Community Analysis and Link Prediction in Dynamic Social Networks

Blaise Ngonmang, Emmanuel Viennet, Maurice Tchuente, and Vanessa Kamga

Abstract Community detection and link prediction are two well-studied problems in social network analysis. They are interesting because they can be used as building blocks for other more complex problems like network visualisation or social recommendation. Because real networks are subject to constant evolution, these problems have also been extended to dynamic networks. This chapter presents an overview on these two problems.

1 Introduction

Many real-world complex systems can be modelled as networks. A network is a set of entities, called nodes or vertices, connected by links also called edges. The semantics of entities and links depends on the underlying system. For example, in social networks, entities are persons and links are social relationships such as friendship, message exchanges, or collaborations. In power grids, vertices correspond to stations and substations, and edges represent physical transmission lines.

Complex networks generated from real-world systems usually share an interesting property called community structure [15]. A community is a set of nodes having more connections between them than with the rest of the network. These communities can be interpreted as modules and can help to analyse and visualise

B. Ngonmang (✉) • V. Kamga
Université de Paris 13, Sorbonne Paris Cité, L2TI, 93430 Villetaneuse, France

Université de Yaoundé 1, IRD UMI 209 UMMISCO-LIRIMA, Equipe IDASCO, BP 812
Yaoundé, Cameroon
e-mail: blaise.ngonmang@univ-paris13.fr; vansylvania.kamga@lirima.org

E. Viennet
Université de Paris 13, Sorbonne Paris Cité, L2TI, 93430 Villetaneuse, France
e-mail: emmanuel.viennet@univ-paris13.fr

M. Tchuente
Université de Yaoundé 1, IRD UMI 209 UMMISCO-LIRIMA,
Equipe IDASCO, BP 812 Yaoundé, Cameroon
e-mail: maurice.tchuente@lirima.org

A. Gamatié (ed.), *Computing in Research and Development in Africa: Benefits, Trends, Challenges and Solutions*, DOI 10.1007/978-3-319-08239-4_5

the network. Global community detection methods generally assume that the entire structure of the network is known. This assumption is not realistic for very large and dynamic networks. Moreover, these methods usually produce very large communities [16] that are not very useful in practice [40]. For that reason, the concept of local community, i.e. a community obtained by exploring the network starting from a node u_0, is considered. The methods introduced for the identification of local communities do not require to access the entire network, allowing real time processing [10, 31, 39].

Work on social networks has for a long time considered only a static view: a snapshot G_t is taken at a particular time t and is analysed. However, networks are dynamic by nature. New nodes appear and some existing nodes disappear. Similarly, links representing social relations are created or ended. This dynamics can be captured by considering T snapshots $G = (G_1, G_2, \dots, G_T)$ of the network at times $1, 2, \dots, T$. One can then design algorithms to predict links as well as local and global communities that are likely to appear in the next snapshot, G_{T+1}.

The remainder of this chapter is organised as follows: Sect. 2 gives some general definitions and observations on complex networks. Global community detection methods are discussed in Sect. 3. Section 4 then presents some recent methods for the detection of local communities detection and the analysis of their dynamic. Section 5 presents link prediction and some methods defined for this problem. Finally Sect. 6 presents the conclusion and draws some perspectives.

2 Complex Networks Analysis

A social network can be represented by a graph $G = (V, E)$, where V is the set of vertices or nodes, and E is the set of edges or links, formed by pairs of vertices. The two nodes u and v are the end vertices of the edge $e = (u, v)$. If the order of end vertices matters in an edge, then the graph is said to be directed otherwise, it is undirected. Links of directed graphs are denoted $e = (u, v)$. The neighbourhood $\Gamma(u)$ of a node u, is the set of nodes v such that $(u, v) \in E$. The degree of a node u is the number of its neighbours or the cardinality of $\Gamma(u)$, i.e. $degree(u) = |\Gamma(u)|$. The degree of node u will also be denoted by d_u. Given this model, all graph theoretic tools can be reused in network analysis. We recall in the rest of this section the main concepts of graph theory that will be used in this paper.

Hereafter, the number of nodes of the network will be denoted by n and the number of edges will be denoted by m. The adjacency matrix of G is an $n \times n$ boolean matrix A defined by $a_{ij} = 1$ if there is a link from i to j, and $a_{ij} = 0$ otherwise. In some applications, it is useful to model the strength of the link between i and a neighbour j, and a_{ij} is a real number. Such networks are said to be weighted.

The spectrum of a graph G is the set of eigenvalues of its adjacency matrix A. The Laplacian matrix L of a graph G is defined by $L = D - A$, where D is the diagonal matrix of order n defined by $d_{ii} = degree(i)$.

A finite path of length k in G is a sequence of edges $e_1 = (u_1, u_2), e_2 = (u_2, u_3), \dots, e_k = (u_k, u_{k+1})$, such that two consecutive edges $e_i = (u_i, u_{i+1})$

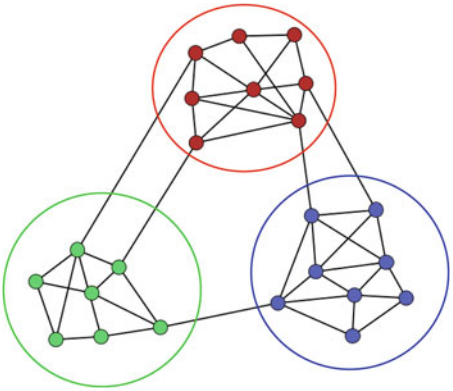

Fig. 1 Example of community structure

and $e_{i+1} = (u_{i+1}, u_{i+2})$ share a common end vertex u_{i+1}. Such a path connects u_1 to u_{k+1}. A path of length k is said to be closed if $u_1 = u_{k+1}$. A connected component of an undirected graph G is a maximal subgraph in which any two vertices are connected to each other by paths. Maximal means that such a component is not connected to any additional vertex in G. A clique is a set of nodes that forms a complete graph, i.e. with all possible links. The term $k - clique$ is used to denote a clique of k nodes.

A link $e = (i, j)$ is internal to a sub-graph $G' = (V', E')$ if i and j are in V'. A link $e = (i, j)$ is external to a sub-graph G' if either i or j is in V', but not both. The density δ of a graph corresponds to the proportion of its existing links compared to the total possible links. The internal density δ_{in} corresponds to the proportion of internal links of a sub-graph compared to the possible internal links. Similarly, the external density δ_{out} corresponds to the proportion of external links of a sub-graph compared to the possible external links.

The clustering coefficient of a network is the number of closed paths of length 3 (or triangles) divided by the number of paths of length 2. It corresponds to the probability that two nodes u and v, connected to a common neighbour w, are also connected.

A community is a set of nodes having a high internal density and a low external density. Figure 1 presents an example of community structure in a network.

It has been observed that many real-world complex networks share some characteristics [5, 35]:

- *Scale-free property:* the degree distribution follows a power law, i.e. the probability that a node has degree k is given by:

$$P(k) = k^{-\gamma} \tag{1}$$

  for a given constant γ usually between 2 and 3.
- *Small world property:* the shortest path between any given pair of nodes is usually small.
- *High clustering coefficient* compared to a random network.

- *Presence of a community structure*. Note, however, that community structure is not always present or easy to detect. This topic is the subject of active research, see, for instance, [8, 26].

The observation of a dynamic network during T time-steps is modelled by $G = (G_1, G_2, \dots, G_T)$, where $G_t = (V_t, E_t)$ is the network observed at time t.

3 Global Communities in Social Networks

Given a network $G = (V, E)$, the global community detection problem can be defined as follows: find a partition $C = \{C_1, C_2, \dots, C_k\}$ of nodes such that the link density is high in each C_i and low between each C_i and the rest of the network. To uncover the community structure in a network, most existing methods translate this (quite informal) definition into a computable *quality function* and then use a greedy algorithm to approximate the optimum of this function and the associated community structure. A very good survey on global community detection can be found in [15]. Note that the former definition is quite restrictive: in some cases, it is interesting to consider *overlapping* communities [21, 43].

3.1 Quality Functions

Quality functions or criteria will be used to assess how good the computed community structure is. Many quality functions have been defined in the literature. Examples are the conductance, the performance, and the modularity [15].

The *conductance* is one of the simplest functions. For a partition $S \cup \overline{S}$, it is defined as the ratio between the number of external links and $min(a(S), a(\overline{S}))$ where $a(C)$ is the total number of links having one end in C. It corresponds to the following expression:

$$\Phi(S) = \frac{|\{(u, v) : u \in S, v \in \overline{S}\}|}{min(\sum_{u \in S} d(u), \sum_{v \in \overline{S}} d(v))} \tag{2}$$

Its values range from 0 to 1. A value close to 0 corresponds to a good partition.

The *performance* [15] is the proportion of pairs of nodes correctly interpreted by the algorithm, i.e. pairs of nodes belonging to the same community and connected with links and pairs of nodes belonging to different communities and not connected. For a community structure $C = (C_1, C_2, \dots, C_n)$, the performance score is given by:

$$P(C) = \frac{|\{(i, j) \in E, C_i = C_j\}| + |(i, j) \notin E, C_i \neq C_j|}{n(n-1)/2} \tag{3}$$

Clearly, $0 \leq P(p) \leq 1$. A value for the performance close to 1 means a good partitioning.

The *modularity* introduced by Girvan and Newman [37] is the most used quality function. The intuition behind this quality function is that a random network is not supposed to have a community structure. For each community, the internal density is compared to the expected internal density in a random network with the same number of nodes and the same degree distribution but without community structure. This corresponds to the formula:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{d_i d_j}{2m}) \delta(C_i, C_j) \tag{4}$$

with m the number of links of the network, A the adjacency matrix, C_i the community of node i. The function $\delta(C_i, C_j)$ is equal to 1 if $C_i = C_j$ and 0 otherwise. The value of Q ranges from -1 to 1. High values of Q are supposed to correspond to partitions with well-separated communities.

Modularity optimisation gives a good way to detect community structures in very large networks. However, this quality function has some drawbacks.

The first drawback is the *instability*. The assumption behind the modularity is that a random network is not supposed to have community structure. The actual community structure is then compared with a *null model* expressed in terms of expectation as presented in Eq. (4). This leads to many possible realisations of the null model and many (very) different community structures with a high modularity, even in random networks [27].

The second drawback is called the *resolution limit*. Indeed, it has been shown in [16] that the size of the community produced by most algorithms depends on the number of nodes in the network. It is then impossible to detect small, even well-separated communities in very large networks.

3.2 Community Detection Methods

Many global community detection methods have been proposed. The main classes of these algorithms are: random walks methods, hierarchical methods, spectral methods.

Random Walks

A random walker on $G = (V, E)$ follows a stochastic process that starts at a node $u_0 \in V$ and at each step i, selects, with probability P_j among its neighbours, the next node j to visit at time $t + 1$ [29]. Usually, this selection is done randomly and uniformly, i.e. $P_j = \frac{a_{ij}}{d_i}$. The length of a random walk is its number of steps. The transition matrix of the random walk is $P = AD^{-1}$, and the probability of going from a vertex i to a vertex j in t steps is $(P^t)_{ij}$.

The idea behind random walk methods is that a random walker on G tends to get trapped into communities. As a consequence, two vertices i and j of the same community tend to see all the other vertices in the same way, i.e. if the length t of the random walk is long enough, the i^{th} row $(P^t)_{i.}$ and the j^{th} row $(P^t)_{j.}$ will be similar. This leads to a definition of distance between nodes and community detection becomes a clustering problem that can be solved using, for example, hierarchical methods [45].

Hierarchical Methods

Hierarchical methods are either top down or bottom up. In top down methods, one starts with all nodes in one unique community. At each step, one tries to separate the existing communities into sub-communities. An example of method ranging in this category is described in [37]. In bottom up methods, one starts with each node belonging to a separate community and at each step, one tries to merge the most similar communities.

Louvain [7] is one of the fastest methods for community detection in complex networks. It can be used in the general case of weighted networks (links have weights expressing the strength of relationships). The algorithm is a bottom up hierarchical method and has two main steps. In the first step, every node of the network is evaluated by computing the weighted modularity gain if it is then added to the community of a neighbour j. Node i is added to the community which produces the maximal positive gain. This process is repeated until there is no more positive gain. At the end of this step one has a partition of the network.

In the second step, each of the previous discovered communities becomes a *super-node*. The weight of the link between two *super-nodes* is the sum of the weights of links between nodes of the corresponding communities. Links between nodes of the same community lead to a self-loop of the corresponding *super-node*.

These two steps are repeated until no more gain in modularity is observed. The speed of this method comes from the observation that the modularity can be optimised locally: only the neighbours are considered during the evaluation. This enables to update the modularity gain in linear time ($O(m)$). This method is only limited by the storage (main memory) capacity and allows to analyse networks with millions of nodes in few minutes.

Relation with Spectral Methods

Random walks on graphs are strongly related to methods that study community structure using spectral properties of graphs. For instance, in [45] it is shown that the distance r_{ij} defined between two nodes i and j, using random walks, is related to the spectral property of the matrix P by the formula:

$$r_{ij}^2 = \sum_{\alpha=2}^{n} \lambda_\alpha^{2t} (v_\alpha(i) - v_\alpha(j))^2 \tag{5}$$

where $(\lambda_\alpha)_{1\leq\alpha\leq n}$ and $(v_\alpha)_{1\leq\alpha\leq n}$, are respectively, the eigenvalues and right eigenvectors.

On the other hand, it is shown in [49] that if two vertices i and j belong to the same community, then the coordinates v_i and v_j are similar in eigenvectors of P that correspond to the largest positive eigenvalues. More recently, Newman has shown that community detection by *modularity maximisation*, community detection by *statistical inference*, and *normalised-cut* graph partitioning can be solved using spectral approaches that use matrix representations of the network [36].

3.3 Dynamics of Global Communities

For community detection in dynamic networks, various methods have been proposed in the literature. Some of these methods try to track the evolution of communities between time-steps, others try to update the existing community structure and finally, the last class of methods try to detect communities that are consistent in all the time steps.

Community Tracking

Algorithms for global community detection in static networks have been used for tracking community structures in dynamic networks. The idea is to detect the communities at each time-step and match them between consecutive time-steps. Examples of such methods were proposed by Palla et al. [44], Greene et al. [20], and Tantipathananandh et al. [50].

The algorithm by Palla et al. is based on the Clique Percolation Method [43]. This approach is not usable in very large graphs. The two other methods can be used with any algorithm and particularly with modularity-based algorithms.

The tracking of communities is very difficult because of the instability of global methods. This is particularly true for modularity-based methods. This drawback can be reduced by using community cores analysis. In [46] and [14] a community core is defined as a set of node that are frequently in the same community during many executions of an unstable algorithm.

Community Updating

The main idea of community updating is to detect the community structure at a reference time-step t_0 and then, for all the future time-steps, the community structure is updated according to the elementary events that can appear in the network. These elementary events are: addition of a node, deletion of a node, creation of a link, and removal of a link. Examples of methods that have been proposed to handle these events can be found in [9, 42].

The main problem with community updating is that it is sensible to the initial partition. Note that community core analysis can help to stabilise the initial partition.

Long-Term Community Detection

A long-term community can be defined as a set of nodes that interact more with each other than with the rest of the network at all time-steps [4]. This is an extension of the classical community definition to the dynamic context. To detect long-term communities, Aynaud and Guillaume [4] have proposed to define an average modularity and to optimise it using a modified version of Louvain's algorithm [7]. In another work, Mitra et al. [33] have proposed a method designed for citation networks. They propose to build a summary network as follows:

- a node a_t is created if author a has published a paper at time t.
- a link is created between nodes a_t and $b_{t'}$ if and only if the paper published by author a at time t cites the paper published by author b at time t'.

A static community detection method can then be used to mine the community structure in this summary network.

4 Dynamics of Local Communities

Global communities give a way to analyse the dynamics at a macroscopic level. Because global communities are either too large or cannot be computed due to the size of the network, we show in this section how to analyse the dynamics using local communities that can be computed without the entire knowledge of the network.

4.1 Local Community Identification

Given a node u_0 of a partially known network $G = (V, E)$, initially limited to u_0 and its neighbours, and with the restriction that new information can only be obtained by getting adjacent nodes one by one, the problem of local community identification is to find the community the node u_0 belongs to.

Table 1 Some quality functions for local community identification

Quality functions	Authors
$R = \frac{B_{in}}{B_{in}+B_{out}}$	Clauset [11]
$M = \frac{D_{in}}{D_{out}}$	Luo et al. [31]
$L = \frac{\frac{\sum_{i \in D} \|\Gamma(i) \cap D\|}{\|D\|}}{\frac{\sum_{i \in B} \|\Gamma(i) \cap S\|}{\|B\|}}$	Chen et al. [10]
$T = \frac{\sum_{i \in D} \frac{\|\Gamma(i) \cap D\|}{(1+d_i)}}{\sum_{i \in D} \|\Gamma(i) \cap S\|(1+d_i)}$	Ngonmang et al. [39]

Most existing algorithms for local community identification use a greedy scheme: initially, the local community D contains only the starting node u_0 and the quality of this initial community is 0. At each step, the external node that maximises the quality function F used by the algorithm is considered. If its inclusion into D increases the quality criterion F, then it is added to D, and the quality F of the community is updated. This procedure is repeated until there is no more external vertex whose inclusion into D increases the quality F. At the end of the algorithm, D contains the local community of u_0.

Let D denote a local community. B is the set of nodes of D that have at least one neighbour out of D and S is the set of nodes out of D that have at least one neighbour in D. Table 1 presents some existing quality functions used in local community identification. In this table, D_{in} corresponds to the set of links having both ends in D. D_{out} corresponds to the set of links having only one end into D. B_{in} and B_{out} have a similar meaning. d_i is the distance from i to u_0, the starting node, and $\Gamma(i)$ is the set of neighbours of node i.

Some problems are similar to local community identification because they only require local information. One of these problems is ego-community detection. This problem consists in detecting the communities between the direct neighbours of a node. One successful approach to solve this problem is the method proposed in [19]. Unfortunately, this method is not suitable for local community detection because it discards the rest of the network. Another similar problem is to consider multiple starting nodes and detect the communities that contain them, as in the work of Danisch et al. [12].

4.2 Application to a Dynamic Behaviour: Churn Prediction

The objective of churn prediction is to estimate the likelihood that a given user will stop using a social network platform in the near future. A churner will thus be defined as a user who has become inactive for a certain period of time. This knowledge can be exploited by the platform operator to take preventive actions: if the user is likely to stop using the platform, it could be interesting to send him some incentives (personalised recommendations, free applications, etc.).

Most of the methods for churn prediction belong to three main categories: feature-based methods, network-based methods, and hybrid methods. Feature-based methods extract attributes from the user profile (age, gender, etc.) and usage (time spent, connexion history, etc.) of the platform and then build a predictive model [24].

Network-based methods use the social links to detect the churners. The methods of this category usually model the churn prediction as a diffusion or contagion process [13]: starting with the known churners as seeds, each seed tries to activate its neighbours at each iteration. This process is repeated until convergence or up to a maximum number of iterations.

Finally hybrid methods combine the two previous ones. One application of local community analysis can be found in this category. Indeed, the hybrid method proposed in [40], for example, proposes to extract some attributes from the local community of the node and to add them to the features of the node in order to build the churn prediction model.

4.3 Prediction of Local Communities

The local community prediction problem in complex networks can be stated as follows: given a dynamic network $G = (G_1, \dots, G_T)$ and the dynamic local community $D = (D_1, \dots, D_T)$ of a node u_0, what will be the local community D_{T+1} at time $T + 1$ in (G_{T+1})?

To solve this problem, two main classes of approaches can be used. The first class consists in predicting for each node v, whether or not it will belong to the local community D_{T+1} of node u_0 at time $T + 1$. The second class first predicts the structure of the network at time $T + 1$ and then computes the local community of u_0 in the predicted network.

To predict the membership of each node u to D_{T+1}, some simple attributes can be computed at each time-step: the position of u with respect to the subsets D, B, and S defined in Sect. 4.1, the number of links with the community, the number of links with nodes outside the community, etc. A supervised learning model can then be used for the prediction. The real challenge here is that one is restricted to nodes having already belonged to the local community of u_0 or to its neighbourhood.

The work in [41], for example, presents a method to predict local communities according to the second approach. The network of the target time-step is predicted and the local communities are computed on that predicted network. To keep the locality on this process, only the local portion of the network containing the starting node is predicted.

This gives a way to locally analyse a dynamic network and make some predictions on its future structure. A more microscopic view of the dynamic is provided by link prediction that will be presented in the next section.

5 Link Prediction

Given a snapshot of a social network at time t, the link prediction problem is to accurately predict the edges that will be added to the network from time t to a given future time t'. The link prediction problem therefore tackles the following question: to what extent can the evolution of a social network be modelled using features intrinsic to the network itself [28]? Formally, consider a network $G = (V, E)$ where V is the set of vertices and E is the set of links. The set of edges $(u, v) \subseteq V$ with $u \neq v$ that are absent in E is denoted $\overline{E}$. In a practical application, $\overline{E}$ can be divided into two parts: the set E' of links that will appear in the future, also called missing links, and the set E'' of edges that will never appear. Clearly, $E' \cup E'' = \overline{E}$ and $E' \cap E'' = \emptyset$. The challenge of link prediction is to produce quickly, accurate approximations for E', even for huge social networks.

As noted by Zhu and Kinzel [53], it is possible, for any discrete prediction algorithm A for sequences, to generate using an algorithm B no more complicated than A, an instance for which A's prediction is always wrong. Moreover, for any prediction algorithm A and an instance x, there exists a sequence y no more complicated than x, such that if A performs better than random on x, then it will perform worse than random on y by the same margin. This shows that, to design a predictor with good performance, it is necessary to have prior knowledge on the problem.

Link prediction is a very active research area because of its wide range of applications. For instance, if G is a social network representing recorded interactions between terrorists, the link prediction can be used to detect underground relationships between them. On the other hand, a link prediction algorithm can be applied to a clients/products network produced by an e-commerce platform, to suggest products that a client is likely to purchase in the near future. Other algorithms and applications related to link prediction in complex networks can be found in [30].

We now present some basic link prediction algorithms according to the following nomenclature: probabilistic methods, transitivity-based methods, and attributes-based methods. After that we introduce an extension of the link prediction problem to dynamic networks.

5.1 Probabilistic Methods

The most naive probabilistic model of link prediction is the Random predictor which randomly chooses a subset of links that are not present in the network and predicts them. Since the subset selection is done randomly, the accuracy of the algorithm is based on luck. The probability of failure of the Random predictor is $\frac{1}{2}$. This method can't be taken seriously when dealing with an application. It only serves as reference point: any serious algorithm must have a better accuracy.

The probabilistic approaches can nevertheless be useful when there is a prior knowledge on the problem. For instance, many complex natural and social systems assemble and evolve through the addition and removal of nodes and links. This dynamics often appears to be a self-organising mechanism governed by evolutionary laws that lead to some common topological features. One of such features is the power-law degree distribution, i.e. the probability that a node has degree k is $P(k) = k^{-\gamma}$, usually with $2 < \gamma \leq 3$. Such networks are said to be scale-free. For such networks, the "preferential attachment principle" states as follows: when a new node is added to the network with m edges that link this new node to m nodes already present, the probability that this new node will connect to a node i with degree d_i is proportional to d_i, i.e. $\pi(d_i) = \frac{d_i}{(\sum_i d_i)}$. It can be shown that a network evolving according to this principle tends to a scale-invariant state with $\gamma = 3$. Clearly, such a model of network growth constitutes an a priori information that can help to design efficient link prediction algorithms. The preferential attachment principle is also known in economy as cumulative advantage: the rich get richer [6, 48].

The preferential attachment is a good illustration of Zhu and Kinzel's observation. Indeed, it gives the worst performance when applied to physical Internet networks where high degree nodes are routers that have a very low probability of being connected by new physical lines.

Recently, Freno et al. [17] have proposed a new approach that is not based on parametric assumptions concerning the modelled distributions. More precisely, they have introduced the Fielder random field model, called Fielder delta statistic that, for each binary edge variable $X_{u,v}$, defines a potential that encapsulates the measure of its role in determining the connectivity of its neighbourhood. The trick is that these potentials can be estimated from data by minimising a suitable objective function. Experiments on some real-world networks have resulted in link prediction algorithms that outperform the solutions proposed by Watts-Strogatz [51] and Barabasi-Albert [6]. Other probabilistic methods for link prediction are reported in [30].

5.2 Transitivity-Based Methods

In mathematics, a binary relation $\Re$ defined on a domain D is said to be transitive if whenever u is in relation with $v(u \Re v)$ and v is in relation with $w(v \Re w)$, then u is in relation with $w(u \Re w)$.

In topological transitivity applied to a complex network $G = (V, E)$, the domain D consists of the set V of nodes of the network, and the relation $\Re$ is represented by the set E of edges. The application of topological transitivity to link prediction is based on the assumption that, as a complex network evolves, it tends to become transitive, i.e.: if at time t, u is related to v and v is related to w, then there is a high probability that at a future time t', u will be related to w. This assumption follows from a common observation made, for instance, on friendship networks: a friend of your friend is likely to be or become your friend. This corresponds to triangles in G, i.e. triples of edges (u, v), (v, w), and (u, w). In graph-theoretic terms, the degree of transitivity of a network G can be measured by the so-called clustering coefficient [51]:

$$C = \frac{\sum_{u \in V} C_u}{|V|} \tag{6}$$

where

$$C_u = \frac{\textit{number of triangles connected to vertex u}}{\textit{number of triples centred on vertex u}} \tag{7}$$

As reported in [35], this coefficient has remarkable values for many current networks: greater than 0.75 for film actors and power grids; between 0.6 and 0.74 for biology co-authorship, train routes and metabolic networks; between 0.30 and 0.59 for math co-authorship, Internet and word co-occurrences in web pages, and less than 0.20 for email messages and freshwater food web.

The basic link prediction methods based on topological transitivity use some local or global properties of the network G, to assign a connection weight $Score(u, v)$, to pairs of nodes (u, v) of V. All non-observed links are then ranked in decreasing order of $Score(u, v)$. In this approach, links with the highest scores are supposed to be of higher existence likelihoods and are produced by the algorithm. Such a measure must reflect the proximity or similarity between nodes u and v. The problem is to design good similarity measures.

Let us denote $\Gamma(u)$ the set of neighbours of node u, and let $|A|$ be the cardinality of a set A. $CN(u, v) = |\Gamma(u) \cap \Gamma(v)|$ [34] corresponds to the number of common neighbours of u and v. The idea is that if u and v have many neighbours in common, then there is a high probability that they will become neighbours in the future. The efficiency of this measure has been experienced with collaborative networks [35]. However, this measure suffers from serious drawbacks. For instance, in a friendship network, the fact that two nodes u and w have a common very popular neighbour v, i.e. with a very high degree d_v, does not necessarily mean that u and w will become friends in the future. They may even be from different continents and never meet. In the same way, in an allocation network, if node u sends a unit of resource to a very popular neighbour v that serves as intermediary, and if node v subdivides the resource and sends equal parts to his neighbours, then the portion received by any neighbour $w \in \Gamma(v)$ will be $\frac{1}{d_v}$. This means that the contribution of an intermediate node v for the "future connection" between u and w is divided by the degree of v. This has motived some authors to introduce $RA(u, v) = \sum_{w \in \Gamma(u) \cap \Gamma(v)} \frac{1}{d_w}$ [52] and the log form $AA(u, v) = \sum_{w \in \Gamma(u) \cap \Gamma(v)} \frac{1}{\log(d_w)}$ [1]. Many other variants of CN have been proposed, but extensive experiments on real-world networks have shown that RA is the best whereas CN is the second best.

A nice link prediction method based on topological transitivity has been introduced by Latapy et al. [2]. Consider a bipartite clients/products network

$G = (\bot, \top, E)$ where $\bot$ is the set of clients, $\top$ the set of products, and E the set of purchases. The $\bot$-projection of G is the graph $G_\bot = (\bot, E_\bot)$ in which $(u, v) \in E_\bot$ if u and v have at least s neighbours in common in G, where s is a given threshold, i.e. $|\Gamma(u) \cap \Gamma(v)| \geq s$. The underlying intuition of the internal link prediction method is that, in a clients/products network, if two clients have bought in the past many common products, then they will probably acquire new common products in the future. This method falls within the transitivity framework as follows: if client A is related to client B in $G_\bot$ and if client B is related to product p in G, then there is high probability for A to be related to p in the future.

Another topological transitivity measure for link prediction is based on random walks already introduced in section "Random Walks" for community detection. In the simplest version of this method, it is assumed that, when a random walker is at node u, it can go in one step to any node $v \in \Gamma(u)$ with probability $\frac{1}{d_u}$. Let $m(u, v)$ denote the average number of steps necessary for a random walker to go from u to v. The commute time is the symmetrical measure $CT(u, v) = m(u, v) + m(v, u)$. This transitivity measure is then used to predict missing links: the smaller $CT(u, v)$ is, the greater is the probability for u and v to establish a connection in the future.

Association rules originally defined for large databases of sales transactions can be adapted for link prediction on a network $G = (V, E)$. Consider $D = \{\Gamma(u) | u \in V\}$. Define frequent groups of nodes as subsets that are included in at least s elements of D, where s is a given threshold. An association rule is an implication of the form $A \to B$, where $A \subseteq V$, $B \subseteq V$, $A \cap B = \emptyset$, and $A \cup B$ is frequent. A rule $A \to B$ holds with confidence c if $c\%$ of neighbourhoods in D that contain A also contain B. Hereafter, we denote $A \to B : c$. The transitivity principle states as follows: if $A \to B : c$ and $B \to C : c'$, then $A \to C : c \times c'$. In the context of an application to co-authorship [25]: A, B and C are sets of co-authors. $A \to B : c$ means that $c\%$ of articles co-authored by A are also co-authored by B, and $B \to C : c'$ means that $c'\%$ of articles co-authored by B are also co-authored by C. As a consequence, if $A \to B : c$ and $B \to C : c'$ are observed, then $A \to C$ is predicted with probability $c \times c'$ (i.e. a new article with $A \cup C$ as co-authors is predicted) [25].

5.3 Attributes-Based Methods

The great specificity for graphs that model social networks is that nodes and links usually have attributes. Consider a phone network in which a node represents a person and each link represents a call. Phone numbers can be used as node attributes and the average number of calls between nodes can be used as link attributes.

The link prediction problem can be expressed as a classification problem for pairs (u, v). The following attributes may be considered when dealing with co-authorship networks [23]: the number of common neighbours $(\Gamma(u) \cap \Gamma(v))$, the number of common keywords $(Kw(u) \cap Kw(v))$, or the total number of articles published by u and v. The class attribute is a binary variable with value 1 if the link will appear and 0 otherwise. All attributes values are normalised to have zero mean and one standard deviation. A classification model such as Decisions Tree (DT), Support Vector Machine (SVM) or Artificial Neural Network (ANN) can then be used. Hasan et al. [23] have shown on two networks (DBPL and BIOBASE) that SVM beats all the most used classification methods.

The similarity between two nodes can use attributes of nodes and links. This is the case for *Abstract* proposed in [25], which takes into account summaries of articles in the bipartite graph Authors/Articles. The idea is that articles already published contain information on topics that interest the co-authors. It is then natural to suppose that authors working in the same domain are more likely to collaborate and co-publish an article in the future. The attributes-based similarity between two u and v authors is then defined as:

$$score(u, v) = cos(V(u), V(v)) \tag{8}$$

where $V(u)$ is a descriptor that encapsulates the attributes for vertex u. It has been shown in [25] that this approach produces very good predictions for some well-known co-authorship networks.

5.4 Link Prediction in Dynamic Complex Networks

Classical link prediction is not sufficient to fully analyse the dynamics of a complex network. Indeed, it supposes that all the created links will last forever. However, in a real interaction complex network, the links between nodes are created and ended. For example, in a collaboration network, a publication between two scientists in a particular year does not guarantee that they will still work together in the future. To fully analyse the dynamics of connections between nodes, one needs a more general model which determines whether or not a particular link will exist at a future time t'.

This can be stated as follows: given a dynamic network $G = (G_1, \ldots, G_T)$, what will be the structure of the following snapshot (G_{T+1})? This problem is a generalisation of the link prediction problem: here not only the non-previously seen links are predicted but also the existing ones to check whether or not they will still exist in the following snapshot.

As a generalisation of the link prediction problem, the class of methods designed for the classical link prediction problem can also be used to solve it. A similarity-based and a supervised learning method to solve this problem can be found in [38]. The supervised method can be described as follows: for each snapshot t of the training period, the following features are computed for each pair of nodes:

- the number of common neighbours
- the number of common community members
- a boolean attribute indicating whether an interaction is present or not between the two nodes
- other similarity between the two nodes (if available)

The real classes are obtained on the test period. It is worth nothing that to reduce the complexity (the number of possible interactions is in the order of $O(n^2)$), only the interactions that are likely to appear, based on the computed similarity scores (topological or attribute based), are considered. Finally, a supervised learning method (SVMs for example) can be used to build the model.

6 Conclusions and Perspectives

In this chapter, we have presented some tools for the analysis of the community structure and for link prediction in social networks in both static and dynamic contexts. This corresponds to the observation of the evolution of such a network at the macroscopic level (global communities), intermediate level (local communities), and microscopic (link prediction) level. Such analysis, and more generally social mining techniques, can lead to many applications in Africa and some interesting work has already been done.

In epidemiology, for example, the work in [18] presents an agent-based model of epidemic spread using social networks. On the other hand, because of the dense interaction patterns, infectious diseases tend to spread more rapidly in communities. As a consequence, an interesting question may be to design models of infection spread, that take into account the community structure of a contact network. An example of attempt in this direction can be found in [3] where the Ross–Macdonald model which describes the dynamics of malaria has been extended to multipatch systems. In this approach, a patch models a community. On the other hand, a nice and simple analytic formula of the basic reproduction number has been proposed in [47] for cellular SIR networks.

In telecommunication networks, thanks to the Data for Development challenge http://www.fr.d4d.orange.com/, many researches have been conducted on the Ivory Coast telecommunication network [6]. For example, the work in [20] presents a method based on the phone calls network and airtime credit, for the evaluation of the socioeconomic state of a country. More recently in a Ph.D. thesis, Guigourès [22] has studied the co-clustering technique that consists of simultaneously partitioning the rows and the columns of a data matrix. Applications to detailed call records from a telecom operator in Ivory Coast have permitted to detect individuals that are the most representative of their profiles. This information was then used to improve the knowledge of users, develop new products and improve urban infrastructure related to mobility.

For co-authorship networks, the work in [32] analyses the evolution of co-publications in the community of researchers involved since 1992 in the African conference for research in applied mathematics and computer science (CARI).

Note that, for Africa to take full benefit of tools for network analysis, an increased effort must be put on data collection.

References

1. L.A. Adamic, E. Adar, Friends and neighbors on the web. Soc. Netw. **25**, 211–230 (2003)
2. O. Allali, C. Magnien, M. Latapy, Link prediction in bipartite graphs using internal links and weighted projection, in *Proceedings of the Third International Workshop on Network Science for Communication Networks (NetSci- Com)* (2011)
3. P. Auger, E. Kouokam, G. Sallet, M. Tchuente, B. Tsanou, The Ross–Macdonald model in a patchy environment. Math. Biosci. **216**(2), 123–131 (2008)
4. T. Aynaud, J.L. Guillaume, Static community detection algorithms for evolving networks, in *WiOpt'10: Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, Avignon (2010), pp. 508–514. http://hal.inria.fr/inria-00492058
5. A.L. Barabasi, *Linked: How Everything is Connected to Everything Else and What It Means*, reissue edn. Plume, (2003)
6. A.L. Barabási, R. Albert, Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
7. V.D. Blondel, J. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. **10** 10008 (2008)
8. R. Campigotto, J.L. Guillaume, M. Seifi, The power of consensus: random graphs have no communities, in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (ACM, New York, 2013), pp. 272–276
9. R. Cazabet, F. Amblard, Simulate to detect: a multi-agent system for community detection. IAT, 402–408. IEEE Computer Society (2011)
10. J. Chen, O.R. Zaiane, R. Goebel, Local communities identification in social networks, in *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'09)* (2009), pp. 237–242
11. A. Clauset, Finding local community structure in networks. Phys. Rev. **72**, 026132 (2005)
12. M. Danisch, J.L. Guillaume, B.L. Grand, Towards multi-ego-centred communities: a node similarity approach. J. Web Based Communities **9**(3), 299–322 (2013)
13. K. Dasgupta, R. Singh, B. Viswanathan, D. Chakraborty, S. Mukherjea, A. Nanavati, A. Joshi, Social ties and their relevance to churn in mobile telecom networks, in *EDBT '08: Proceedings of the 11th International Conference on Extending Database Technology* (2008), pp. 668–677
14. V. De Leo, G. Santoboni, F. Cerina, M. Mureddu, L. Secchi, A. Chessa, Community core detection in transportation networks. Phys. Rev. E **88**(4), 042810 (2013)
15. S. Fortunato, Community detection in graphs. Phys. Rep. **486**, 75–174 (2010)
16. S. Fortunato, M. Barthélemy, Resolution limit in community detection. Proc. Natl. Acad. Sci. **104**(1), 36–41 (2007)
17. A. Freno, C. Garriga Gemma, M. Keller, Learning to recommend links using graph structure and node content, in *Neural Information Processing Systems Workshop on Choice Models and Preference Learning* (2011)
18. E. Frías-Martínez, G. Williamson, V. Frías-Martínez, An agent-based model of epidemic spread using human mobility and social network information, in *SocialCom/PASSAT* (IEEE), pp. 57–64
19. A. Friggeri, G. Chelius, E. Fleury, Egomunities, exploring socially cohesive person-based communities. CoRR. abs/1102.2623 (2011)
20. D. Greene, D. Doyle, P. Cunningham, Tracking the evolution of communities in dynamic social networks, in *Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining, ASONAM '10* (IEEE Computer Society, Washington, 2010), pp. 176–183
21. S. Gregory, Finding overlapping communities using disjoint community detection algorithms, in *Complex Networks* (2009), pp. 47–61
22. R. Guigourès, Utilisation des modèles de co-clustering pour l'analyse exploratoire des données. Ph.D. thesis in Applied mathematics, University of Paris 1 Panthéon Sorbonne
23. M.A. Hasan, V. Chaoji, S. Salem, M. Zaki, Link prediction using supervised learning, in *Proceedings of SDM 06 Workshop on Link Analysis, Counterterrorism and Security* (2006)

24. H. Hwang, T. Jung, E. Suh, An ltv model and customer segmentation based on customer value: a case study on the wireless telecommunication industry. Expert Syst. Appl. **26**(2), 181–188 (2004)
25. V. Kamga, M. Tchuente, E. Viennet, Prévision de liens dans les graphes bipartites avec attributs. Revue des Nouvelles Technologies de l'Information (RNTI-A6) (2013)
26. I. Keller, E. Viennet, A characterization of the modular structure of complex networks based on consensual communities, in *2012 Eighth International Conference on Signal Image Technology and Internet Based Systems (SITIS)* (IEEE, Naples, Italy 2012), pp. 717–724
27. I. Keller, E. Viennet, A characterization of the modular structure of complex networks based on consensual communities, in *2013 International Conference on Signal-Image Technology Internet-Based Systems (SITIS)* (2012), pp. 717–724
28. D. Liben-Nowell, J. Kleinberg, The link prediction problem for social networks, in *Proceedings of the Twelfth International Conference on Information and Knowledge Management, CIKM '03* (ACM, New York, 2003), pp. 556–559
29. L. Lovász, Random walks on graphs: a survey. In *Combinatorics, Paul Erdös is Eighty*, eds. by D. Miklás, V.T. Sás, T. Szönyi (János Bolyai Mathematical Society, Budapest, 1996), pp. 353–398
30. L. Lu, T. Zhou, Link prediction in complex networks: a survey. Phys. A Stat. Mech. Appl. **390**(6), 1150–1170 (2011)
31. F. Luo, J.Z. Wang, E. Promislow, Exploring local community structure in large networks, in *WI'06* (2006), pp. 233–239
32. G.R. Meleu, P. Melatagia, Analyse et modélisation du cari: croissance de la communauté de chercheurs du cari, in *Conférence de Recheche en Informatique(CRI'2013), Yaoundé* (2013), pp. 83–87
33. B. Mitra, L. Tabourier, C. Roth, Intrinsically dynamic network communities. CoRR. abs/1111.2018 (2011)
34. M.E.J. Newman, Clustering and preferential attachment in growing networks. Phys. Rev. E **64**, 025102 (2001)
35. M.E.J. Newman, The structure and function of complex networks. SIAM Rev. **45**, 167–256 (2003)
36. M.E.J. Newman, Spectral methods for network community detection and graph partitioning. Phys. Rev. **884**, 042822 (2013)
37. M. Newman, M. Girvan, Community structure in social and biological networks. Proc. Natl. Acad. Sci **99**, 7821–7826 (2002)
38. B. Ngonmang, E. Viennet, Toward community dynamic through interactions prediction in complex networks, in *2013 International Conference on Signal-Image Technology Internet-Based Systems (SITIS)* (2013), pp. 462–469
39. B. Ngonmang, M. Tchuente, E. Viennet, Local communities identification in social networks. Parallel Process. Lett. **22**(1) (2012)
40. B. Ngonmang, E. Viennet, M. Tchuente, Churn prediction in a real online social network using local community analysis, in *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'12)* (2012), pp. 282–290
41. B. Ngonmang, S. Sean, R. Kirche, Monetization and services on a real online social network using social network analysis, in *2013 IEEE 13th International Conference on Data Mining Workshops* (2013), pp. 185–193
42. N. Nguyen, T. Dinh, Y. Xuan, M. Thai, Adaptive algorithms for detecting community structure in dynamic social networks, in *2011 Proceedings IEEE INFOCOM* (2011), pp. 2282–2290
43. G. Palla, I. Derényi, T. Vicsek, Uncovering the overlapping community structure of complex networks in nature and society. Nature **435**, 814–818 (2005)
44. G. Palla, A.-L. Barabási, T. Vicsek, Quantifying social group evolution. Nature **446**(7136), 664–667 (2007)
45. P. Pons, M. Latapy, Computing communities in large networks using random walks. J. Graph Algorithms Appl. **10**(2), 191–218 (2006)

46. M. Seifi, I. Junier, J.B. Rouquier, S. Iskrov, J.L. Guillaume, Stable community cores in complex networks, in *Complex Networks*. Studies in Computational Intelligence, vol. 424 (Springer, Berlin/Heidelberg, 2013), pp. 87–98
47. A. Sidiki, M. Tchuente, An analytical formula for the basic reproduction number on cellular sir networks, in *Actes du Colloque Africain de Recherche en Informatique*, (2012)
48. H.A. Simon, On a class of skew distribution functions. Biometrika **42**, 198–216 (1955)
49. I. Simonsen, K.A. Eriksen, S. Maslov, K. Sneppen, Diffusion on complex networks: a way to probe their large-scale topological structures. Phys. A Stat. Theor. Phys. **336**(1–2), 163–173 (2004)
50. C. Tantipathananandh, T. Berger-Wolf, D. Kempe, A framework for community identification in dynamic social networks, in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '07* (ACM, New York, 2007), pp. 717–726
51. D. Watts, S. Strogatz, Collective dynamics of 'small-world' networks. Nature **393**, 440–442 (1998)
52. T. Zhou, L. Lü, Y.C. Zhang, Predicting missing links via local information. Eur. Phys. J. B **71**, 623 (2009)
53. H. Zhu, W. Kinzel, Antipredictable sequences: harder to predict than random sequences. Neural Comput. **10**, 2219–2230 (1998)