the radius R is defined as the minimum eccentricity of the set of vertices $R = \min\{\varepsilon_v : v \in V\}$.

4.2.7. Average geodesic distance

$$L(G) = \frac{1}{\frac{1}{2}n(n-1)} \sum_{u \geq v} \delta(u, v) \tag{22}$$

$$L(G) = \frac{1}{n(n-1)} \sum_{u \geq v} d(u, v) \tag{23}$$

The average geodesic distance L gives an idea on how far apart nodes will be, on average. For static networks, all combinations of vertex pairs in a network are computed as in (22), where $\delta(u, v)$ is the geodesic distance between nodes u and v and $\frac{1}{2}n(n-1)$ is the number of possible edges in a network of n nodes. Tang et al. [57, 58] defined the *characteristic temporal path* length as the natural extension of the average geodesic distance to time-varying graphs. It is defined as the average temporal distance over all pairs of nodes in the graph as shown in (23), with the temporal distance $d_{u,v}$ between u and v as the temporal length of the temporal shortest path from u to v.

4.2.8. Average degree

$$D(G) = \frac{1}{n}\frac{1}{t} \sum_{i=0}^{t} \sum_{u=1}^{n} \sum_{v=1}^{n} a_t[u, v] \tag{24}$$

The average degree is the mean of the edges of all vertices in a network for all time steps t, with a_t being the adjacency matrix at time t, as shown in (24).

4.2.9. Reciprocity

$$r = \frac{\#mutual}{\#mutual + \#asymmetric} \tag{25}$$

For static networks, reciprocity r is a specific quantity of directed networks that measures the tendency of pairs of nodes to form mutual connections between each other. The value of reciprocity represents the probability that two nodes in a directed network point to each other. For each of the $(n(n-1))/2$ dyads in the network are assigned to one of the three types: mutual, that is, node i has a tie to node j and node j has a tie to i; asymmetric, that is, either i has a tie to j or j has a tie to i but not both; or null, that is, neither the i to j tie nor the j to i tie is present [59]. Given this, reciprocity can be computed using (25), where *#mutual* denotes the number of mutual dyads and *#asymmetric* the number of asymmetric dyads. In an undirected network, reciprocity is always maximum ($r = 1$) because all pairs of nodes are symmetric, that is, dyads are of the type mutual. Brandenberger [24] analyzed the temporal dynamics of reciprocity in congressional collaborations using relational event models.

4.2.10. Density

$$\rho(G) = \frac{m(G)}{m_{max}(G)} \tag{26}$$

Density ρ explain the general level of connectedness of a network. It is computed by measuring the proportion of edges in the network relative to the maximum possible number of edges as seen in (26), where $m(G)$ is the total number of edges of network G and $m_{max}(G)$ the number of possible edges of network G, which is $\frac{n(n-1)}{2}$ for undirected networks and $n(n-1)$ for directed ones.

4.2.11. Global clustering coefficient

Cui et al. [60] propose two definitions of the temporal clustering coefficient of a temporal network. The definitions are temporal-delayed clustering coefficient and the temporal-weighted clustering coefficient.

5. Link analysis

In network theory, link analysis is a data analysis technique used to evaluate relationships (connections) between nodes. Link analysis has been used for the investigation of fraud detection, terrorist networks, computer security analysis, search engine optimization (SEO), market research and medical research, among others. To find the most valuable, authoritative or influential node or the list of nodes in networks, link analysis algorithm were devised to solve this problem in the past. By exploring the relationship between links and the content of web pages, the PageRank algorithm [45] is one of the seminal methods employed to the build of modern and efficient search engines and the information retrieval system in the web.

5.1. Incremental PageRank algorithm

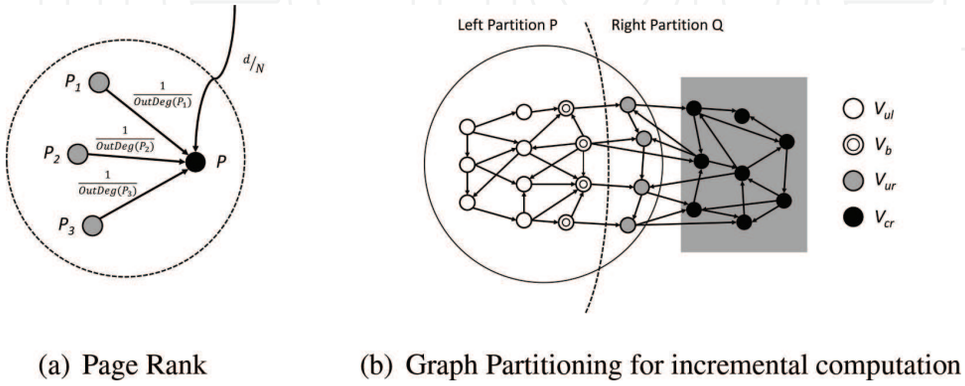
There have been several attempts to improve the original PageRank algorithm [45]. The purpose of several of these improvements was to adapt this algorithm for streaming data. In the original algorithm, each page rank is dependent of the ranks of the pages pointing to it. The PageRank value of a page p is mathematically written as:

$$PR(p) = {}^{d}/_{n} + (1-d) \sum_{q,\, p \in G} {}^{PR(q)}/_{OutDegree(q)} \tag{27}$$

where n is the number of vertices/pages in the graph and $OutDegree(q)$ is the number of hyperlinks on the node/page q. Eq. (27) illustrates an example of computing PageRank of a page P from the pages (**Figure 8(a)**) P_1, P_2 and P_3 pointing to it using (28):

$$PR(p) = {}^{d}/_{N} + (1-d)\left(\frac{PR(P_1)}{OutDeg(P_1)} + \frac{PR(P_2)}{OutDeg(P_2)} + \frac{PR(P_3)}{OutDeg(P_3)}\right) \tag{28}$$

Desikan et al. [49] provided a solution for the update of nodes' PageRank values in evolving graphs in an incremental fashion. The algorithm explores the fact that the web evolves incrementally and with small changes between updates. **Figure 8(b)** shows the two partitions created; one of them, partition P, is unchanged since the last computation, and it has only outgoing edges to the other partition. The other partition, partition Q, is the rest of the graph, which has changed since the last time the metric was computed. The principal idea is to find a partition P in a way that there are no incoming links in the graph from the other partition Q (includes all changed nodes). Then the computation of the PageRank of partition Q can be done separately, scaled and merged with the rest of the graph to get the updated PageRank values of the vertices in this partition. The PageRank of partition Q is computed, taking the border vertices that belong to partition P and have edges pointing to the vertices in partition Q. The PageRank values of partition P are obtained by simple scaling, due to the addition of new nodes. Let the graph of **Figure 8(b)** at the new time be $G(V, E)$: V_b is the vertex on the border of the left partition (only outgoing edges to the right partition); V_{ul}, vertex on the left partition remains unchanged; V_{ur}, vertex on the right partition remains unchanged but whose PageRank is affected by vertices in the changed component; V_{cr} is the vertex on the right partition which has changed, or there has been a new addition. Desikan et al. [61] proposed a divide and conquer approach for efficient PageRank computation based on these assumptions. Other page rank analysis algorithms, in which it is desirable to estimate the page rank on a dynamic evolving graph stream, are available in [62, 63]. Sarma et al. [62] developed a method that can estimate the page rank distribution, the mixing time and the conductance of the graph. Bahmani et al. [63] developed a method for real-time estimation of the personalized page rank in graph streams. Zhang et al. [64] proposed a method for an approximate personalized PageRank on dynamic graphs. The update of PageRank node values in dynamic graph streams has been extensively used to leverage the efficiency of large-scale link analysis, inclusively with applications that have known issues with scalability like, for example, text streams [65].



(a) Page Rank (b) Graph Partitioning for incremental computation

Figure 8. PageRank and graph partitioning used in Desikan et al. [49] incremental page rank.

5.2. Link prediction

To understand the association between two specific nodes, researchers commonly study the dynamics of evolving graphs. In link prediction, the problem we wish to solve is the prediction of the likelihood of a future association between two nodes, knowing that there is no association yet between the nodes, that is, no edge between the nodes. Link prediction is used in bioinformatics, where potential protein connections are inferred from known connections, and during the research of terrorist/criminal networks, where potential criminal connections are inferred from current knowledge of the relationships between criminals. Link prediction is a complex problem. For a social network $G(V, E)$, there are $n^2 - m$ possible edges to infer from our current graph. This is true if we randomly select a non-existing edge. If G is sparse, then $m \approxeq n$. Thus, in limit situations, with a high amount of nodes, we have a n^2 edges to choose from, and the probability of inferring correctly at random is $1/n^2$. Commonly, social networks derived from real-world phenomena are sparse, so inferring random edges is expected to have low accuracy. As networks evolve, it is expected that nodes are added to the network, and the number of possible links grows quadratically while it is expected that new edges grow in a linear fashion with added new nodes. Thus, it is a problem that gets worse in evolving networks as time goes forward.

5.2.1. Common neighbors

Newman has verified a significant correlation between the number of common neighbors of u and v at time t, and there is the probability that u and v will connect or collaborate at sometime after t [66]. The common-neighbors predictor concept is based on the assumption that two—not yet connected—nodes with common neighbors will get connected sometime in the future. This introduction between unconnected nodes means the effect of "closing the triangle".

5.2.2. Jaccard coefficient

The Jaccard coefficient is a common metric in measuring the similarity between different samples. It is used throughout validation tasks in information retrieval research. It measures the probability that both u and v have a feature f, for a randomly selected feature f that either u or v has.

$$JC = \frac{|N_u \cap N_v|}{|N_u \cup N_v|} \tag{29}$$

where N_u is the list of neighbors of u, and N_v is the list of neighbors of v. Thus, in (29), the numerator is the number of common neighbors, and the divisor is the number of unique u and v neighbors.

5.2.3. Adamic/Adar

This measure—also called frequency-weighted common neighbors—refines the simple counting of common features in JC by weighting rarer features more heavily [67]. The Adamic/Adar predictor is based on the intuitiveness of considering unusual features more critical in predicting future outcomes. In this example u and v would have to be introduced by

a common friend z, person z will have to choose to introduce the pair (u,v) from N_z pairs of his connections.

$$PA = \sum_{z \varepsilon N_u \cap N_v} \frac{1}{\log|N_z|} \tag{30}$$

5.2.4. Preferential attachment

Another concept, this time with lower complexity, is the preferential attachment [68]. This metric is only in need of node degree information. The intuitiveness is that those nodes with a higher degree have more probability to connect to each other than with a neighbor with a lower degree.

$$PA = |N_u| * |N_v| \tag{31}$$

5.2.5. Katz

The Katz concept [46] is based on the assumption that the closer connected nodes are with a higher number of paths in the network, and these nodes will have more probability of connecting in the future. The concept is also called "exponentially damped path counts".

$$Katz_{score} = \sum_{L=1}^{\infty} \beta^L |Path^L_{u,v}| \tag{32}$$

where β^L is exponentially damped by length $|Path^L_{u,v}|$ of the *Path*, with L being the number of hops between u and v.

5.2.6. Recent developments

Ibrahim and Chen [69] present a method for link prediction in dynamic networks by integrating temporal information, community structure and node centrality in the network providing greater weights for frequently occurring links. Wahid-Ul-Ashraf et al. [70] described the parallelism between Newton's law of universal gravitation and the link prediction tasks. To apply this law, the authors attributed nodes with the notion of mass and distance. Node centrality could be considered as mass, and the authors inclusively tested this concept with degree centrality. The distance between nodes was considered obtainable through several possible methods, that is, by retrieving the shortest path, path count or inverse similarity, by using previously stated measures like Adamic/Adar, Katz score or others. Choudhury and Uddin [71] considered the evolutionary aspects of community network structure. They build dynamic similarity metrics or dynamic features to measure similarity/proximity between actor pairs.

6. Community detection

As a consequence of both global and local heterogeneity of edge distribution in a graph, specific regions of a graph evidence the high concentration of edges within particular regions,

called *communities*, whereas interregions have low concentrations of edges. In the context of networks, these occurrences of groups of nodes in a network that are more densely connected internally than with the rest of the network are called *community structures*. Also known as *modules* or *clusters*, communities can, therefore, be straightforwardly defined as similar groups of nodes. A complete definition using the concept of density can be the following: communities can be understood as densely connected groups of vertices in the network, with sparser connections between them.

6.1. Finding communities in static networks

Fortunato [72] has a comprehensive survey about methods and techniques regarding finding communities. Hierarchical clustering methods can be of two types: agglomerative algorithms, in which clusters are iteratively merged if their similarity is sufficiently high, and divisive algorithms, in which clusters are iteratively split by removing edges connecting vertices with low similarity. ***Divisive algorithms***: One of the most known divisive algorithms is the one proposed by Girvan and Newman [73]. The philosophy of divisive algorithms is the idea that a simple way to identify communities in a graph is to detect the edges that connect vertices of different communities and remove them so that the clusters get disconnected from each other. ***Agglomerative algorithms***: Examples of agglomerative algorithms are the ones that assume that high values of modularity indicate good partitions. So the partition corresponds to maximum value of modularity on a graph. Therefore a modularity measure $\mathcal{Q}$ is used to evaluate the quality of the community structure of a graph. Modularity serves as the objective function during the process of calculating the communities [74]. Modularity $\mathcal{Q}$ with higher values means better community structures. Therefore, to obtain a global higher modularity, the objective is to find community assignments for each one of the nodes of the network such that $\mathcal{Q}$ is also maximized. A greedy algorithm based on modularity optimization has been introduced by Blondel et al. [75] where initially all vertices of the graph are put in different communities (**Figure 9(b)**). The first step consists of a sequential sweep over all vertices, for each of the neighbors picks the community that yields the largest increase of modularity

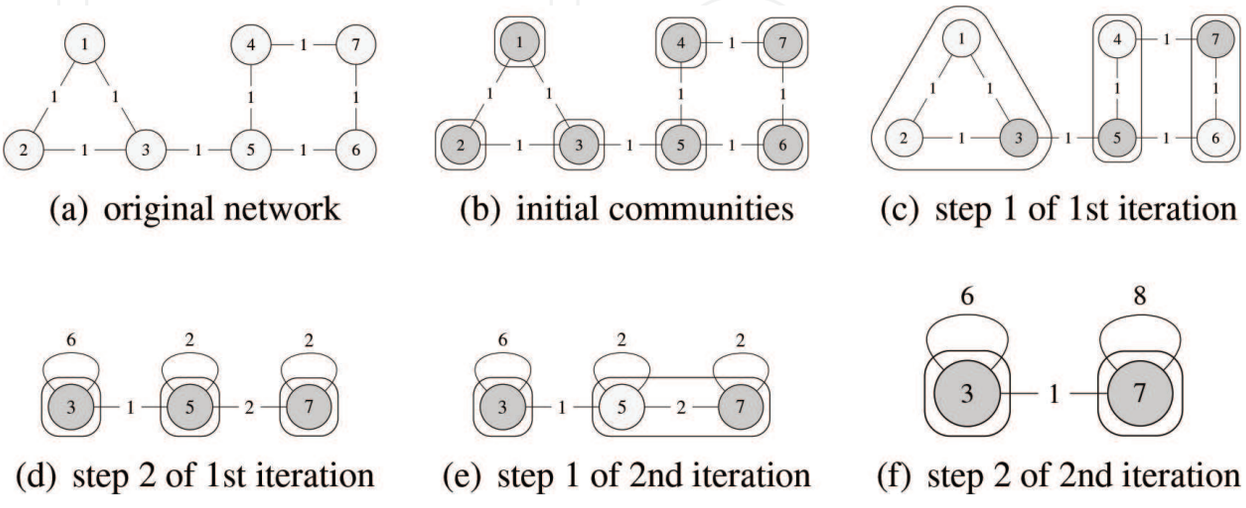


Figure 9. Example of an agglomerative community detection algorithm. In this case the original Louvain [75] with all algorithm steps.

(**Figure 9(c)**). At the end of the sweep, one obtains first-level partition. In the second step, communities are replaced by super vertices, and the weight of the edge between the super vertices is the sum of the weights of the edges between the represented communities at the lower level (**Figure 9(d)**). The two steps of the algorithm are then repeated, yielding new hierarchical levels and supergraphs (**Figure 9(f)**).

6.2. Finding communities in dynamic networks

When discussing methods for finding communities in dynamic networks, the division of methods for slowly evolving networks and streaming networks is consensual [27]. In the following section, an algorithm for both scenarios will be presented and analyzed.

6.2.1. Slowly evolving networks

When moving from static community detection to dynamic community detection, often, static techniques are used to detect communities in evolving network. The Louvain algorithm by Blondel et al. [75] is no exception, and it is still one of the fastest ways to perform community detection on evolving networks by considering individual static snapshots. Frequently employed in dynamic network community detection scenarios by performing individual runs of the algorithm in snapshots of the network, this approach is computationally inefficient and does not allow the tracking of communities in a fine-grained way between static snapshots. The community detection work referenced in the Fortunato [72] survey was later complemented by an incremental community detection algorithm based on modularity and was proposed by Shang et al. [76]. The algorithm applies the principles of events in the life of communities (growth, contraction, merging, splitting, birth and death) as defined by Palla et al. [77] and, in each one of the iterations, calculates the modularity gain of affected communities. This allows to detect and track communities over time in incremental networks. This algorithm only considers the addition of new edges and relies on the original two-step approach used in community detection for static communities. The QCA [78], presented as a fast and adaptive algorithm, provides efficient identification of the community structure of dynamic social networks by allowing the addition and removal of nodes and edges dynamically. The algorithm starts with the initial communities calculated via the Louvain method, and then it applies the adaptive node community changes by considering each node as an autonomous agent demonstrating flocking behavior toward their preferable neighboring groups [79]. The AFOCS [80] community detection algorithm for dynamic networks shares the same principles of QCA being only modified in order to allow the possibility of detection of overlapping communities. A detailed comparison between QCA and AFOCS was presented by Nguyen et al. [80]. Label propagation techniques and specifically speaker-listener label propagation (SLPA) were used in community detection over large networks. LabelRank [81] and GANXiSw [81, 82] used the SLPA technique to perform static network community detection while LabelRankT [83] was designed to handle dynamic networks. Being designed for overlapping community detection, all of the previous algorithms also work in a non-overlapping mode, with satisfactory performance for low overlapping density networks [84].

http://dx.doi.org/10.5772/intechopen.79041

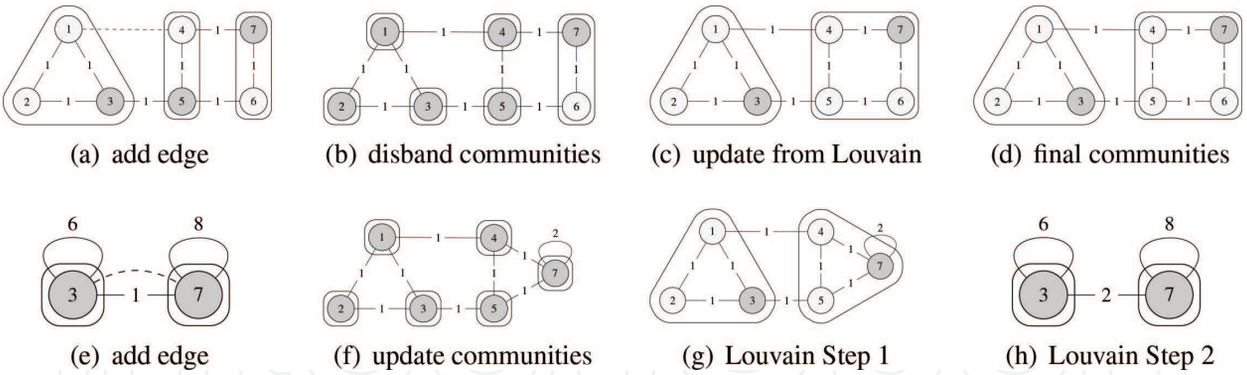


Figure 10. Example of Cordeiro et al. [85] dynamic Louvain for the addition of a cross-community edge (1–4). Top figures show the lower-level network. At the bottom, are shown, the corresponding upper-level network with aggregated communities.

Cordeiro et al. [85] presented a modularity-based dynamic community detection algorithm. The algorithm is a modification of the original Louvain method where dynamically added and removed nodes and edges only affect their related communities. In each iteration, the algorithm remains unchanged in all the communities that were not affected by modifications to the network. By reusing community structure obtained by previous iterations, the local modularity optimization step operates in smaller networks where only affected communities are disbanded to their origin. The stability of communities is also an improvement over the original algorithm (**Figure 10**). Given that only parts of the network change during iterations, the nondeterminism of the algorithm will have a reduced effect on the community assignment. Most node-community assignments remain unchanged between snapshots, providing better community stability than its counterparts.

6.2.2. Streaming networks

For the cases when a large number of edges, representing interactions, arrive continuously, in some cases at high or very high rates, and are superposed over much larger networks, streaming graph algorithms should be preferred to perform community detection. In streaming scenarios, the ability to perform the deletion of edges in community detection algorithms is important. In short, as discussed in Section 3.3, this will dictate if the method of analysis is to be performed over the sliding window of edges, and therefore edges are deleted from the tail end of the sliding window, or over a landmark window, in case there is no possibility to delete or forget old edges. Several methods were proposed for dynamic community discovery in graph streams. Wang et al. [86] motivated by the variability of the underlying social behavior of individuals over different graph regions modeled the problem according to the so-termed *local heterogeneity*, where a local weighted-edge-based pattern (LWEP) summary is efficiently maintained and used afterward to cluster the graph stream and perform dynamic community detection in weighted graph streams. Taking an almost linear time, Raghavan et al. [87] investigated a simple label propagation algorithm that uses the network structure alone as its guide and requires neither optimization of a predefined objective function nor prior information about the communities. By analyzing the problem of real-time community detection in large networks and having by baseline the algorithm proposed by Raghavan et al. [87] with

linear time $O(m)$ on a network with m edge-label propagation, or "epidemic" community detection, Leung et al. [88] proposed a method with near-linear time community detection in graphs. Leung et al. identified the characteristics and drawbacks of the base [87] algorithm and extended it by incorporating different heuristics to facilitate reliable and multifunctional real-time community detection. Yun et al. [89] proposed two efficient streaming memory-limited clustering algorithms for community detection based on spectral methods. Yun and Proutière [90] proposed community detection via random and adaptive sampling. Sariyuce et al. [91] proposed SONIC, a find-and-merge type of overlapping community detection algorithm that can efficiently handle streaming updates. Recently, Hollocou et al. [92] proposed SCoDA, a linear streaming algorithm, for community detection in very large networks.

7. Visualization of evolving networks

The visualization of networks is known to be challenging, and this task gains additional complexity when moving from static to evolving networks. In this section an overview of the methods and techniques is presented, currently used for the visualization of evolving networks.

7.1. Challenges of evolving networks' visualization

The dynamics of social networks remain a challenge regarding visualization [93]. Many researchers argue that traditional graph visualization methods have issues when applied to evolving networks. Additionally, the application of conventional node-link methods to large-scale networks provides low-quality cluttered insights. The overlap of nodes in these conditions is not appropriated when trying to extract information from the network. Zaidi et al. [94] and Aggarwal and Subbian [27] presented an overview of the different techniques and methods that exist for the analysis and visualization of dynamic networks. It included the discussion of the basic definitions, formal notations and a set of the most important and recent work regarding analysis and the visualization of dynamic networks. While static graph visualizations are often divided into node-link and matrix representations, Beck et al. [95, 96] presented a hierarchical taxonomy of dynamic graph visualization techniques. This survey about the state of the art in visualizing dynamic graphs identified the representation of time as the major distinguishing feature for dynamic graph visualizations. Two major visualization categories were found: in one category, graphs are represented as animated diagrams or in a second one, visualizations are a set of static charts based on a timeline. Similar conceptual dynamic network visualization categories were devised by Moody et al. [97], and the authors divide dynamic network visualizations also called as network movies into static flip books, where the node position remains constant but edges cumulate over time and dynamic movies, where nodes move as a function of changes in relations. The graph animation is often used to lower the cognitive effort required to follow the transition from one visualization to the next, according to Brandes and Corman [98]. To facilitate the simultaneous analysis of state and change, a layered three-dimensional network visualization was proposed by Brandes and

Corman [98] in which the evolution of the network is unrolled, and each step is represented as a layer. A complex network with a larger number of links may prevent users from recognizing salient structural patterns. To overcome this common problem with visualization, two widely known link reduction algorithms, namely minimum spanning trees (MSTs) and pathfinder networks (PFNETs), were analyzed and compared by Chen and Morris [99]. Bender-deMoll and McFarland [100] propose a framework for visualizing social networks and their dynamics and presented a tool that enables debate and reflection on the quality of visualizations used in empirical research. With the focus on the evolution of communities over time, Falkowski et al. [101] proposed two approaches to analyze the evolution of two different types of online communities on the level of subgroups. This analysis was conducted by observing changes in the interaction behavior of the members of the communities. Chen [102] devised a generic approach for detecting and visualizing emerging trends and transient patterns in scientific literature. Other recent work of interest is presented by Beck et al. [103, 104] and the visualization of evolving graphs with multiple visual metaphors of Burch [105]. The combination of dynamic network visualization with graph sampling techniques is often used [106].

8. Conclusion and future trends

This chapter provided an overview of the methods and techniques for modeling, analyzing, measuring and visualizing evolving social network analysis. In the past, static techniques were adapted to dynamic networks with relative success, but nowadays, with the advent of social media, scale and velocity of most of those static techniques reveal weaknesses that only can be addressed by methods and techniques designed for dealing with evolving data. After presenting two areas of direct applicability of evolving network analysis such as criminological research and research on journalism, the ways on how dynamic networks can be represented and modeled according to their timescale, windowing strategies and methods of analysis were discussed. These theoretical aspects were then used to present elementary network measures, link analysis methods, community detection methods and visualization techniques. It is clear that in recent years this area of research will continue to have significant development in the future, several problems are still unsolved and many of them can be significantly improved. The areas of applicability of evolving networks and social network analysis are also broader, with many of the abovementioned techniques moving from well-succeeded areas like world wide web, communication, telecommunication and mobile networks, to newer areas like social network recommendations, news and blog analysis and social network event detection. Specifically in the area of social network event detection, the detection of unusual patterns, anomalies or changes in trends in the social streams can lead to valuable information, which can be used timely in many real-word scenarios [107]. Cordeiro [108] addressed the monitoring and tracking of the dynamics of social network communities with the objective to unveil real-world events, whereas Cordeiro [109] was devoted to the problem of mining the twitter stream to unravel events, interactions and communities in real time. Future trends of social network analysis will continue to be driven by future trends and characteristics of the network data, such as the size of data, which is incredibly getting large,

and changes in space and time. On one side, there is the urge for scalable and efficient social network analysis methods, and on the other side, there is the need for methods to study the dynamics and evolution of social networks, able to deal with future velocity and timescale dimensions of the network data. Stray [110] focused network analysis as a tool to bridge the "research to reporting" gap in journalism, starting with two use cases (Seattle Art World [111] and Hot Wheels [112]) and the recent state-of-the-art network analysis and visualizations applied to the Panama Papers case [113] where graph databases and entity recognition were used to build interactive network maps from structured data and raw documents. Therefore, it is expected that the study of evolving networks will continue to be a significant strand of research in the context of social network analysis in the near future.

Acknowledgements

This work was fully financed by the Faculty of Engineering of the Porto University. Rui Portocarrero Sarmento also gratefully acknowledges funding from FCT (Portuguese Foundation for Science and Technology) through a PhD grant (SFRH/BD/119108/2016). The authors also want to thank the reviewers for the constructive reviews provided in the development of this publication.

Author details

Mário Cordeiro[1]*, Rui P. Sarmento[1], Pavel Brazdil[2] and João Gama[2]

*Address all correspondence to: mario.miguel.cordeiro@gmail.com

1 Faculty of Engineering, University of Porto, Portugal

2 INESC TEC–LIAAD, Portugal

References

[1] Castells M. The Rise of the Network Society: The Information Age: Economy, Society, and Culture, volume I. Chichester, West Sussex: Wiley-Blackwell; 2010. ISBN: 9781405196864

[2] Watts DJ. The "new" science of networks. Annual Review of Sociology. 2004;**300**(1):243-270. ISSN: 0360-0572

[3] Ognyanova K, Monge P. A multitheoretical, multilevel, multidimensional network model of the media system: Production, content, and audiences. Annals of the International Communication Association. 2013;**370**(1):67-93. ISSN: 2380-8985

[4] Sakaki T, Okazaki M, Matsuo Y. Earthquake shakes twitter users: Real-time event detection by social sensors. In: Proceedings of the 19th international conference on World Wide Web. Raleigh, North Carolina, USA: ACM; 2010. pp. 851-860. ISBN: 9781605587998

[5] Kwak H, Lee C, Park H, Moon S. What is Twitter, A social network or a news media? In Proceedings of the 19th international conference on World wide web (WWW '10). New York, USA: ACM; 2010;**1120**(2):591-600. DOI: http://dx.doi.org/10.1145/1772690.1772751

[6] Hounshell B. The Revolution Will Be Tweeted—Life in the Vanguard of the New Twitter Proletariat; 2011

[7] Lardinois F. The Short Lifespan of a Tweet: Retweets Only Happen Within the First Hour; 2010

[8] Barabási A-L. The origin of bursts and heavy tails in human dynamics. Nature. May 2005;**4350**(7039):207-211. ISSN: 1476-4687

[9] Coles N. It's not what you know—It's who you know that counts: Analysing serious crime groups as social networks. British Journal of Criminology. 2001;**410**(4):580-594. ISSN: 00070955

[10] Papachristos A. Murder by structure: Dominance relations and the social structure of gang homicide. American Journal of Sociology. 2009;**1150**(1):74-128. ISSN: 0002-9602

[11] Papachristos A. The coming of a networked criminology. Measuring Crime and Criminality. 2011;**17**:101-140. ISSN: 08942366

[12] Carrington PJ. Crime and social network analysis. In: Sage Handbook of Social Network Analysis. 2011. pp. 236-255

[13] Haynie DL, Soller B. Network Analysis in Criminology. New York, NY: Springer New York; 2014. pp. 3265-3275. ISBN: 978-1-4614-5690-2

[14] Klimt B, Yang Y. Introducing the enron corpus. In: Machine Learning. stitutep1:0 wwceasccaers2004168. CEAS - First Conference on Email and Anti-Spam, Mountain View, California, USA, July 30-31, 2004

[15] Barbosa D, Filipe N, Gama J. A Social Network Analysis of the Portuguese connection in Panama Papers. FEP Working Papers 592, Universidade do Porto, Faculdade de Economia do Porto; August 2017

[16] Magouirk J, Atran S, Sageman M. Connecting terrorist networks. Studies in Conflict and Terrorism. 2008;**310**(1):1-16. ISSN: 1057610X

[17] Qi X, Duval RD, Christensen K, Fuller E, Spahiu A, Wu Q, Wu Y, Tang W, Zhang C. Terrorist networks, network energy and node removal: A new measure of centrality based on Laplacian energy. Social Networking. 2013;**020**(01):19-31. ISSN: 2169-3285

[18] Gill JC, Malamud BD. Reviewing and Visualizing the Interactions of Natural Hazards; 2014. ISSN: 19449208

[19] Malm A, Nash R, Moghadam R. Social network analysis and terrorism. In: The Handbook of the Criminology of Terrorism. Hoboken, NJ, USA: John Wiley & Sons, Inc.; Jan 2017. pp. 221-231

[20] Berlusconi G. Social network analysis and crime prevention. In: Crime Prevention in the 21st Century. Cham: Springer International Publishing; 2017. pp. 129-141

[21] Fu JS. Leveraging social network analysis for research on journalism in the information age. Journal of Communication. Apr 2016;**660**(2):299-313. ISSN: 00219916

[22] Shumate M, Pilny A, Catouba Y, Kim J, Pena-y Lillo M, Rcooper K, Sahagun A, Yang S. A taxonomy of communication networks. Annals of the International Communication Association. Jan 2013;**370**(1):95-123. ISSN: 2380-8985

[23] Shumate M, Contractor N. Emergence of Multidimensional Social Networks. 3rd ed. Los Angeles: Sage Publishing; 2013. pp. 449-474. ISBN: 9781412987721

[24] Brandenberger L. Trading favors-examining the temporal dynamics of reciprocity in congressional collaborations using relational event models. Social Networks. 2018;**54**: 238-253. ISSN: 0378-8733

[25] Oliveira MDB, Gama J. An overview of social network analysis. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2012;**20**(2):99-115

[26] Tabassum S, Pereira FSF, Fernandes S, Gama J. Social network analysis: An overview. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2018;**00**(0): e1256

[27] Aggarwal C, Subbian K. Evolutionary network analysis: A survey. ACM Computing Surveys (CSUR). 2014;**470**(1):1-36. ISSN: 03600300

[28] Spiliopoulou M. Evolution in social networks: A survey. In: Social Network Data Analytics. Boston, MA: Springer; 2011. pp. 149-175. ISBN: 978-1-4419-8461-6

[29] Thompson WHH, Brantefors P, Fransson P. From static to temporal network theory—applications to functional brain connectivity. bioRxiv. Dec 2016:096461

[30] Kim H, Anderson R. Temporal node centrality in complex networks. Physical Review E. Feb 2012;**850**(2):026107. ISSN: 1539-3755

[31] Casteigts A, Flocchini P, QuattrociocchiW, Santoro N. Time-varying graphs and dynamic networks. In Proceedings of the 10th international conference on Ad-hoc, mobile, and wireless networks (ADHOC-NOW'11). In: Frey H, Li X, Ruehrup S, editors. Berlin, Heidelberg: Springer-Verlag; 2011:346-359

[32] Santoro N, Quattrociocchi W, Flocchini P, Casteigts A, Amblard F. Time-Varying Graphs and Social Network Analysis: Temporal Indicators and Metrics. In 3rd AISB Social Networks and Multiagent Systems Symposium. United Kingdom. 2011:32-38

[33] Guha S, Meyerson A, Mishra N, Motwani R, OCallaghan L. Clustering data streams: Theory and practice. IEEE Transactions on Knowledge and Data Engineering. 2003; **150**(3):515-528. ISSN: 1041-4347

[34] Gehrke J, Korn F, Srivastava D. On computing correlated aggregates over continual data streams. In: Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data-SIGMOD '01. pp. 13-24; 2001. ISBN: 1581133324

[35] Gama J. Knowledge Discovery from Data Streams. 1st ed. Boca Raton, Florida: Chapman & Hall/CRC; 2010. ISBN: 1439826110, 9781439826119

[36] Datar M, Gionis A, Indyk P, Motwani R. Maintaining stream statistics over sliding windows. In: Proceedings of the Thirteenth Annual {ACM-SIAM} Symposium on Discrete Algorithms. pp. 635-644; 2002. ISSN: 0097-5397

[37] Babcock B, Datar M, Motwani R. Sampling from a moving window over streaming data. In: Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '02. pp. 633-634, Philadelphia, PA, USA: Society for Industrial and Applied Mathematics; 2002a. ISBN: 0-89871-513-X

[38] Babcock B, Babu S, Datar M, Motwani R, Widom J. Models and issues in data stream systems. In: Proceedings of the Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS '02. New York, NY, USA: ACM; 2002b. pp. 1-16. ISBN: 1-58113-507-6

[39] Brandes U. A faster algorithm for betweenness centrality. Journal of Mathematical Sociology. 2001;**25**(0):163-177

[40] Nasre M, Pontecorvi M, Ramachandran V. Betweenness centrality-incremental and faster. CoRR. 2013;**abs/1311**:2147

[41] Kas M, Wachs M, Carley KM, Carley LR. Incremental algorithm for updating betweenness centrality in dynamically growing networks. In: 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013). Aug 2013a. pp. 33-40

[42] Kas M, Carley KM, Carley LR. Incremental closeness centrality for dynamically changing social networks. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM '13, New York, NY, USA: ACM, 2013b. pp. 1250-1258. ISBN: 978-1-4503-2240-9

[43] Sariyuce AE, Kaya K, Saule E, Catalyiirek UV. Incremental algorithms for closeness centrality. In: Proceedings-2013 IEEE International Conference on Big Data, Big Data 2013. pp. 487-492; 2013. ISBN: 9781479912926

[44] Ramalingam G, Reps T. An incremental algorithm for a generalization of the shortest-path problem. Journal of Algorithms. September 1996;**210**(2):267-305. ISSN: 0196-6774

[45] Page L, Brin S, Motwani R, Winograd T. The pagerank citation ranking: Bringing order to the web. In: Proceedings of the 7th International World Wide Web Conference. Brisbane, Australia; 1998. pp. 161-172

[46] Katz L. A new status index derived from sociometric analysis. Psychometrika. 1953; **180**(1):39-43. ISSN: 00333123

[47] Society AM. How Google Finds your Needle in the Web's Haystack. http://www.ams.org/publicoutreach/feature-column/fcarc-pagerank; 2007. [Accessed: February, 2018]

[48] Bahmani B, Chowdhury A, Goel A. Fast incremental and personalized PageRank. Proceedings of the VLDB Endowment. December 2010;**40**(3):173-184. Divide and Conquer Approa. ISSN: 2150-8097

[49] Desikan P, Pathak N, Srivastava J, Kumar V. Incremental page rank computation on evolving graphs. In: Special Interest Tracks and Posters of the 14th International Conference on World Wide Web, WWW '05. New York, NY, USA: ACM; 2005. pp. 1094-1095. ISBN: 1-59593-051-5

[50] Kim KS, Choi YS. Incremental iteration method for fast pagerank computation. In: Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication, IMCOM '15. New York, NY, USA: ACM; 2015. pp. 80:1-80:5. ISBN: 978-1-4503-3377-1

[51] Qi X, Fuller E, Wu Q, Wu Y, Zhang C-Q. Laplacian centrality: A new centrality measure for weighted networks. Information Sciences. July 2012;**194**:240-253. ISSN: 0020-0255

[52] Sarmento RP, Cordeiro M, Brazdil P, Gama J. Efficient incremental Laplace centrality algorithm for dynamic networks. In: International Workshop on Complex Networks and their Applications. Cham: Springer; 2017. pp. 341-352

[53] Goh KI, Barabási AL. Burstiness and memory in complex systems. EPL. 2008;**81**(4):48002. https://doi.org/10.1209/0295-5075/81/48002

[54] Holme P, Saramaki J. Temporal Networks. Berlin Heidelberg: Springer; 2013. ISBN: 3642364608

[55] Holme P. Network reachability of real-world contact sequences. Physical Review E-Statistical, Nonlinear, and Soft Matter Physics. 2005;**710**(4):046119 ISSN: 15393755

[56] Nicosia V, Tang J, Mascolo C, Musolesi M, Russo G, Latora V. Graph Metrics for Temporal Networks; 2013. ISSN: 18600832

[57] Tang J, Scellato S, Musolesi M, Mascolo C, Latora V. Small-world behavior in time-varying graphs. Physical Review E. Sep 2009;**81**:055101

[58] Tang J, Musolesi M, Mascolo C, Latora V. Characterising temporal distance and reachability in mobile and online social networks. ACM SIGCOMM Computer Communication Review. Jan 2010;**400**(1):118. ISSN: 01464833

[59] Wasserman S, Faust K. Social Network Analysis: Methods and Applications; 1994. ISSN: 0094-0496

[60] Cui J, Zhang Y-Q, Li X. On the clustering coefficients of temporal networks and epidemic dynamics. In: 2013 {IEEE} International Symposium on Circuits and Systems (ISCAS2013), Beijing, China, May 19–23, 2013; 2013. pp. 2299-2302

[61] Desikan PK, Pathak N, Srivastava J, Kumar V. Divide and Conquer Approach for Efficient Pagerank Computation. Palo Alto, CA: ACM; 2006. ISBN: 1595933522

[62] Sarma AD, Gollapudi S, Panigrahy R. Estimating PageRank on graph streams. Journal of the ACM. 2011;**580**(3):1-19. ISSN: 00045411

[63] Bahmani B, Chakrabarti K, Xin D. Fast personalized PageRank on MapReduce. In: Proceedings of ACM SIGMOD International Conference on Management of Data. 2011. pp. 973-984. ISBN: 9781450306614

[64] Zhang H, Lofgren P, Goel A. Approximate personalized pagerank on dynamic graphs. In: Krishnapuram B, Shah M, Smola AJ, Aggarwal CC, Shen D, Rastogi R, editors, In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA: ACM, August 13-17, 2016. 2016. pp. 1315-1324. ISBN: 978-1-4503-4232-2

[65] Sarmento RP, Cordeiro M, Brazdil P, Gama J. Incremental textrank-automatic keyword extraction for text streams. In: Proceedings of the 20th International Conference on Enterprise Information Systems-Volume 1: ICEIS. INSTICC, SciTePress, 2018. pp. 363-370. ISBN: 978-989-758-298-1

[66] Newman M. Clustering and preferential attachment in growing networks. Physical Review E. 2001;**640**(2):025102

[67] Adamic LA, Adar E. Friends and neighbors on the Web. Social Networks. 2003;**250**(3): 211-230. ISSN: 03788733

[68] Hu K, Xiang J, Yang W, Xu X, Tang Y. Link prediction in complex networks by multi degree preferential-attachment indices. CoRR. 2012;**abs/1211**:1790

[69] Ibrahim NMA, Chen L. Link prediction in dynamic social networks by integrating different types of information. Applied Intelligence. 2015;**420**(4):0 738-0 750

[70] Wahid-Ul-Ashraf A, Budka M, Musial-Gabrys K. Newton's gravitational law for link prediction in social networks. In: COMPLEX NETWORKS, Volume 689 of Studies in Computational Intelligence. Cham: Springer; 2017. pp. 93-104

[71] Choudhury N, Uddin S. Evolutionary community mining for link prediction in dynamic networks. In: Cherifi C, Cherifi H, Karsai M, Musolesi M, editors. Complex Networks & Their Applications. Vol. VI. Cham: Springer International Publishing; 2018. pp. 127-138. ISBN: 978-3-319-72150-7

[72] Fortunato S. Community detection in graphs. Physics Reports. 2010;**4860**(3–5):75-174. ISSN: 03701573

[73] Girvan M, Newman MEJ. Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America. 2002; **990**(12):7821-7826. ISSN: 0027-8424

[74] Newman M, Girvan M. Finding and evaluating community structure in networks. Physical Review E. 2004;**69**(2):1-16. https://doi.org/10.1103/PhysRevE.69.026113

[75] Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment. 2008;**20080**(10). ISSN: 17425468

[76] Shang J, Liu L, Xie F, Chen Z, Miao J, Fang X, Wu C. A real-time detecting algorithm for tracking community structure of dynamic networks. SNAKDD, 18th ACM SIGKDD, 12; 2012

[77] Palla G, Barabasi A-L, Vicsek T. Quantifying social group evolution. Nature. April 2007; **4460**(7136):664-667

[78] Nguyen NP, Dinh TN, Xuan Y, Thai MT. Adaptive algorithms for detecting community structure in dynamic social networks. In: INFOCOM. IEEE. 2011a. pp. 2282-2290

[79] Ye Z, Hu S, Yu J. Adaptive clustering algorithm for community detection in complex networks. Physical Review E. Oct 2008;**78**:046115

[80] Nguyen NP, Dinh TN, Tokala S, Thai MT. Overlapping communities in dynamic networks: Their detection and mobile applications. In: Ramanathan P, Nandagopal T, Levine BN, editors, Proceedings of the 17th Annual International Conference on Mobile Computing and Networking, MOBICOM 2011, Las Vegas, Nevada, USA: ACM, September 19-23, 2011. pp. 85-96; 2011b. ISBN: 978-1-4503-0492-4

[81] Xie J, Szymanski BK, Liu X. SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process. In: IEEE International Conference on Data Mining, ICDM. 2011. pp. 344-349

[82] Xie J, Szymanski BK. Towards linear time overlapping community detection in social networks. Lecture Notes in Computer Science, LNAI. 2012;**7301**:25-36

[83] Xie J, Chen M, Szymanski BK. LabelRankT: Incremental community detection in dynamic networks via label propagation. In Proceedings of the Workshop on Dynamic Networks Management and Mining (DyNetMM '13). New York, USA: ACM; 2013:25-32

[84] Xie J, Kelley S, Szymanski BK. Overlapping community detection in networks: The state-of-the-art and comparative study. ACM Computing Surveys (CSUR). 2013b;**45**. ISSN: 0360–0300

[85] Cordeiro M, Sarmento RP, Gama J. Dynamic community detection in evolving networks using locality modularity optimization. Social Network Analysis and Mining. 2016;**60**(1): 1-20

[86] Wang CD, Lai JH, Yu PS. Dynamic community detection in weighted graph streams. In: Proceedings of the 2013 SIAM International Conference on Data Mining. 2013. pp. 151-161

[87] Raghavan UN, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks. Physical Review E. 2007;**760**(3):036106. ISSN: 15393755

[88] Leung IXY, Hui P, Liò P, Crowcroft J. Towards real-time community detection in large networks. Physical Review E-Statistical, Nonlinear, and Soft Matter Physics. 2009;**790**(6): 1-10. ISSN: 15393755

[89] Yun S, Lelarge M, Proutière A. Streaming, memory limited algorithms for community detection. CoRR. 2014;**abs/1411**:1279

[90] Yun S, Proutière A. Community detection via random and adaptive sampling. CoRR. 2014;**abs/1402**:3072

[91] Sariyüce AE, Gedik B, Jacques-Silva G, Wu K, Çatalyürek ÜV. SONIC: Streaming overlapping community detection. Data Mining and Knowledge Discovery. 2016;**300**(4): 819-847

[92] Hollocou A, Maudet J, Bonald T, Lelarge M. A linear streaming algorithm for community detection in very large networks. CoRR. 2017;**abs/1703**:02955

[93] Aggarwal CC. Social Network Data Analytics. 1st ed. New York, NY, USA: Springer Publishing Company, Incorporated; 2011. ISBN: 1441984615, 9781441984616

[94] Zaidi F, Muelder C, Sallaberry A. Analysis and visualization of dynamic networks, CoRR. 2014;**abs/1409**:5034

[95] Beck F, Burch M, Diehl S, Weiskopf D. The state of the art in visualizing dynamic graphs. In: Proceedings State of the Art Reports (STARs). 2014. pp. 83-10. ISSN: 19464606

[96] Beck F, Burch M, Diehl S, Weiskopf D. A taxonomy and survey of dynamic graph visualization. Computer Graphics Forum. 2017;**360**(1):133-159. ISSN: 14678659

[97] Moody J, McFarland D, Bender-deMoll S. Dynamic network visualization. American Journal of Sociology. 2005;**1100**(4):1206-1241. ISSN: 0002-9602

[98] Brandes U, Corman SR. Visual unrolling of network evolution and the analysis of dynamic discourse. Information Visualization. 2003;**20**(1):40-50. ISSN: 14738724

[99] Chen C, Morris S. Visualizing evolving networks: Minimum spanning trees versus pathfinder networks. In: Proceedings - IEEE Symposium on Information Visualization, INFO VIS. 2003. pp. 67-74. ISBN: 0780381548

[100] Bender-deMoll S, McFarland D a. The art and science of dynamic network visualization. Journal of Social Structure. 2006;**70**(2):1-38. ISSN: 14620324

[101] Falkowski T, Bartelheimer J, Spiliopoulou M. Mining and visualizing the evolution of subgroups in social networks. In: Proceedings-2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings), WI'06. 2007. pp. 52-58. ISBN: 0769527477

[102] Chen C. CiteSpace II: Detecting and visualizing emerging trends and transient patterns in scientific literature. Journal of the American Society for Information Science and Technology. 2006;**570**(3):359-377. ISSN: 15322882

[103] Beck F, Burch M, Diehl S. Towards an aesthetic dimensions framework for dynamic graph visualisations. In: Proceedings of the International Conference on Information Visualisation; 2009. pp. 592–597. ISBN: 978-0-7695-3733-7

[104] Beck F, Burch M, Diehl S. Matching application requirements with dynamic graph visualization profiles. In: Proceedings of the International Conference on Information Visualisation; 2013. pp. 11-18. ISBN: 9780769550497

[105] Burch M. The dynamic graph wall: Visualizing evolving graphs with multiple visual metaphors. Journal of Visualization. 2017;**200**(3):461-469. ISSN: 18758975

[106] Hu P, Lau W. A survey and taxonomy of graph sampling. arXiv.org. 2013;**abs/1308.5865**: 1-34

[107] Cordeiro M, Gama J. Online Social Networks Event Detection: A Survey. Cham: Springer International Publishing; 2016. pp. 1-41. ISBN: 978-3-319-41706-6

[108] Cordeiro MMF. Event detection: Monitoring and tracking the dynamics of social networks communities. In: 2016 17th IEEE International Conference on Mobile Data Management (MDM). Vol 2. June 2016a. pp. 18-19

[109] Cordeiro MMF. Mining the twitter stream: Unravel events, interactions, and communities in real-time. In: 2016 17th IEEE International Conference on Mobile Data Management (MDM), Vol 2. June 2016b. pp. 49-54

[110] Stray J. Network Analysis in Journalism-KDD Data Science+Journalism Workshop 2017. https://drive.google.com/file/d/0B3Rg-4kycp0Qczl0MkJYVXlNVFU/view, 2017. [Accessed: April, 2018]

[111] Romano T. In seattle art world, women run the show-seattle times. https://www.seattletimes.com/entertainment/visual-arts/in-seattle-art-world-women-run-the-show/; 2017. [Accessed: April, 2018]

[112] Sampson ZT, Gartner L, Shadd D. Hot Wheels-Tampa Bay Times. http://www.tampabay.com/projects/2017/investigations/florida-pinellas-auto-theft-kids-hot-wheels/; 2017. [Accessed: April, 2018]

[113] Heymann S. Panama papers: How Linkurious Enables Icij to Investigate the Massive Mossack Fonseca Leaks. https://linkurio.us/blog/panama-papers-how-linkurious-enables-icij-to-investigate-the-massive-mossack-fonseca-leaks/; 2017. [Accessed: April, 2018]