WILEY WIREs DATA MINING AND KNOWLEDGE DISCOVERY

**OVERVIEW**

# Social network analysis: An overview

Shazia Tabassum[1,2] | Fabiola S. F. Pereira[3] | Sofia Fernandes[1,2] | João Gama[1,2]

[1]INESC TEC, Rua Dr. Roberto Frias, Porto, Portugal

[2]University of Porto, Porto, Portugal

[3]Federal University of Uberlandia, Uberlandia, Brazil

**Correspondence**
João Gama, INESC TEC, Porto, Portugal.
Email: jgama@fep.up.pt

Social network analysis (SNA) is a core pursuit of analyzing social networks today. In addition to the usual statistical techniques of data analysis, these networks are investigated using SNA measures. It helps in understanding the dependencies between social entities in the data, characterizing their behaviors and their effect on the network as a whole and over time. Therefore, this article attempts to provide a succinct overview of SNA in diverse topological networks (static, temporal, and evolving networks) and perspective (ego-networks). As one of the primary applicability of SNA is in networked data mining, we provide a brief overview of network mining models as well; by this, we present the readers with a concise guided tour from analysis to mining of networks.

This article is categorized under:

Application Areas > Science and Technology
Technologies > Machine Learning
Fundamental Concepts of Data and Knowledge > Human Centricity and User Interaction
Commercial, Legal, and Ethical Issues > Social Considerations

**KEYWORDS**

evolving networks, social network analysis, temporal networks

## 1 | INTRODUCTION

Nowadays the data generated from many of the real world applications are represented as a network of interconnected objects. The main objective is to extract more information than the traditional way of investigating independent objects. Of course, it increases the complexity of handling data as well. One of the major class of data networks is social networks. A social network can be constructed from relational data and can be defined as a set of social entities, such as people, groups, and organizations, with some relationships or interactions between them. These networks are usually modeled by graphs, where vertices represent the social entities and edges represent the ties established between them.

Some of the common examples of social networks are given in Table 1. The underlying structure of such networks is the object of study of social network analysis (SNA). SNA methods and techniques were thus designed to discover patterns of interaction between social actors in social networks. Hence, the focus of SNA is on the relationships established between social entities rather than the social entities themselves. In fact, the main goal of this technique is to examine both the contents and patterns of relationships in social networks in order to understand the relations among actors and the implications of these relationships.

A network is defined by the relation/link between its nodes as given in the examples above. An example network graph is given in Figure 1, with different colors shown communities (refer section 5 to read about communities). There can be distinct relations between a single set of nodes in a network. For example, in a product network, the relation could be based on "similarity" or "brought together" in a set of products. Similarly, there can be unique/distinct relations between multiple sets of nodes, for example user–product networks. These type of networks are heterogeneous networks. When the network comprises of two sets of nodes, it is called a two-mode network. Some examples of two-mode networks include user–product

**TABLE 1** Some examples of social networks

| Examples | Applications |
| --- | --- |
| Friendship networks | College/school students, organizations or web (Facebook, MySpace, etc.) |
| Follower networks | Twitter, LinkedIn, Pinterest, etc. |
| Preference similarity networks | Pinterest, Instagram, Twitter, etc. |
| Interaction networks | Phone calls, Messages, Emails, Whatsapp, Snapchat, etc. |
| Co-authorship networks | Dblp, Science direct, Wikibooks, other scientific databases, etc. |
| User–user citation networks | Dblp, Science direct, Wikibooks, other scientific databases, etc. |
| Spread networks | Epidemics, Information, Rumors, etc. |
| Co-actor networks | IMDB, etc. |

networks (Amazon, eBay, etc.), membership or affiliation networks (actor–movies (IMDB), user–group (youtube), user–channel (youtube), user–project (GitHub), user–organization, etc.), user–preference networks (Pinterest, Instagram, Twitter), citation networks, user–stock investment. These two-mode networks can be transformed into single-mode networks between a single set of nodes like the examples given above and then analyzed. However, two-mode networks can also be analyzed using methods discussed by Borgatti and Everett (1997) and Latapy, Magnien, and Del Vecchio (2008).

Apart from social networks, numerous data networks are also formed between objects other than social entities, like sensors, products, words/texts, brain neurons, proteins, genes, geographical locations, predators and preys, and web-pages, and so on. Though the SNA measures were primarily designed to analyze social networks, they can also be employed to analyze data networks like these.

Common tasks of SNA involve the identification of the most influential, prestigious or central actors, using statistical measures; the identification of hubs and authorities, using link analysis algorithms; discovering communities, using community detection techniques, and how information propagates in the network, using diffusion algorithms. These tasks are extremely useful in the process of extracting knowledge from networks and, consequently, in the process of problem solving. Due to the appealing nature of such tasks and to the high potential opened by this kind of analyses, SNA has become a popular approach in a myriad of fields, from biology to business. For instance, some companies use SNA in order to maximize positive word-of-mouth of their products by targeting the customers with higher network value (those with higher influence and support) (Domingos & Richardson, 2001; Leskovec, Adamic, & Huberman, 2007; Richardson & Domingos, 2002). Other companies, such as the ones operating in the sector of mobile telecommunications, apply SNA techniques to the phone
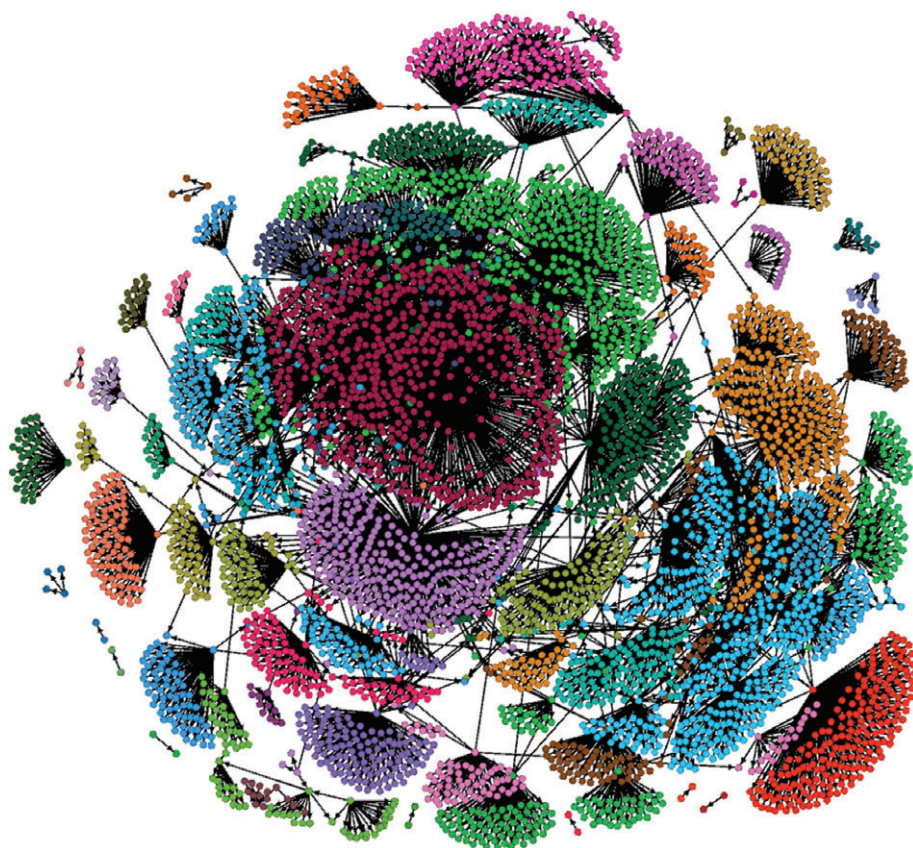


**FIGURE 1** An example network representation

call networks and use them to identify customer's profiles and to recommend personalized mobile phone tariffs, according to these profiles. These companies also use SNA for churn prediction, that is, to detect customers who may potentially switch to another mobile operator by detecting changes in the patterns of phone contacts (Dasgupta et al., 2008; Wei & Chiu, 2002). Another interesting application emerges from the domain of fraud detection. For instance, SNA can be applied to networks of organizational communications (e.g., Enron company dataset) in order to perform an analysis of the frequency and direction of formal/informal email communication, which can reveal communication patterns among employees and managers. These patterns can help identify people engaged in fraudulent activities, thus promoting the adoption of more efficient forms of acting toward the eradication of crime (Shetty & Adibi, 2004; Xu & Chen, 2005).

Despite the fact the origins of network studies go back a few centuries in recent years we witnessed an impressive advance in network-related fields, mainly due to the growing interest in social networks (Abraham, Hassanien, & Sná, 2009; Aggarwal, 2009; Barabási, 2016; Charu & Aggarwal, 2011; Furht, 2010; Wasserman & Faust, 1994; Zafarani, Abbasi, & Liu, 2014), which became a "hot" topic and a focus of considerable attention. For this reason, a lot of students, practitioners, and researchers are willing to enter the field and explore, even superficially, the potential of SNA techniques for the study of their problems. Bearing this in mind, in this paper our aim is to provide a general and succinct overview of the essentials of SNA for those interested in knowing more about the area and strongly oriented to use SNA in practical problems and different classes of networks.

The document is organized as follows. We begin by pointing out some types of representations that can be used to model social networks. Then, we introduce the best known statistical measures to analyze them with a different perspective of networks: the entire social network and ego-network. Afterward, we talk a little about the link analysis task and explain how it can be used to identify influential and authoritative nodes. Then, we discuss the task of link prediction referring to the state of the art techniques. Later, we devote a section to the problem of finding communities in networks. After introducing the main concepts, we provide a succinct overview of evolving network analysis and its applicability. Further, we present a brief illustration on representing the temporal property of evolving networks. Note that the evolving networks are temporal but the temporal networks need not always be evolving. Finally, this overview ends with the identification of the current trends arising in the field of SNA.

## 2 | ANALYZING SOCIAL NETWORKS

### 2.1 | Representation of social networks

A social network consists of a finite set of vertices and the relations, or ties, defined on them (Wasserman & Faust, 1994). The established relationships can be of personal, or professional, nature and can range from casual acquaintance to close familiar bonds. Besides social relations, links can also represent the flow of information/goods/money, interactions, similarities, among others. The structure of such networks is usually represented by graphs. Therefore, networks are often regarded as equivalent to graphs.

A graph is composed of two fundamental units: vertices and edges. Every edge is defined by a pair of vertices, also called its endpoints. Vertices are able to represent a wide variety of individual entities (e.g., people, organizations, countries, papers, products, plants, and animals) according to the application field. In turn, an edge is a line that connects two vertices and, analogously, it can represent numerous kinds of relationships between individual entities (e.g., communication, cooperation, friendship, kinship, acquaintances, and trade). Edges may be directed or undirected, depending if the nature of the relation is asymmetric or symmetric. Formally, a graph $G$ consists of a nonempty set $V(G)$ of vertices and a set $E(G)$ of edges, being defined as $G = (V(G), E(G))$. According to Diestel (1990), the order of a graph $G$ is given by the total number of vertices $n$ or, mathematically, $|V(G)| = n$. Analogously, the size of a graph $G$ is the total number of edges $|E(G)| = m$. The maximum number of edges in a graph is $m_{\max} = \frac{n(n-1)}{2}$, for undirected graphs, and $m_{\max} = n(n-1)$, for directed ones.

In the literature, two main types of graph-theoretic data structures are referred to represent graphs: the first one is list structures and the second is matrix structures. These structures are appropriate to store graphs in computers in order to further analyze them using automatic tools. List structures, such as incidence lists and adjacency lists, are suitable for storing sparse graphs since they reduce the required storage space. On the other hand, matrix structures such as incidence matrices, adjacency matrices or sociomatrices, Laplacian matrices (contains both adjacency and degree information) and distance matrices (identical to the adjacency matrices with the difference that the entries of the matrix are the lengths of the shortest paths between pairs of vertices) are appropriate to represent full matrices. Several types of graphs can be used to model different kinds of social networks. For instance, graphs can be classified according to the direction of their links. This leads us to the differentiation between undirected and directed graphs. Undirected graphs (or undirected networks) are graphs whose edges connect unordered pairs of vertices or, in other words, each edge of the graph connects concomitantly two vertices. A stricter
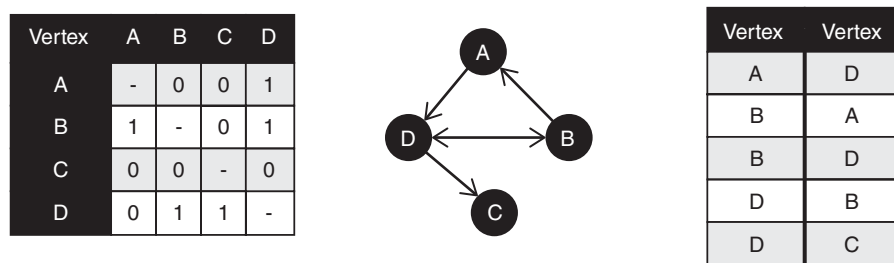
| Vertex | A | B | C | D |
|--------|---|---|---|---|
| A | - | 0 | 0 | 1 |
| B | 1 | - | 0 | 1 |
| C | 0 | 0 | - | 0 |
| D | 0 | 1 | 1 | - |

| Vertex | Vertex |
|--------|--------|
| A | D |
| B | A |
| B | D |
| D | B |
| D | C |

**FIGURE 2** A directed and unweighted graph $G$ represented by means of an adjacency matrix (left-side of the figure) and an adjacency list (right-side of the figure)

type of graph is the so-called directed graph (or directed network). Directed graphs, or in the abbreviation form digraphs, can be straightforwardly defined as graphs whose all edges have an orientation assigned (also called arcs), so the order of the vertices they link matters. Formally, in a directed graph 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 $v_1$ to $v_2$, being the first vertex $v_1$ called initial vertex, or tail, and the second vertex $v_2$ called the terminal vertex, or simply head. Graphically, directed edges are depicted by arrows, indicating the direction of the linkage. This type of graphs can be either cyclic, that is, graphs containing closed loops of edges or "ring" structures, or acyclic (e.g., trees). A typical example of an undirected graph is Facebook™ since, in this social network, the established friendship tie is mutual, or reciprocal (e.g., if I accept a friend request from a given person then it is implicitly assumed that I and that person are friends of each other). Likewise, Twitter™ is an example of a directed graph since a person can be followed by others without necessarily follows them. In this case, the tie between a pair of individuals is directed, with the tail being the follower and the head being the followed, meaning that a one-way relationship is established.

Regarding the values assigned to edges, we can make a distinction between unweighted and weighted graphs. Unless it is explicitly said, we always assume that graphs are unweighted. Unweighted graphs are binary since edges are either present or absent. On the other hand, weighted graphs are richer graphs since each edge has associated a weight $w \in \mathbb{R}_0^+$ providing the user with more information about, for instance, the strength of the connection of the pair of vertices it joins. According to Granovetter (1973, 1995) in social networks the weight of a tie is generally a function of duration, emotional intensity, the frequency of interaction, intimacy, and exchange of services. Therefore, strong ties usually represent close friends, and weak ties represent acquaintances. In other kinds of networks, the weight of a tie can represent a variety of things, depending on the context; for instance, a tie can represent the number of seats among airports, the number of exchanged products, and so on. For undirected and unweighted graphs, adjacency matrices are binary (as a consequence of being unweighted) and symmetric (as a consequence of being undirected, meaning that $a_{ij} = a_{ji}$), with $a_{ij} = 1$ representing the presence of an edge between vertices $i$ and $j$, and $a_{ij} = 0$ representing the absence of an edge between vertex pair $(i, j)$. For directed and weighted graphs the entries of such matrices take values from interval $[0, \max(w)]$ and are nonsymmetric. In both cases, we deal with non-negative matrices. In Figure 2, we provide an example of how a graph can be represented by an edge list and by an adjacency matrix.

## 2.2 | Node-level statistical measures

In this section, we present some graph measures and popular metrics used in the analysis of social networks. These measures are useful in the sense that they provide us insights about the role of nodes in the network. Studying the role of individuals and how they interact in the network context aims at understanding the behavior of the social systems that generated those networks, which is normally the final goal of such analysis. The measures we will introduce in the following sections can be divided according to the level of analysis one wants to perform: at the level of small units, such as nodes, or at the level of the whole network. The former explores general measures of centrality as a way to understand how the position of a vertex is within the overall structure of the graph and, therefore, helps identify the key players in the network. The latter provides more compact information and allows the assessment of the overall structure of the network, giving insights about important properties of the underlying social phenomena.

Centrality or prestige is a general measure of how the position of an actor is within the overall structure of the social network and can be computed resorting to several metrics. The most widely used are degree, betweenness, closeness, and eigenvector centrality. The first three were proposed by Freeman (1978) and were only designed for unweighted networks. Recently, Brin and Page (2012) came up with extensions to weighted networks. The fourth metric—eigenvector centrality—was later proposed by Bonacich (1987) and has its foundations on spectral graph theory. It became especially popular after being used as the basis of the well-known Google's PageRank algorithm, which we will talk about in the next section. Although more actor-level statistical measures were proposed in the literature, in this section we will focus on explaining the mentioned measures of centrality. These measures determine the relative importance of an actor within the network, showing how the relationships are concentrated in a few individuals and, therefore, giving an idea about their social power. Higher

centrality measures are associated to powerful actors in the network, since their central position offers them several advantages, such as easier and quicker access to other actors in the network (useful for accessing resources such as information) and the ability to exert control over the flow between the other actors (Freeman, 1978). These central actors are also called "focal points." At the end of the section, we will also introduce the concept of transitivity and explain how it can be computed using a clustering coefficient.

The reader must take into account that some of these actor-level metrics (e.g., degree, betweenness, and closeness) may need to be normalized in order to perform comparisons of networks with different orders and sizes.

### 2.2.1 | Degree or valency

The degree, or valency, of a node $v$, usually denoted as $k_v$, is a measure of the immediate adjacency and the involvement of the node in the network and is computed as the number of edges incident on a given node or, similarly, as the number of neighbors of node $v$. The neighborhood $N_v$ is thus defined by the set of nodes that are directly connected to $v$. Degree can be computed in, at least, two different ways: based on the adjacency matrix or based on the neighborhood of a node. In Equations (1) and (2), we present each one of the alternatives, for undirected networks:

$$k_i = \sum_{j=1}^{n} a_{ij}, \qquad 0 < k_i < n, \tag{1}$$

where $a_{ij}$ is the entry of the $i$th row and $j$th column of the adjacency matrix

$$k_v = |N_v|, \qquad 0 < k_v < n, \tag{2}$$

where $|N_v|$ is the neighborhood of node $v$.

Despite its simplicity, degree is an effective measure to assess the importance and influence of an actor in a social network. Yet, it has some limitations. The main one is that it does not take into consideration the global structure of the network. For directed networks, there are two variants of degree centrality: in-degree, denoted by $k_v^+$, and out-degree, denoted by $k_v^-$. The former is given by the number of incoming nodes (i.e., number of edges ending at vertex $v$) and the later by the number of outgoing nodes, (i.e., number of edges beginning at vertex $v$), as defined in Equation (3). The measure of degree in directed networks is also referred to as prestige. This expression is especially used in the literature of social networks, since it was developed for measuring the prominence or importance of actors in the network. There are two types of prestige: support and influence. The first is related to the in-degree centrality, which is seen as a measure of support, and the second is related to the out-degree centrality, which is seen as a measure of influence

$$k_i^+ = \sum_{j=1}^{n} a_{ji}, \qquad k_i^- = \sum_{j=1}^{n} a_{ij}. \tag{3}$$

On weighted networks, strength is the equivalent of degree, being computed as the sum of the weights of the edges adjacent to a given node, as expressed by Equation (4):

$$k_i^w = \sum_{j=1}^{n} a_{ji}^w. \tag{4}$$

A significant research effort was undertaken in studying the degree distribution of several types of networks, which turned it possible to classify a network based on this distribution. For instance, Barabási and Albert (1999) and Barabási and Bonabeau (2003) discovered that most real networks follows a power-law distribution, at least asymptotically. This means that, in these networks, the distribution of the vertex degree is very heterogeneous and highly right-skewed, with a large majority of vertices having a low degree and a small number having a high degree. These networks are known as scale-free, an expression coined by the same researchers. Other common functional forms are exponential (e.g., railways and power grids networks) and power-laws with exponential cut-offs (e.g., networks of movie actors and some collaboration networks).

### 2.2.2 | Betweenness

Node betweenness $b$ measures the extent to which a node lies between other nodes in the network and can be computed as the percentage of shortest paths that pass through the node. The formula is presented in Equation (5). Nodes with high betweenness occupy critical roles in the network structure, since they usually have a network position that allow them to work as an interface between tightly-knit groups, being "vital" elements in the connection between different regions of the network. From the social networks perspective "interactions between two nonadjacent actors might depend on other actors in the set of actors, especially the actors who lies on the paths between the two" (Wasserman & Faust, 1994), which stresses

out the importance of a good value of betweenness. These actors are also called gatekeepers since they tend to control the flow of information between communities

$$b_v = \sum_{s,t \in V(G) \backslash v} \frac{\sigma_{st}(v)}{\sigma_{st}}, \tag{5}$$

where $\sigma_{st}$ denotes the number of shortest paths between vertices $s$ and $t$ (usually $\sigma_{st} = 1$) and $\sigma_{st}(v)$ expresses the number of shortest paths passing through node $v$. This quantity can also be computed for edges. The betweenness of an edge $b_e$ is commonly defined as the number of shortest paths between nodes that run along a given edge of the network. It is quite useful in SNA since it allows discovering bridges and local bridges which are, by definition, edges with high betweenness. In the context of SNA, bridges are connections outside an individual's circle of acquaintances. These connections are of great interest for individuals seeking to access new information and resources, since they ease the diffusion of information across entire communities (Kossinets & Watts, 2006). However, situations like these are quite rare in real-world scenarios and, even if they happen, the advantages they confer are usually temporary, due to the temporal instability of such edges. A more common and realistic situation are local bridges. Equation (6) indicates how this measure can be computed:

$$b_e = \sum_{u,v \in V(G)} \frac{\sigma_{uv}(e)}{\sigma_{uv}}, \tag{6}$$

where $\sigma_{uv}(e)$ expresses the number of shortest paths passing through edge $e$. The sum indicates that this fraction needs to be computed for every pair of nodes $u$ and $v$ in the network.

### 2.2.3 | Closeness

Closeness $Cl_v$ is a rough measure of the overall position of an actor in the network, giving an idea about how long it will take to reach other nodes from a given starting node. Formally, it is the mean length of all shortest paths from one node to all other nodes in the network. Due to its definition usually this measure is only computed for nodes within the largest component (for the definition of components refer section 3.2) of the network, using the formula presented in Equation (7). In the social networks context, closeness is a measure of reachability that measures how fast a given actor can reach everyone in the network

$$Cl_v = \frac{n-1}{\sum_{u \in V(G) \backslash v} d(u,v)}. \tag{7}$$

### 2.2.4 | Eigenvector centrality

This metric is based on the assignment of a relative score to each node and measures how well a given actor is connected to other well-connected actors. This score is given by the first eigenvector of the adjacency matrix. The basic idea behind eigenvector centrality is that the power and status of an actor are recursively defined by the power and status of his/her alters. Alters is a term frequently used in the egocentric approach of social networks analysis, and it refers to the actors that are directly connected to a specific actor, called ego. In other words, we can say that the centrality of a given node $i$ is proportional to the sum of the centralities of $i$'s neighbors. This is the assumption behind the eigenvector centrality formula, which is as follows:

$$x_i \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij} x_j, \tag{8}$$

where $x_i/x_j$ denotes the centrality of node $i/j$, $a_{ij}$ represents an entry of the adjacency matrix $A$ (($a_{ij}$) = 1 if nodes $i$ and $j$ are connected by an edge and ($a_{ij}$) = 0 otherwise) and $\lambda$ denotes the largest eigenvalue of $A$.

Eigenvector centrality is a more elaborated version of the degree, once it assumes that not all connections have the same importance by taking into account not only the quantity, but especially the quality of these connections.

### 2.2.5 | Local clustering coefficient

Social networks are naturally transitive, which means that a given actor's friends are also likely to be friends. This property of transitivity is quantified by a clustering coefficient that can be global, that is, computed for the whole network, or local, that is, computed for each node. Watts and Strogatz (1998) proposed a local version of the clustering coefficient, denoted $c_i$, $i = (1, \ldots, n)$. In this context, transitivity is a local property of a node's neighborhood that indicates the level of cohesion

between the neighbors of a node. This coefficient is, therefore, given by the fraction of pairs of nodes, which are neighbors of a given node that are connected to each other by edges:

$$C_i = \frac{2\,|\,e_{jk}\,|}{k_i(k_i-1)} : v_j, \quad v_k \in N_i, \quad e_{jk} \in E \tag{9}$$

where $N_i$ is the neighborhood of node $v_i$, $e_{jk}$ represents the edge that connects node $v_j$ to node $v_k$, $k_i$ is the degree of node $v_i$, and $|e_{jk}|$ indicates the proportion of links between the nodes within the neighborhood of node $v_i$.

## 2.3 | Network-level statistical measures

Before explaining each one of the network-level statistical measures, there are three fundamental concepts that should be first introduced: path, geodesic distance between two nodes and eccentricity of a vertex.

A path is a sequence of nodes in which consecutive pairs of nonrepeating nodes are linked by an edge; the first vertex of a path is called the start vertex and the last vertex of the path is called the end vertex. Of particular interest is the concept of geodesic distance, or shortest path, between nodes $i$ and $j$, denoted as $d(i, j)$. The geodesic distance can be defined as the length of the shortest path, or the minimal path, between nodes $i$ and $j$.

In turn, the eccentricity is the greatest geodesic distance between a given vertex and any other in the graph, as defined in Equation (10). These three concepts are on the basis of most of the network-level metrics we are going to introduce, namely, the diameter/radius, the average geodesic distance, the average degree, the reciprocity, the density and the global clustering coefficient

$$\epsilon = \max_{i \in V(G) \backslash v} d(v,i). \tag{10}$$

### 2.3.1 | Diameter and radius

The diameter $D$ is given by the maximum eccentricity of the set of vertices in the network and, analogously, the radius $R$ can be defined as the minimum eccentricity of the set of vertices, as defined in Equation (11). Sparser networks have generally greater diameter than full matrices, due to the existence of fewer paths between pairs of nodes. Leskovec, Kleinberg, and Faloutsos (2005) discovered that, for certain types of real-world networks, the effective diameter shrinks over time, contradicting the conventional wisdom of increasing diameters. In the context of SNA, this metric gives an idea about the proximity of pairs of actors in the network, indicating how far two nodes are, in the worst of cases

$$D = \max\{\epsilon : v \in V\}, \quad R = \min\{\epsilon : v \in V\}. \tag{11}$$

### 2.3.2 | Average geodesic distance

The average geodesic distance for all combinations of vertex pairs in a network is usually denoted by $l$ and is given by Equation below:

$$l = \frac{1}{\frac{1}{2}n(n-1)} \sum_{i \geq j} d(i,j), \tag{12}$$

where $d(i, j)$ is the geodesic distance between nodes $i$ and $j$, and $\frac{1}{2}n(n-1)$ is the number of possible edges in a network comprising $n$ nodes. This metric gives an idea of how far apart nodes will be, on average. For instance, in the SNA context the average geodesic distance can be used to measure the efficiency of the information flow within the network.

When there is the case of a network having more than one connected component, the previous formula does not hold, since the geodesic distance is conventionally defined as infinite when there is no path connecting two vertices. In such situations, it is more appropriate to use the harmonic average geodesic distance, defined in Equation (13), once it turns infinite distances into zero nullifying their effect on the sum:

$$l^{-1} = \frac{1}{\frac{1}{2}n(n+1)} \sum_{i \geq j} d(i,j). \tag{13}$$

### 2.3.3 | Average degree

The average degree is simply the mean of the degrees of all vertices in a network, as represented in Equation (14). According to Costa et al. (2011) the average degree can be used to measure the global connectivity of a network

$$\overline{k} = \frac{1}{n}\sum_{i=1}^{n} k_i. \tag{14}$$

### 2.3.4 | Reciprocity

Reciprocity $r$ is a specific quantity for directed networks that measures the tendency of pairs of nodes to form mutual connections between each other. There are several ways to compute this metric. The most popular and intuitive way is to compute the ratio of the number of mutual connections in the network to the number of all connections, as shown in Equation (15):

$$r = \frac{\#mut}{\#mut + \#asym}, \qquad 0 < r < 1, \tag{15}$$

where *#mut* denotes the number of mutual dyads and *#asym* the number of asymmetric dyads. Adopting this definition, the value of reciprocity represents the probability that two nodes in a directed network point to each other. By definition, in an undirected network, reciprocity is always maximum $r = 1$, since all pairs of nodes are symmetric. Taking the definitions of Wasserman and Faust (1994), we say that an asymmetric dyad is a pair of nodes that has an arc going in the direction of one node or the other, but not both directions. In turn, a mutual dyad is defined by a pair of nodes connected by two arcs, each one going in a different direction (e.g., $a \to b$ and $b \to a$, being $a$ and $b$ two nodes in a network).

### 2.3.5 | Density

Density $\rho$ is an important network-level measure, which is able to explain the general level of connectedness in a network. It is given by the proportion of edges in the network relative to the maximum possible number of edges, as defined in Equation (16):

$$\rho = \frac{m(G)}{m_{\max(G)}}, \qquad 0 < \rho < 1, \tag{16}$$

where $m(G)$ is the number of edges in the network and $m_{\max(G)}$ denotes the number of possible edges, which is $\frac{n(n-1)}{2}$ for undirected networks and $n(n-1)$ for directed ones. Density is a quantity that goes from a minimum of 0, when a network has no edges at all, to a maximum of 1, when the network is perfectly connected (also called complete graph or clique). Therefore, high values of are associated to dense networks, and low values of density are associated to sparse networks.

### 2.3.6 | Global clustering coefficient

There are several ways to compute the global version of the clustering coefficient. We adopt the one proposed by Watts and Strogatz (1998), that obtains the global clustering coefficient $c$, for the whole network, through the computation of the average of all local values $c_i$ ($i = 1, \ldots, n$), as shown in Equation (17). Small-world networks (Watts & Strogatz, 1998), such as the ones we find in real social contexts, are characterized by high global clustering coefficients, meaning that the property of transitivity among nodes emerges more often and in a stronger way, increasing the probability of clique formation

$$c = \frac{1}{n}\sum_{i} c_i. \tag{17}$$

## 3 | EGO NETWORKS

An ego network is a local network of a particular node. An ego represents a focal node from the network and all the other nodes in the ego network connected to it are called alters. An ego centric network maps the relationships of an ego with alters and also between themselves. An ego network can be of level/radius $l = 1$ (i.e., comprising of nodes only adjacent to the ego), $l = 2$ (adding nodes adjacent to the nodes at $l = 1$), ..., $l = D$ (diameter of the graph) gives the whole network being considered. The levels generally studied in the ego network analysis are 1 and/or 2. The alters with direct connections to the ego are called primary alters, while the alters adjacent to primary alters are called secondary alters and so on. Wellman (1996) describes an ego network as a personal network. The studies made by Everett and Borgatti (2005) indicate that the local ego betweenness is highly correlated with the betweenness of the actor in the complete network.

## 3.1 | Ego network analysis

Network analysis, from the viewpoint of egos, has attracted much attention over the last decade. Some of the predominant reasons include scalability, because of the exponentially growing data it has been difficult to analyze the whole network en masse. Analyzing ego networks not only gives insights into the whole network but also can be exploited extensively with memory constraints. The growth pattern of ego networks highly influences the growth of social networks (Tabassum & Gama, 2016a). Epasto et al. (2015) argue that it is possible to address important graph mining tasks by analyzing the ego-nets of a social network and performing independent computations on them. Akcora and Ferrari (2014) show how social trust can be measured from user's ego network connections. Secondly, in the prevalent trend of user profile building and personalization in applications (Liu, Wang, Zhang, & Yin, 2018), preferences (Pereira, Tabassum, Amo, & Gama, 2018), recommendations (Sun & Zhu, 2013), and services (Wang et al., 2008), ego network analysis and mining is quite pertinent. It is applicable in many other realms including IOT to tackle tremendous data generated from personal interactions. Also serves in studying the structural behaviors of influential, powerful, or controlling nodes, and so on. Though there are numerous applications and advantages of ego network analysis and mining, the research work in this area is still in its infancy, mainly because of the restrictive structure of ego networks. Nevertheless, below we discuss some ego-based network level measures that can be used for ego network analysis besides the conventional statistical measures and the SNA measures discussed above. Though they are network level measures, they are designed to particularly address the properties/characteristics of an ego (node level) in its personal network but not the alters'. Some of the metrics briefed below (effective size, efficiency, and constraint) were introduced by Burt (2009) to analyze personal networks. To get a more detailed explanation of those metrics the readers can also go through (Hanneman & Riddle, 2005). The practitioners of R (Ihaka & Gentleman, 1996) can use egonets package for implementing these measures.

Before we discuss the measures below, the readers need to understand the concepts of redundancy and structural holes. When more than one path exists between two nodes in a network it is said to be redundant. A structural hole is a separation/link between non redundant contacts. It can also be a bridge between two nodes connected to different clusters.

## 3.2 | Number of components

Components are subgraphs in which all the pairs of vertices are connected to each other by at least one path and have no connections with the rest of the graph. In the directed graphs, when the nodes are reachable from every other node while ignoring directions is called a weakly connected component. If every node is mutually reachable from every other node, then the components are strongly connected.

In an ego network, the components are considered by ignoring the connections to the ego. The measure of the number of components in egos neighborhood shows the importance of ego as a bridge between components. The more the number of components (large) in ego's neighborhood, the ego is treated to be more important in regards to reaching many groups with a single point of contact (example: spreading information or virus).

## 3.3 | Effective size

The measure of effective size portrays the control of ego over alters or the benefit received for every unit invested over alters. It is the average nonredundancy score of all the primary alters in an ego network. It is given by the number of alters in the ego network minus the average degree of primary alters in the ego network while not considering the edges adjacent to the ego. If there are no edges between alters themselves then the ego is the only bridge between them with the highest betweenness. For example, when two primary alters are strongly connected to each other the information benefit to the ego from both of them is probably the same. In this context of information benefit they are considered redundant. The same case applies when two primary alters are not connected to each other but are connected to a mutual secondary network. Therefore information benefits are maximized in a nonredundant network. A limitation of this measure is that if there is another bridge or path between two nodes except ego and is not considered because it is not included in the ego network, then the controlling power of ego will be overestimated.

In the sparse networks like social networks (where density ≈ 0) the effective size of network increases as the number of alters increase, whereas in the dense networks it remains constant. Therefore when the number of alters increase in the sparser networks we can assume that the effective size is increasing. The maximum limit of effective size is equal to the number of alters in the network

$$ ES_e = \sum_j \left[ 1 - \sum_q p_{eq} m_{jq} \right], \qquad q \neq e,j, \tag{18} $$

where $p_{eq}$ is proportion of $i$'s energy invested in relationship with $q$, and $m_{jq}$ is calculated as $j$'s interaction with $q$ divided by $j$'s strongest relationship with anyone. In the simplest form it can be given as

$$ES_e = n - \frac{\sum\limits_{a=1}^{n} d_a - 1}{n}, \tag{19}$$

where $n$ is the number of alters and $d_a$ is the degree of an alter $a$. Borgatti (1997) reformulated the above equations in a more simplest form which is given as Equation (20), where $t$ is the total number of ties to the ego network while excluding the ties to ego:

$$ES_e = n - \frac{2t}{n}. \tag{20}$$

## 3.4 | Efficiency

Efficiency is a normalized form of the effective size of an ego-network. It is the effective size of an ego-network divided by the number of alters in it. Efficiency always lies between 0 and 1 (inclusive).

For example, if there are two or more ego networks of different sizes and you want to compare the benefits/authority, and so on, (application-specific) of the egos in them over their alters, then the efficiency is an appropriate measure as it averages per alter. The ego-network with high clustering coefficient would have less efficiency.

## 3.5 | Constraint

Constraint is similar to redundancy as in the measure of effective size but quite distinguishable as it also considers structural holes around every alter in the network. If the network is more redundant with less structural holes then the ego is more constrained. When two or more alters in an ego network are connected to each other then the ego is constrained in its actions. Constraint of an ego $e$ is the sum of constraints of all the alters over ego in the ego-network. Constraint ($C_{ej}$) of each alter $j$ over the ego $e$ is dependent on the proportion of its relationship to the ego and all the other alters in the network $P_{ej}$ and the sum of proportional relationship with the other alters in the network $P_{qj}$ and also their proportional relation with the ego $P_{eq}$. Which is given in the equation below:

$$C_{ej} = \left( P_{ej} + \sum_{q} P_{eq} P_{qj} \right)^2, \quad q \neq e, j. \tag{21}$$

One of the applicability of the above ego-centric measures was given by Burt (2009) to optimize structural holes in a network to yield maximum benefits from them. He analyzed the ego networks of managers and their growth pattern. He found that managers with increasing effective size reached top positions. He also included that though the effective size has positive effects on information benefits and control, it is dominated by the negative effects of constraint. According to his observations, the players/egos with relationships free of structural holes at their own end and rich in structural holes at other end are structurally autonomous and well positioned for the information and control benefits he regarded. Leveraging his observations can help investigate networks to gather information and insights. Tabassum and Gama (2016a) employed these measures to compare evolving ego-networks and their samples, to evaluate their efficiency.

## 3.6 | Krackhardt efficiency

Krackhardt Efficiency is one of the four measures defined by Krackhardt to measure the extent to which a graph is an out-tree. It is a measure of nonredundancy in multiple components of a graph or multiple weak components of a digraph (for the definition of components refer section 3.2). If we consider an ego network as a hierarchical structure and ego being the root node then the non-redundancy of edges can be calculated using this measure. The graph is said to be highly efficient if there are ($N_i - 1$) number of links between $N_i$ number of nodes in every component $G_i$ of a graph $G$. Efficiency is inversely proportional to the density of each component in the graph. If the density increases, the efficiency decreases. It is given by the equation below, where $E(G)$ is the total number of edges in graph $G$. In Figure 3 the krackhardt efficiency is calculated considering an undirected graph. Additionally, other measures defined by Krackhardt can also be employed over ego networks to identify specific properties delineated by them.

**FIGURE 3** Structurally different ego-networks for demonstration
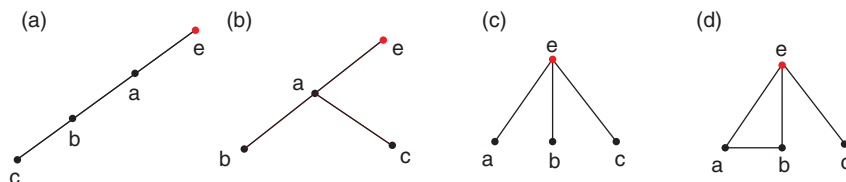
$$1 - \frac{E(G) - \sum\limits_{i=1}^{n} (N_i - 1)}{\sum\limits_{i=1}^{n} (N_i(N_i - 1) - (N_i - 1))} \qquad (22)$$

*Demonstration:* To analyze how the position of an ego is reflected by the above measures, we show some simple examples of ego networks in Figure 3, where e is an Ego with few alters. Their respective measures are demonstrated in Table 2 which expresses the relation between these measures.

## 4 | LINK ANALYSIS

In certain network settings, such as the web, one may be interested in finding the most valuable, authoritative or influential node (e.g., webpage), or a list of them. To perform this task several link analysis algorithms were devised, being the HITS29 and the Brin and Page (2012) algorithms the most popular ones. These algorithms explore the relationship between links and the content of web pages, in order to improve the task of information retrieval in the web, being of extreme importance for the design of efficient search engines. Since the development of these methods was motivated by the problem of web queries, for the sake of simplicity, we will explain them in this context.

### 4.1 | Hubs and authorities

Before introducing any of these algorithms, first, it is necessary to define some elementary concepts, namely, the concepts of hubs and authorities. In the web context, a hub can be understood as a web page that points to many other web pages or, in other words, as a compilation of web pages that address a specific topic. The quality of a hub is usually determined by the quality of the authorities it points to. On the other hand, authorities are web pages cited by many different hubs, which means that their relevance is measured by the number of inward links they receive. Typically, good authoritative pages are reliable sources of information about a given topic. In the following section, we explain the foundations of PageRank algorithm.

#### 4.1.1 | PageRank algorithm

PageRank is a link analysis algorithm based on the concept of eigenvector centrality. This algorithm is used by Google™ Internet search engine to rank web pages according to the value of the information they carry, so the most valuable ones appear at the top of the search results. The idea of the algorithm is that information on Web can be ranked according to link popularity (the more web pages are linked to a given web page the more popular that web page is). Nevertheless, in this process of weighting web pages, not only the number of links or, equivalently, the degree of a node is relevant, but also the importance of the web pages linking to them. Therefore, PageRank measures the relative importance of a set of web pages based not only on the quantity but especially the quality of their links.

The basic PageRank is computed as follows (according to the definition provided by Easley and Kleinberg (2010)):

Initialization: In a network of $n$ nodes (or web pages), assign a PageRank value of $1/n$ to each node, and choose the number of iterations $k$ of the algorithm. (a) Update the PageRank values of each node by sequentially applying the following rule: Basic PageRank Update Rule: divide the actual PageRank value of node $p$ by the number of its outgoing links and pass these equal shares to the nodes it points to. Note that if a node $p$ has no outgoing links, the PageRank share is passed to itself. The

**TABLE 2** Effective size, efficiency, Krackhardt efficiency, and constraint measures of ego-networks from Figure 3

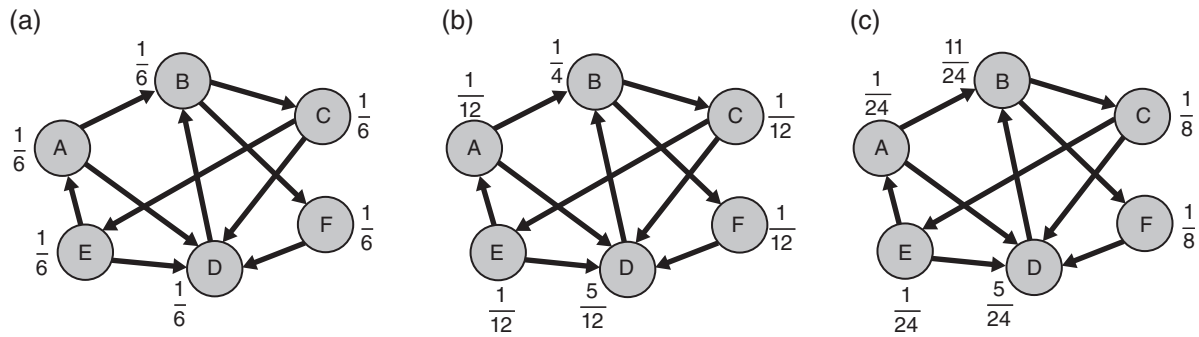| Measures/networks | Figure 3a | Figure 3b | Figure 3c | Figure 3d |
| --- | --- | --- | --- | --- |
| Effective size | 2.0 | 1.0 | 3.0 | 2.3333 |
| Efficiency | 0.6666 | 0.3333 | 1.0 | 0.7777 |
| Krackhardt efficiency | 0.6667 | 0.6667 | 0.6667 | 0.4444 |
| Constraint | 1.25 | 1.2222 | 0.3333 | 0.6111 |

**FIGURE 4** Illustration of the process behind Pagerank algorithm in a network comprised of six nodes. The first network (a) corresponds to the initialization step. In network (b) are shown the updated Pagerank values at the end of the first iteration of the algorithm. Note that node D is so far the most authoritative node, with a Pagerank value of $\frac{5}{12}$. The rightmost network (c) corresponds to the second (and last) iteration of Pagerank. Here, we notice that node B overtakes the position of node D in terms of Pagerank values

update of a node's PageRank value is performed by summing the shares it receives in each iteration. (b) Apply this rule until the $k$th iteration, or until convergence. To illustrate, consider the following example: in a network comprised of nodes termed A, B, C, D, E, and F, how can we find the most influential node, using the PageRank algorithm? First, and according to the initialization step of the algorithm, each node is assigned an equal PageRank of $PR = \frac{1}{n} = \frac{1}{6}$, as represented in Figure 4a. Then, these values are updated $k$ times (for the sake of simplicity we consider only two iterations) by applying the Basic PageRank Update Rule. To apply the rule, first it is necessary to compute the shares of all nodes. Then, for each node we sum all shares the node receives. The result of this sum will be its new PageRank value, as given by Table 3 and represented in Figure 4b. For instance, the share of node D, which has only one outgoing link, is computed as $share(D) = \frac{\frac{1}{6}}{1} = \frac{1}{6}$. Its new PageRank value is given by the sum of the shares of its ingoing links, namely, those coming from nodes A, C, E, and F:

$$PR(D) = \frac{1}{12} + \frac{1}{12} + \frac{1}{12} + \frac{1}{6} = \frac{5}{12}. \tag{23}$$

After computing these values for all nodes in the network, we repeat the process for the second iteration $k = 2$, obtaining the results shown in Table 4 and Figure 4c. This rule is applied iteratively until the convergence of the PageRank values, or until the $k$th iteration. Since we consider only two iterations, we can try to draw some conclusions and interpret the results based only on the information available in Tables 3 and 4. Therefore, at the end of the first iteration, node D seemed to be the most promising one, with a PageRank of 5/12; nevertheless, at the second iteration node B overtakes the position of D, being now assigned to the first place of the ranking of nodes. This sudden change befits the idea behind PageRank algorithm, that measures the quality, instead of the quantity, of a node's connections. Therefore, and besides node D is the one receiving more incoming links, the importance of the nodes linking to them is not that significant. On the other hand, node B has only two incoming links, but one of them is of great importance, namely, node D. This is the main reason why node B receives the larger PageRank value at the end of the second iteration, turning into the most influential, or authoritative, node in the network. If B was an actor, he/she would be considered the most important one, once this PageRank value means that a great part of the information that flows through the network passes through it.

## 4.2 | Link prediction

The link prediction problem has been traditionally approached in a static environment: given a snapshot of the network, the goal is to infer which links are missing (Liben-Nowell & Kleinberg, 2003). Assuming the network will evolve to other state in the future, the link prediction problem may also be interpreted as the problem of predicting which links are more likely to appear in the future, given the current state of the network.

**TABLE 3** Updated Pagerank values after the first iteration $K = 1$

| Node | A | B | C | D | E | F |
|------|------|------|------|------|------|------|
| Shares | 1/12 | 1/12 | 1/12 | 1/6 | 1/12 | 1/6 |
| Updated Pagerank | 1/12 | 1/4 | 1/12 | 5/12 | 1/12 | 1/12 |

**TABLE 4** Updated Pagerank values at the end of the second (and last) iteration $K = 2$

| Node | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| Shares | 1/24 | 1/8 | 1/24 | 5/12 | 1/24 | 1/12 |
| Updated Pagerank | 1/24 | 11/24 | 1/8 | 5/24 | 1/24 | 1/8 |

Temporal link prediction refers to the problem of link prediction in time-evolving networks, in which multiple snapshots of the network are available. In global terms, this problem is defined as follows: given the states of the network for the previous $T$ time instants, how to predict future (new or re-occurring) links? Thus, given the sequence of states of the network at instants 1 to $T$, the goal is to predict which are the links which are more likely to occur at instant $T + 1$ (Fernandes, Tork, & Gama, 2017).

The problem of temporal link prediction has been tackled by considering both time-agnostic (in which the temporal information is not exploited) and time-aware methods (in which temporal information is incorporated).

In the context of time-agnostic methods, the most common approach is to collapse all the network time stamps and to apply the traditional methods, designed for static environments, to the collapsing result. These traditional approaches consist in computing a similarity score between each pair of network nodes so that higher scores reflect higher similarity. These scores are usually defined based on the node neighborhood of the nodes (Adamic & Adar, 2003; Newman, 2001; Salton & McGill, 1986) or on the paths between the nodes (Katz, 1953; Lü, Jin, & Zhou, 2009; Papadimitriou, Symeonidis, & Manolopoulos, 2012).

O'Madadhain, Hutchins, and Smyth (2005) and Wang, Satuluri, and Parthasarathy (2007) considered classification oriented methods. The idea of these methods is to use a set of network features to train a classifier, which is further applied to predict future links. Both works used logistic regression, however, in the second work, the authors considered co-occurrence probabilities in addition to the topological and semantic features.

Regarding time-aware methods, the most common approach is to consider time-series-based methods. The idea of such methods is to use the state of the network in the given instants to construct time-series, which are further forecasted. The differences among the existing methods reside on at least one of the following specifications: (a) the type of feature being modeled by the time-series, (b) the forecasting models, and (c) the forecasts post-processing. In this context, one of the first methods was proposed by Huang and Lin (2009), which modeled occurrence frequency using ARIMA. The major limitation of this work is its inability to predict new links, that is, the method only predicts re-occurring links. This limitation was addressed in precedent works by considering other types of features such as similarity scores (da Silva Soares & Prudêncio, 2012; Güneş, Gündüz-Öğüdücü, & Çataltepe, 2016; Hajibagheri, Sukthankar, & Lakkaraju, 2016). These methods also extended the existing work by considering other forecasting models. Moreover, da Silva Soares and Prudêncio (2012) and Hajibagheri et al. (2016) subjected the forecasting result to an SVM.

Dunlavy, Kolda, and Acar (2011) and Spiegel, Clausen, Albayrak, and Kunegis (2011) considered the modeling of the time-evolving network as a tensor and resorted to tensor decomposition techniques, combined with forecasting models to estimate the future time slice of the tensor, corresponding to the future state of the network.

Bringmann, Berlingerio, Bonchi, and Gionis (2010) and Juszczyszyn, Musial, and Budka (2011) proposed methods which were based on the mining of evolution patterns.

## 5 | COMMUNITY DETECTION

One of the unique features of social networks is that they tend to show community structure. This property usually arises as a consequence of both global and local heterogeneity of edges distribution in a graph. Thus, we often find high concentrations of edges within certain regions of the graph, called communities, and low concentration of edges between those regions. Communities, also known as modules or clusters, can be straightforward defined as similar groups of nodes. A more complete definition is built upon the concept of density: communities can be understood as densely connected groups of vertices in the network, with sparser connections between them. The connections can be directed (Twitter) and undirected (Facebook).

According to Newman and Girvan (2004), there are two main lines of research in discovering communities in network data. The first has its origins in Computer Science and is known as graph partitioning, while the second has been mainly pursued by sociologists and is usually referred as block-modeling, hierarchical clustering or community structure detection. The former originally arose in the Computer Sciences field due to the necessity of finding the best way to allocate tasks to processors so as to minimize the communications between them. This network optimization task aimed at enhancing the computation, in a parallel computing environment. The latter was motivated by the discovery of community groups within society,
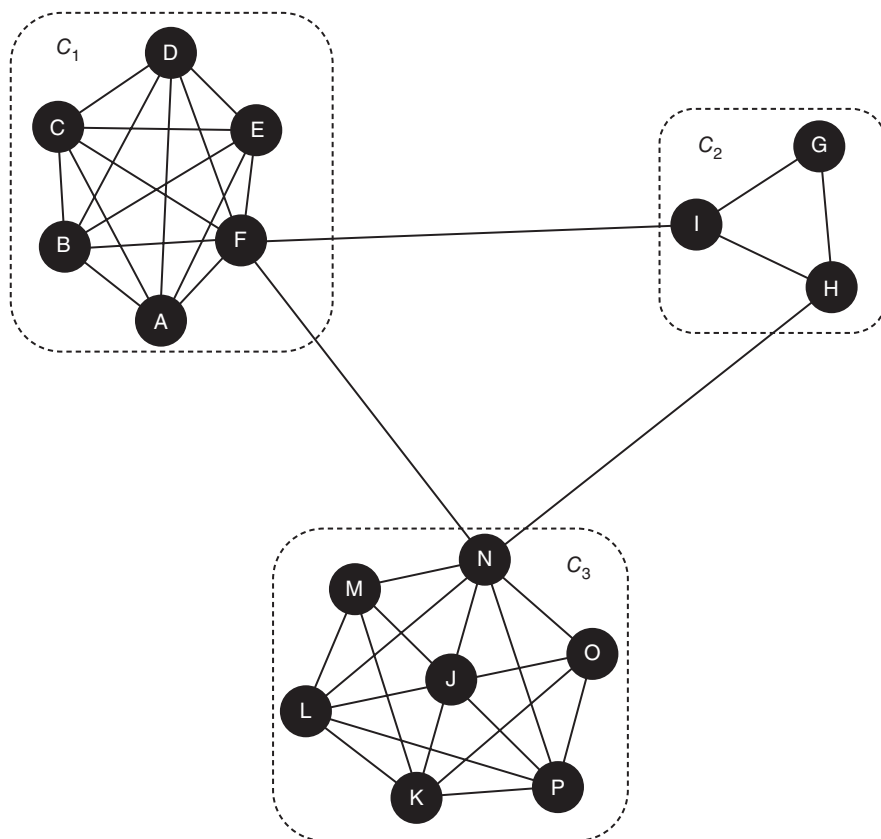
**FIGURE 5** An example network with three distinct communities: $C_1 = \{A,B,C,D,E,F\}$, $C_2 = \{G,H,I\}$, $C_3 = \{J,K,L,M,N,O,P\}$

in order to simplify the analysis of social phenomena through the arrangement of people according to their similarities. The main process behind community detection algorithms is based on dividing the original graph, into a set of disjoint subgraphs, through the optimization of a given objective function (e.g., modularity). The aim of both approaches is to discover groups of related vertices in the network and, if possible, the corresponding hierarchical organization, based only on the information provided by network topology. This is usually done by iteratively removing the bridges between groups of vertices, as suggested by Girvan and Newman (2002).

To better understand the introduced concepts, Figure 5 depicts a simple network comprising three communities, named, $C_1$, $C_2$, and $C_3$. In this picture we represent an ideal situation since each community is itself a complete graph, or a clique, of varying size ($C_1 = K_6$, $C_2 = K_3$, and $C_3 = K_7$). Also, the density of ties between communities is very low. The few ties that exist are bridges, since they are the only available connections between different parts of the network.

In real life we can find several examples of such tight groups. There is a long list of examples, so we will only name a few. Society is a rich environment for finding communities, once people have the natural tendency to form groups. These groups can be families, circles of friends, working and/or religious groups, towns, nations, and so on. If we also consider groups formed by companies, or by customers of a given product, we can identify communities with relevance to economics and business fields. Biology is another activity where methods for finding communities are useful, especially within the scope of metabolic networks. For instance, in protein–protein interaction networks we can find groups of proteins with similar functions within the cell. We can also find virtual online communities in the network of Internet, or groups of topic-related web pages, which may be useful for the development of automatic and efficient recommendation systems. The importance of studying these communities is intuitive in domains such as SNA. To highlight this importance Fortunato (2010) stated that the analysis of the structural position of nodes, in each module, can help identify central actors (those within central positions), often associated to group control and stability functions, as well as intermediate actors, who are those who lie at the boundaries of communities and play a key role in the spread and exchange of new ideas and information, creating bridges between communities. Other interesting possibility opened by the task of discovering communities is the one that focus on the analysis of coarse-grained descriptions of the original graph. An example is the study of graphs obtained by considering vertices as communities and edges between them as an indicator of overlap between communities. This strategy is used by Ågotnes (2010) for the detection of transitions in clusters.

The following sections are devoted to the introduction of the most popular (not necessarily the best) methods to solve the problem of finding communities. The great majority of these traditional algorithms assume partitions of vertices, instead of covers (Moreno, 1934), that is, they do not allow overlap of communities, so each vertex is assigned to a single community.

However, if one suspects that the nature of his/her network implies the existence of overlapping communities, a possible choice is the Clique Percolation Method (CPM), proposed by the physicists Palla, Derényi, Farkas and Vicsek (2005). The main feature of this prominent approach is its ability to find overlapping communities in a network, by allowing vertices to belong to more than one group. This characteristic is especially appealing in social sciences, as people tend to belong to more than one community (e.g., family, work, friends, etc.) at the same time.

For those interested in using CPM to detect overlapping communities, Palla and his colleagues developed the CFinder software package, which is freely available at www.cfinder.org.

## 5.1 | Hierarchical clustering

Hierarchical clustering is a popular class of methods for finding clusters, since it does not require any assumptions regarding their number, membership, and size. Hierarchical clustering algorithms produce a flexible nested structure (smaller clusters within larger clusters which, in turn, are embedded in even larger clusters), typically represented by means of a dendrogram, that uncovers the multilevel structure of the network. Such features are highly desired in domains where little information is available concerning the community structure of a network. In addition, these methods proved to be quite effective in solving cluster analysis problems, thus becoming attractive for graph partitioning and community detection purposes.

The procedure of traditional hierarchical clustering is quite intuitive, being strongly based on the definition of similarity. Usually, the first step is the selection of the similarity measure that will be used to assess how alike two objects are according to a given global, or local, property. Examples of such measures are the cosine similarity, the Jaccard index, the Euclidean or Manhattan distances, the Hamming distance, among others. The next step is to compute the similarity matrix between all pairs of objects. Then, one chooses the approach to group them—the agglomerative or the divisive—and, depending on the choice, selects a given distance measure to compute the similarity between clusters (e.g., single linkage, complete linkage, Ward's method, etc.). The result is a dendrogram illustrating the arrangement of clusters returned by the hierarchical algorithm.

In the context of graphs, the goal of hierarchical clustering is to sequentially group similar nodes. Such similarity/ dissimilarity may be quantified based on the structural properties of the nodes in the network. For example, the similarity measure between two nodes may be defined as the number of common neighbors so that nodes sharing most of their neighbors are grouped in the same cluster (Wasserman & Faust, 1994). When considering such measure, it may occur that nodes of the same community are not grouped in the same cluster (Newman, 2004). Other metrics and strategies have been proposed in order to capture the community structure of the networks using hierarchical clustering. In particular, as mentioned before, there are two general approaches:

- Divisive methods: this class of methods focuses on identifying and removing the spanning links between densely connected regions (Easley & Kleinberg, 2010), namely, bridges and local bridges. A well-known algorithm exploring this method is the one proposed by Girvan and Newman (2002).
- Agglomerative methods: this class of methods focuses on the tightly-knit parts of the network, rather on the connections at their boundaries. Walktrap (Pons & Latapy, 2005) is an example of an algorithm based on this method. In the next section we present one of the best known and widely used divisive hierarchical algorithm for finding communities, especially in social networks: the algorithm of Girvan and Newman.

In order to select the best partition, that is, the best number of communities $k$, a typical strategy is to compute the value of modularity (Newman, 2003) for every possible number of clusters and select the number that maximizes this function.

Finally, it is noteworthy that communities usually have a hierarchical structure (Porter, Onnela, & Mucha, 2009). For example, in a friendship ego-network of a given individual, we may find a large community corresponding to people the individual met at his home town. Then, among such people, we can find smaller communities such as his family or his school colleagues. The exploration of community structure in a hierarchical way allows to capture such different levels of homogeneity in the communities.

## 5.2 | Girvan–Newman algorithm

Among the most popular algorithms, or even the most popular one, for solving community detection problems is the one devised by Girvan and Newman (2002) and known as the Girvan–Newman algorithm. The Girvan–Newman algorithm is a divisive hierarchical technique that deconstructs the initial full network into progressively smaller connected pieces, until the point where there are no edges to remove and each node represents itself a community. Bearing in mind that communities are cohesive groups of nodes, with sparser connections between them, the criterion to remove the edges, proposed by them,

is the graph-theoretic centrality measure edge betweenness. The reason behind this choice is related to the fact that this centrality measure is able to identify edges that lie on a large number of shortest paths between nodes and, therefore, are believed to connect different nonoverlapping communities. Thus, the main idea of this algorithm is that if we identify and remove bridges, we isolate the existing communities in a network.

Since it is based on the concept of betweenness it is only suitable for networks of moderate order (up to a few thousand nodes), due to the high cost of computing it. The input of the algorithm is a full graph and the output is a hierarchical structure, such as a dendrogram, where communities at any level correspond to a horizontal cut through this hierarchical tree. The steps of the algorithm can be summarized as follows:

1. Compute the betweenness of all edges in the network;
2. Remove the edge with highest betweenness. This step may cause the network to split into separate disconnected parts, which constitute the first level of regions in the partitioning of the graph.
3. Repeat the previous steps until there are no edges to remove in the graph. Note that the obtained smaller components, within larger components, are the regions nested within the larger regions found in the first steps. Due to its popularity, almost all standard software libraries have this algorithm implemented. For instance, in R (Ihaka & Gentleman, 1996) we can use the edge.betweenness.community function, provided by library igraph, to apply the Girvan–Newman algorithm.

## 5.3 | Modularity optimization

A widely used and very popular class of methods to detect communities in networks is modularity maximization. Modularity $Q$ is a quality function that attempts to measure the merit of a given partition of the network into communities. It has been used to compare the quality of the partitions obtained by different community detection methods, but also as an objective function to optimize. According to Newman (2006), Modularity is, up to a multiplicative constant, the number of edges falling within groups minus the expected number in an equivalent network with edges placed at random. Based on this definition, we can deduce that modularity is a measure that explicitly takes into account the heterogeneity of the edges. The basic idea is that a network shows meaningful community structure if the number of edges between communities is fewer than expected on the basis of random choice. By assumption, the higher its value the better the partition, meaning that the found communities are internally densely connected and externally sparsely connected, since there are more edges falling within groups than what would be expected by chance. Modularity is computed as:

$$Q = \frac{1}{2m} \sum \left[ A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j), \tag{24}$$

where $m$ is the number of edges, $k_i$ and $k_j$ represent, respectively, the degree of nodes $i$ and $j$, $A_{ij}$ is the entry of the adjacency matrix that gives the number of edges between nodes $i$ and $j$, $\frac{k_i k_j}{2m}$ represents the expected number of edges falling between those nodes, $c_i$ and $c_j$ denote the groups to which nodes $i$ and $j$ belong, and $\delta(c_i, c_j)$ represents the Kronecker delta.

From the formula, we can deduce that $Q \in [-1, 1]$, being either negative or positive. If positive, then there is a possibility of finding community structure on the network. If $Q$ is not only positive, but also large, then the corresponding partition may reflect the real community structure. According to Clauset, Newman, and Moore (2004), in practice it was found that a modularity of about 0.3 is a good indicator of the existence of meaningful communities.

Following this reasoning, and knowing that the higher the modularity, the best the obtained network division is, a natural approach would be maximizing this measure, by computing it for every possible partition of the network and selecting the partition returning the higher value. This simple idea gave rise to a new class of methods whose foundations are set on the maximization of modularity. Albeit this approach is quite attractive, the exhaustive search over all possible divisions is usually intractable. This undesired effect of computational inefficiency has been circumvented by adapting a number of heuristic methods to this specific optimization problem. Following this strategy one can obtain a fairly good approximation of the global optimum (in this case, the maximum value of modularity) in an acceptable time. Algorithms that employ this strategy are, for instance, the one proposed by Blondel, Guillaume, Lambiotte, and Lefebvre (2008), which performs a hierarchical optimization of modularity by exploring greedy techniques, and the one proposed by Guimera and Amaral (2005), that applies the Simulated Annealing procedure to the modularity optimization problem. Those interested in knowing more about the problem of finding communities in networks, can refer to the recently released survey by Fortunato (2010).

## 6 | EVOLVING NETWORKS

Networks with changes in the number/behavior/features of nodes and links as a function of time are known as evolving networks. For example, the social network of people living in this world can be considered evolving, as new nodes get added while some of the nodes expire and connections/relations between the nodes keep on forming and breaking across time. This phenomenon causes changes in the structure of the network as a whole, across time. There exist many examples of time-evolving networks, as shown above, most of them being generated from real-world applications. Statistical analysis of these networks in itself is a challenge because of their transient structure and distributions, added to their size and complexities. Last two decades have encountered extensive research in the area of network analysis to study the evolution of structure and properties of these networks for various purposes like generating synthetic or artificial networks, developing models for graph mining (link prediction/recommendation, community/anomaly detection, event detection/prediction, classification/segmentation, pattern mining, etc.), decision making, solution optimization/influence maximization, network analytics, and so on. In the section below (section 6.1), we present a brief survey of works that indulged in the evolution analysis of networked data.

In section 6.2 we specifically focus on the temporal property of evolving networks. Note that the evolving graphs are temporal but the temporal graphs need not always be evolving they can be dynamic though that is, nodes need not be added or deleted in the networks (sometimes the temporal property is only associated with edges).

### 6.1 | Evolving network analysis

Apart from using the SNA measures given in section 2 for a static network analysis, these measures are also used to study the dynamic and evolving properties of networks. We delineate some works below, which used these properties to characterize evolving networks based on the regularities from their results of the analysis.

Researchers in this area (Aiello, Chung, & Lu, 2000; Albert & Barabâsi, 2000; Barabâsi et al., 2002; Huberman & Adamic, 1999b; Krapivsky, Rodgers, & Redner, 2001) found that most of the real world data networks grow by following a power-law degree (in-degree, out-degree, or undirected) distribution (at least asymptotically) (for the definition of degree distribution refer to Oliveira and Gama (2012)). Barabâsi and Albert (1999) reasoned it was because of the preferential attachment of the nodes, that is, new nodes attach preferentially to the already well-connected nodes. Dorogovtsev and Mendes (2001) proved that different kinds of preferential linking produce different types of scale-free networks (whose degree distribution follows a power-law). Reed and Jorgensen (2004) proved that if a stochastic process that grows exponentially, and is observed once "randomly," the distribution of the observed state will follow power-laws in one or both tails.

Huberman and Adamic (1999a) while exploring the growth dynamics of world wide web, explained that for sites which are typically organized in hierarchical, tree-like, fashion, the number of pages added at any given time to a site will be proportional to those already existing there and the growth rate $g(t)$ of number of pages per site is uncorrelated from one time interval to the other about a positive mean value $g_0$. As a consequence, each particular growth rate gives rise to a power law distribution. They also indicated that the evolutionary dynamics of the web are dominated by occasional bursts in which a large number of pages suddenly appear at a given site. These bursts are responsible for the long tail of the probability distribution and make average behavior to depart from typical realizations. They concluded that those networks evolve in an asymptotically self-similar structure without having a natural scale.

While most of the previous works concentrated only on the sparsity of the networks, Faloutsos, Faloutsos, and Faloutsos (1999) derived some more interesting relationships in the evolution of networks. They analyzed the Internet topologies over three instances in a year, where the size of network increased by 45%. As a consequence they defined four power-law relationships (one is an approximation) on a growing network, which are stated as: *Power-Law 1 (rank exponent):* The outdegree, $d_v$, of a node $v$, is proportional to the rank of the node, $r_v$, to the power of a constant, $R$. *Power-Law 2 (outdegree exponent):* The frequency of an outdegree, $f_d$ (the number of nodes with outdegree $d$), is proportional to the outdegree $d$ to the power of a constant, $O$. *Power-Law 3 (Eigen exponent):* The eigenvalues, $\lambda_i$, of a graph are proportional to the order, $i$, to the power of a constant, $E$. Where $i$ is the order of $\lambda_i$ in the decreasing sequence of eigenvalues. *Approximation 1 (hop-plot exponent):* The total number of pairs of nodes, $P(h)$, within $h$ hops, is proportional to the number of hops to the power of a constant, $H$.

Newman (2001) tested the previous theories of clustering and preferential attachment over growing networks. He proved empirically the probability of a link between two nodes is strongly positively correlated with the number of mutual acquaintances/neighbors exponentially and the number of previous links linearly.

Further ahead Barabâsi et al. (2002) analyzed the topological properties of very sparse co-authorship networks (like approximately 70 K nodes and 70 K edges aggregated) over smaller time steps (compared to the above-referenced literature) of 1 year for 7 years. Based on their empirical measurements the authors discovered that the average degree of these networks increases in time, and the node separation decreases. In addition, they also found the clustering coefficient of these networks decays with time while relative size of the largest cluster increases.
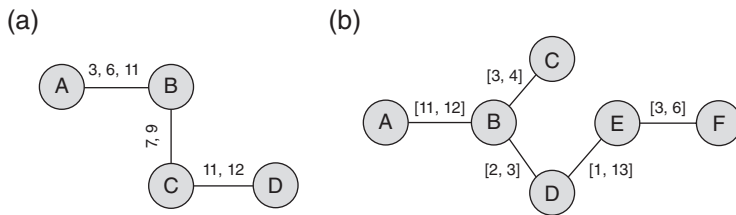
(a)

(b)



**FIGURE 6** Temporal networks represented as (a) contact sequence network and (b) interval graph

Leskovec et al. (2007) extensively studied the evolving properties of networked data from seven different domains with varied time spans and nodes and edges arriving at different speeds. Based on their results they showed that the average degree of these networks increases as a function of time (as given by Barabâsi et al. (2002)), hence densifying these networks over time and their densification follows a power law given in Equation (25), where $e(t)$ and $n(t)$ denote the number of edges and nodes of the graph at time $t$, and $a$ is an exponent that generally lies strictly between 1 and 2. Added to that, the effective diameters of those networks keep decreasing over time. Leskovec, Backstrom, Kumar, and Tomkins (2008) tried to find the preference of nodes for the formation of new edges in the network evolution process. They demonstrated that the preferential attachment concept given by Barabási and Albert (1999) based on degree and age of nodes is inherently nonlocal and biased. On the contrary, the locality of nodes plays an important role in edge formation, and most of the new edges form by typically closing triangles. Therefore locality with preferential attachment can supplement similar network generating models

$$e(t) \propto n(t)^{a}. \tag{25}$$

The works referred above have mainly focused on the analysis of evolving networks to yield tractable mathematical models or simple generative models of network growth. Consequently, these models can be used to explore their latent properties. The success of these models depends on their ability to incorporate the growth statistics of those networks. Additionally, some studies used the growth statistics of networks to evaluate their sampling techniques (Leskovec & Faloutsos, 2006; Tabassum & Gama, 2016a, 2016b). However, the analysis of networks is also a primary step in developing models in an evolving network mining task.

## 6.2 | Temporal networks

Networks explicitly representing the times *when* edges are active are defined as temporal networks. A classical example of a temporal network application is on disease contagion through physical proximity (Holme & Saramaki, 2012). Normally, the spreading of pathogenic organisms occurs through contact between two individuals and a temporal network is the best structure to represent this scenario. Social networks can also be represented as temporal networks, as they are increasingly ubiquitous and complex on their interactions (Holme, 2014). Another examples of networks that can be represented as temporal networks are: face-to-face communications (Cattuto, Quaggiotto, Panisson, & Averbuch, 2013), flights networks (Wu et al., 2014) and phone calls (Tabassum & Gama, 2016c). There are many definitions in literature that formalize temporal networks (here, invariably also called temporal graphs). Kim and Anderson (2012) defined the *time-ordered graphs* and Nicosia et al. (2013) call them *time-varying graphs*, but generally all definitions represent a set of time-edges among a set of nodes during an observation interval that takes into account their temporal ordering.

Formally, a temporal network $G = (V, E)$ is a set $E$ of edges registered among a set of nodes $V$ during an observation interval $[0, T]$. An edge between two nodes $u, v \in V$ is represented by a quadruplet $e = (u, v, t, \delta t)$, where $0 \leq t \leq T$ is the time at which the edge started and $\delta t$ is its duration. The edges can also be called contacts.

The above definition is classical for representing flight graphs and phone calls networks, for example. But there are extensions to this definition. When contacts are instantaneous, $\delta t \rightarrow 0$, the temporal network is defined as a *contact sequence graph* (Holme & Saramaki, 2012). These graphs are used to represent systems in which the duration of the contact is less important (e-mails, sexual networks, likes in social networks). Another variation is to define temporal networks with edges that are not active over a set of times but rather over a set of intervals $e = (u, v, t_{init}, t_{end})$. These are the *interval graphs* (Holme & Saramaki, 2012), good for modeling follow/unfollow relationships in Twitter network (Pereira, Amo, & Gama, 2016) or infrastructural systems like the Internet. In fact, interval graphs can be transformed into contact sequence graphs and most of the network analysis techniques hold in both cases.

Figure 6 illustrates two temporal networks. Let us consider the context of Twitter social network. In Figure 6a, we have a contact sequence graph, representing mention interactions among users. The nodes are users and an edge $(u, v, t)$ indicates that $u$ mentioned $v$ in a tweet posted at $t$.[1] The times of when the interactions occurred are represented next to the edges and the duration of the interactions are negligible. We can see that the users $A$ and $B$ interacted at times 3, 6, and 11, the users $B$ and $C$ interacted at 7 and 9 and so on. Now in the same Twitter context, we can consider the interval graph in Figure 6b

where the edges represent follower/followee relationships and the intervals indicate that these relationships start at $t_{init}$ and finish at $t_{end}$. As an example, $E$ starts following $F$ at 3 and unfollows at 6.

## 7 | CONCLUSIONS AND FUTURE TRENDS

In this article, we presented a concise overview of social network analysis methods, objectives, and applicability. A number of SNA measures and related tasks in view of different types of networks are demonstrated. With the discovery of networks in most of the applications' generated data and the quality of information extracted, network analysis is gaining much popularity these days. The complexity is increasing as the available amount of data is increasing. Advancements in processing, manipulating and mining high-velocity massive scale networks is one of the current vital concerns. With the predominance of web 2.0, IOT and Industry 4.0, and others, it is only going to get more demanding and challenging.

As the networks generating in real time are not static but dynamic and evolving, the recent works have been profoundly interested in exploring the growth patterns, mining problems and resolving the challenges associated with it. Some of the latest works are concentrated in designing algorithms for faster, incremental and memory efficient computations of SNA measures (discussed in this paper) for very large and high velocity graphs, which is still a challenge for many of the metrics that needs traversing entire or most of the graph on every update. Link prediction, community detection, clustering, and so on, are actively studied fields in recent decades but still lack efficiency. The current and future trends also include application-specific usability, scalability, and enhancements in these areas. The emerging lines of applicability are in the areas of social reputation, smart cities, multiagent systems, intelligent objects, bio-informatics, earth sciences, cognitive sciences, mobility patterns, recommendations, and so on, besides the existing realms of fraud detection, social media, gene expressions, protein interactions, marketing, churn prediction, and so on. The recent surging demands by these applications on the complex real world have also required advancements in the area of diverse network topologies like multilevel, heterogeneous, evolving networks, and so on. Therefore, this article paves the way for a basic understanding of more complex problems associated with network analysis.

### CONFLICT OF INTEREST

The authors have declared no conflicts of interest for this article.

### RELATED WIREs ARTICLE

An overview of social network analysis

### NOTE

[1]*Tweet* is a slang term to describe what a user posts in Twitter. *Mention* is a tweet that contains another user's username anywhere in the body of the tweet.

### ORCID

*João Gama* http://orcid.org/0000-0003-3357-1195

### REFERENCES

Abraham, A., Hassanien, A.-E., & Sná, V. (2009). *Computational social network analysis: Trends, tools and research advances*. London: Springer Science & Business Media.

Adamic, L. A., & Adar, E. (2003). Friends and neighbors on the web. *Social Networks, 25*, 211–230.

Aggarwal, C. C. (2009). Models for incomplete and probabilistic information. In *Managing and mining uncertain data* (pp. 1–34). Springer.

Ågotnes, T. (2010). Mec-monitoring clusters' transitions. In *Stairs 2010: Proceedings of the fifth starting AI researchers' symposium* (Vol. 222, p. 212). Amsterdam: IOS Press.

Aiello, W., Chung, F., & Lu, L. (2000). A random graph model for massive graphs. In *Proceedings of the thirty-second annual ACM symposium on theory of computing* (pp. 171–180). Portland, Oregon: ACM.

Akcora, C. G., & Ferrari, E. (2014). Discovering trust patterns in ego networks. In *2014 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)* (pp. 224–229). New York, NY: IEEE.

Albert, R., & Barabási, A.-L. (2000). Topology of evolving networks: Local events and universality. *Physical Review Letters*, *85*, 5234–5237.

Barabási, A.-L. (2016). *Network science*. New York, NY: Cambridge University Press.

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

Barabâsi, A.-L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., & Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications*, *311*, 590–614.

Barabási, B. A.-L., & Bonabeau, E. (2003). Scale-free. *Scientific American*, *288*, 50–59.

Blondel, V., Guillaume, J.-L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, *10*, P10008.

Bonacich, P. (1987). Power and centrality: A family of measures. *American Journal of Sociology*, *92*, 1170–1182.

Borgatti, S. P. (1997). Structural holes: Unpacking burt's redundancy measures. *Connect*, *20*, 35–38.

Borgatti, S. P., & Everett, M. G. (1997). Network analysis of 2-mode data. *Social Networks*, *19*, 243–269.

Brin, S., & Page, L. (2012). Reprint of: The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, *56*, 3825–3833.

Bringmann, B., Berlingerio, M., Bonchi, F., & Gionis, A. (2010). Learning and predicting the evolution of social networks. *IEEE Intelligent Systems*, *25*, 26–35.

Burt, R. S. (2009). *Structural holes: The social structure of competition*. Cambridge, MA: Harvard University Press.

Cattuto, C., Quaggiotto, M., Panisson, A., & Averbuch, A. (2013). Time-varying social networks in a graph database: A neo4j use case. In *First international workshop on graph data management experiences and systems* (Vol. 11). New York, NY: ACM.

Charu, A. C., & Aggarwal, R. (2011). *Social network data analytics*. New York: Springer-Verlag New York Inc.

Clauset, A., Newman, M., & Moore, C. (2004). Finding community structure in very large networks. *Physical Review E*, *70*, 066111.

Costa, L. d. F., Oliveira, O. N., Jr., Travieso, G., Rodrigues, F. A., Villas Boas, P. R., Antiqueira, L., … Correa Rocha, L. E. (2011). Analyzing and modeling real-world phenomena with complex networks: A survey of applications. *Advances in Physics*, *60*, 329–412.

Dasgupta, K., Singh, R., Viswanathan, B., Chakraborty, D., Mukherjea, S., Nanavati, A. A., & Joshi, A. (2008). Social ties and their relevance to churn in mobile telecom networks. In *Proceedings of the 11th international conference on extending database technology: Advances in database technology* (pp. 668–677). New York, NY: ACM.

Diestel, R. (1990). *Graph decompositions: A study in infinite graph theory*. Gloucestershire: Clarendon Press Oxford.

Domingos, P., & Richardson, M. (2001). Mining the network value of customers. In *Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 57–66). New York, NY: ACM.

Dorogovtsev, S. N., & Mendes, J. F. (2001). Scaling properties of scale-free evolving networks: Continuous approach. *Physical Review E*, *63*, 056125.

Dunlavy, D. M., Kolda, T. G., & Acar, E. (2011). Temporal link prediction using matrix and tensor factorizations. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *5*, 10.

Easley, D., & Kleinberg, J. (2010). *Networks, crowds and markets: Reasoning about a highly connected world*. New York, NY: Cambridge University Press.

Epasto, A., Lattanzi, S., Mirrokni, V., Sebe, I. O., Taei, A., & Verma, S. (2015). Ego-net community mining applied to friend suggestion. *Proceedings of the VLDB Endowment*, *9*, 324–335.

Everett, M., & Borgatti, S. P. (2005). Ego network betweenness. *Social Networks*, *27*, 31–38.

Faloutsos, M., Faloutsos, P., & Faloutsos, C. (1999). On power-law relationships of the internet topology. In *ACM SIGCOMM computer communication review* (Vol. 29, pp. 251–262). New York, NY: ACM.

Fernandes, S. d. S., Tork, H. F., & Gama, J. M. P. d. (2017). The initialization and parameter setting problem in tensor decomposition-based link prediction. In *2017 I.E. international conference on data science and advanced analytics (DSAA)* (pp. 99–108). Los Alamitos, CA: IEEE.

Fortunato, S. (2010). Community detection in graphs. *Physics Report*, *486*, 75–174.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, *1*, 215–239.

Furht, B. (2010). *Handbook of social network technologies and applications*. New York, NY: Springer Science & Business Media.

Girvan, M., & Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, *99*, 7821–7826.

Granovetter, M. (1995). *Getting a job: A study of contacts and careers*. Chicago, IL: University of Chicago Press.

Granovetter, M. S. (1973). The strength of weak ties. *American Journal of Sociology*, *78*, 1360–1380.

Guimera, R., & Amaral, L. (2005). Functional cartography of complex metabolic networks. *Nature*, *433*, 895–900.

Güneş, İ., Gündüz-Öğüdücü, Ş., & Çataltepe, Z. (2016). Link prediction using time series of neighborhood-based node similarity scores. *Data Mining and Knowledge Discovery*, *30*, 147–180.

Hajibagheri, A., Sukthankar, G., & Lakkaraju, K. (2016). Leveraging network dynamics for improved link prediction. arXiv preprint arXiv:1604.03221

Hanneman, R. A., & Riddle, M. (2005). *Introduction to social network methods*. Riverside, CA: University of California, Riverside.

Holme, P. (2014). Analyzing temporal networks in social media. *Proceedings of the IEEE*, *102*, 1922–1933.

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

Huang, Z., & Lin, D. K. (2009). The time-series link prediction problem with applications in communication surveillance. *INFORMS Journal on Computing*, *21*, 286–303.

Huberman, B. A., & Adamic, L. A. (1999a). Evolutionary dynamics of the world wide web. arXiv preprint cond-mat/9901071

Huberman, B. A., & Adamic, L. A. (1999b). Internet: Growth dynamics of the world-wide web. *Nature*, *401*, 131–131.

Ihaka, R., & Gentleman, R. (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, *5*, 299–314.

Juszczyszyn, K., Musial, K., & Budka, M. (2011). Link prediction based on subgraph evolution in dynamic social networks. In *2011 I.E. third international conference on privacy, security, risk and trust (PASSAT) and 2011 I.E. third inernational conference on social computing (SocialCom)* (pp. 27–34). New York, NY: IEEE.

Katz, L. (1953). A new status index derived from sociometric analysis. *Psychometrika*, *18*, 39–43.

Kim, H., & Anderson, R. (2012). Temporal node centrality in complex networks. *Physical Review E*, *85*, 026107.

Kossinets, G., & Watts, D. J. (2006). Empirical analysis of an evolving social network. *Science*, *311*, 88–90.

Krapivsky, P., Rodgers, G., & Redner, S. (2001). Degree distributions of growing networks. *Physical Review Letters*, *86*, 5401–5404.

Latapy, M., Magnien, C., & Del Vecchio, N. (2008). Basic notions for the analysis of large two-mode networks. *Social Networks*, *30*, 31–48.

Leskovec, J., Adamic, L. A., & Huberman, B. A. (2007). The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)*, *1*, 5.

Leskovec, J., Backstrom, L., Kumar, R., & Tomkins, A. (2008). Microscopic evolution of social networks. In *Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 462–470). New York, NY: ACM.

Leskovec, J., & Faloutsos, C. (2006). Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 631–636). New York, NY: ACM.

Leskovec, J., Kleinberg, J., & Faloutsos, C. (2005). Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining* (pp. 177–187). New York, NY: ACM.

Liben-Nowell, D., & Kleinberg, J. (2003). The link prediction problem for social networks. In *Proceedings of the twelfth annual ACM international conference on information and knowledge management (CIKM)* (pp. 556–559). New York, NY: ACM.

Liu, C., Wang, J., Zhang, H., & Yin, M. (2018). Mapping the hierarchical structure of the global shipping network by weighted ego network analysis. *International Journal of Shipping and Transport Logistics*, *10*, 63–86.

Lü, L., Jin, C.-H., & Zhou, T. (2009). Similarity index based on local paths for link prediction of complex networks. *Physical Review E*, *80*, 046122.

Moreno, J. L. (1934). *Who shall survive? A new approach to the problem of human interrelations* (Vol. 58). Washington: Nervous and Mental Disease Publishing Co.

Newman, M. E. (2001). Clustering and preferential attachment in growing networks. *Physical Review E*, *64*, 025102.

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

Newman, M. E. (2004). Detecting community structure in networks. *The European Physical Journal B*, *38*, 321–330.

Newman, M. E. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America*, *103*, 8577–8582.

Newman, M. E., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, *69*, 026113.

Nicosia, V., Tang, J., Mascolo, C., Musolesi, M., Russo, G., & Latora, V. (2013). Graph metrics for temporal networks. In *Temporal networks* (pp. 15–40). Berlin, Heidelberg: Springer Berlin Heidelberg.

Oliveira, M., & Gama, J. (2012). An overview of social network analysis. *WIREs Data Mining and Knowledge Discovery*, *2*, 99–115.

O'Madadhain, J., Hutchins, J., & Smyth, P. (2005). Prediction and ranking algorithms for event-based network data. *ACM SIGKDD Explorations Newsletter*, *7*, 23–30.

Palla, G., Derényi, I., Farkas, I., & Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. arXiv preprint physics/0506133.

Papadimitriou, A., Symeonidis, P., & Manolopoulos, Y. (2012). Fast and accurate link prediction in social networking systems. *Journal of Systems and Software*, *85*, 2119–2132.

Pereira, F. S. F., Amo, S., & Gama, J. (2016). Evolving centralities in temporal graphs: A Twitter network analysis. In *2016 17th IEEE international conference on mobile data management (MDM)*.

Pereira, F. S. F., Tabassum, S., Amo, S., & Gama, J. (2018). Processing evolving social networks for change detection based on centrality measures. In *Learning from data streams in evolving environments*. Cham: Springer.

Pons, P., & Latapy, M. (2005). Computing communities in large networks using random walks. In *ISCIS* (Vol. 3733, pp. 284–293). Berlin, Heidelberg: Springer.

Porter, M. A., Onnela, J.-P., & Mucha, P. J. (2009). Communities in networks. *Notices of the AMS*, *56*, 1082–1097.

Reed, W. J., & Jorgensen, M. (2004). The double pareto-lognormal distribution-a new parametric model for size distributions. *Communications in Statistics—Theory and Methods*, *33*, 1733–1753.

Richardson, M., & Domingos, P. (2002). Mining knowledge-sharing sites for viral marketing. In *Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 61–70). New York, NY: ACM.

Salton, G., & McGill, M. J. (1986). *Introduction to modern information retrieval*. New York, NY: McGraw-Hill, Inc.

Shetty, J., & Adibi, J. (2004). *The enron email dataset database schema and brief statistical report*. Information Sciences Institute technical report, University of Southern California, *4*, 120–128.

da Silva Soares, P. R., & Prudêncio, R. B. C. (2012). Time series based link prediction. In *The 2012 international joint conference on neural networks (IJCNN)* (pp. 1–7). New York, NY: IEEE.

Spiegel, S., Clausen, J., Albayrak, S., & Kunegis, J. (2011). Link prediction on evolving data using tensor factorization. In *Pacific-Asia conference on knowledge discovery and data mining* (pp. 100–110). Berlin, Heidelberg: Springer.

Sun, J., & Zhu, Y. (2013). Microblogging personalized recommendation based on ego networks. In *2013 IEEE/WIC/ACM International Joint Conferences on Web intelligence (WI) and intelligent agent technologies (IAT)* (Vol. 1, pp. 165–170). New York, NY: IEEE.

Tabassum, S., & Gama, J. (2016a). Evolution analysis of call ego-networks. In *International conference on discovery science* (pp. 213–225). Cham: Springer.

Tabassum, S., & Gama, J. (2016b). Sampling evolving ego-networks with forgetting factor. In *2016 17th IEEE international conference on mobile data management (MDM)* (Vol. 2, pp. 55–59). New York, NY: IEEE.

Tabassum, S., & Gama, J. (2016c). Sampling massive streaming call graphs. In *Proceedings of the 2016 ACM symposium on applied computing, SAC'16* (pp. 923–928). New York, NY: ACM.

Wang, C., Satuluri, V., & Parthasarathy, S. (2007). Local probabilistic models for link prediction. In *Seventh IEEE international conference on data mining (ICDM 2007)* (pp. 322–331). New York, NY: IEEE.

Wang, H., Shi, X., Li, Y., Chang, H., Chen, W., Tang, J., & Martins, E. (2008). User profile management for personalized telecom service. In *The 9th international conference for young computer scientists, 2008. ICYCS 2008* (pp. 1087–1092). New York, NY: IEEE.

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications* (Vol. 8). New York, NY: Cambridge University Press.

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

Wei, C.-P., & Chiu, I.-T. (2002). Turning telecommunications call details to churn prediction: A data mining approach. *Expert Systems with Applications*, *23*, 103–112.

Wellman, B. (1996). Are personal communities local? A dumptarian reconsideration. *Social Networks*, *18*, 347–354.

Wu, H., Cheng, J., Huang, S., Ke, Y., Lu, Y., & Xu, Y. (2014). Path problems in temporal graphs. *Proceedings of the VLDB Endowment*, *7*, 721–732.

Xu, J., & Chen, H. (2005). Criminal network analysis and visualization. *Communications of the ACM*, *48*, 100–107.

Zafarani, R., Abbasi, M. A., & Liu, H. (2014). *Social media mining: An introduction*. New York, NY: Cambridge University Press.