Survey on Social Community Detection

Michel Plantié and Michel Crampes

Abstract

Community detection is a growing field of interest in the area of Social Network applications. Many community detection methods and surveys have been introduced in recent years, with each such method being classified according to its algorithm type. This chapter presents an original survey on this topic, featuring a new approach based on both semantics and type of output. Semantics opens up new perspectives and allows interpreting high-order social relations. A special focus is also given to community evaluation since this step becomes important in social data mining.

1 Introduction

As social networks gain prominence, the first obvious question that comes to a researcher's mind in observing these networks is: how to extract meaningful knowledge from these data? In seeking a response, the network structure proves to be of utmost importance. Identifying high-order structures within networks yields insights into their functional organization, which in turn contributes more knowledge while offering many possible actions, including marketing plans, recommendations and user interface adaptations. Community detection may become a more complicated task given that social networks can be structured on many different levels, yet communities reduce the complexity of a network's original graph in a substantial way, thus revealing its macro-structure and uncovering more semantic knowledge. A growing number of community detection methods have recently been published. The goal here is to assess the state-of-the-art in this area, by focusing on the qualities

Laboratoire de Genie Informatique et d'Ingenierie de Production (LGI2P), {michel.plantie,michel.crampes}@mines-ales.fr EMA - Ecole des Mines, Site EERIE, Parc Scientifique Georges Besse, F-30035 Nîmes cedex 1 - France

and shortcomings of each method. A number of partial surveys have been conducted over the past few years; though this body of work has exposed different approaches in the field, such efforts are often limited to specific network structures. This chapter is intended to present three analytical approaches to community detection that encompass most of the main methods and techniques. The first approach, which is also the most widespread, considers the social network as a graph and then analyzes its structure with graph properties and algorithms built around the graph structure. The second approach associates the social network with a hypergraph and analyzes its structure through hypergraph properties and algorithms based on hypergraph structures, as exemplified in [54]. The third and final approach uses the properties of concept lattices in order to analyze the social network structure in association with hypergraph properties and algorithms based on Galois lattices and hypergraph structures, e.g. [55, 60]. As opposed to graphs, both hypergraphs and Galois lattices have been poorly analyzed in surveys on community detection strategies. These structures offer very efficient tools for managing communities and this discussion will demonstrate how researchers have applied them. The chapter will be organized as follows. To ensure a good understanding of all elements being addressed in this survey, Section 2 will give all necessary definitions, section 3 will then deliver a state-of-the-art from previous surveys on the community detection topic. The next section will classify each community detection method according to a graph type of classification, and lastly Section 5 will lend insight into all possible evaluations of community detection algorithms, as this area of investigation has only been sparsely studied in previous surveys.

2 Definitions: Social Network Community detection and other Definitions

2.1 Unipartite and Bipartite graphs

A graph is a representation of a set of objects called vertices, some of which are connected by links. Object connections are depicted by links, also called edges. Such a mathematical structure may be referred to as a unipartite graph. A good example of this type of graph is the well-known Zakary Karate club [77] (shown figure 1).

A special case of this graph is known as the bipartite graph, i.e. whose vertices can be divided into two disjoint sets A and B such that the edges only connect one vertex in A to one in B, in considering that A and B are independent sets. Vertices of A are not connected to any other vertices within A, and the same applies for B. For example, let A be a set of individuals and

B a set of photos showing these same individuals. Bipartite graphs may take the form of graphs, hypergraphs or Galois lattices.

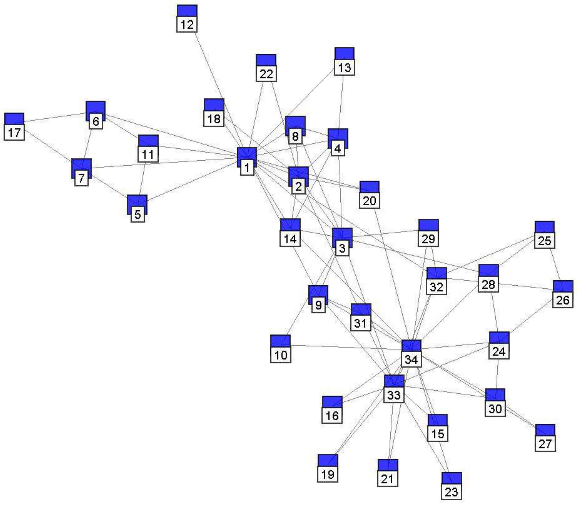

Fig. 1 Depiction of the Zachary Karate Club graph example (with a different display for better visibility)

2.2 Hypergraph

A hypergraph [4] H is a pair (V, E) where V = v1, v2, ..., vn is a non-empty (usually limited) set and E = E1, E2, ..., Em is a family of not empty subsets of V. The elements of V are the vertices of H. The elements of E are the edges (also called hyperedges) of H. A set of social communities can be viewed as a hypergraph whose vertices are the individuals and whose hyperedges are the communities. Most researchers in the field of community detection seek to partition individuals into communities, i.e. non-intersecting hyperedges. Some authors have attempted to find overlapping communities, i.e. connected hypergraphs. A bipartite graph can be displayed as a hypergraph with individuals at the vertices and properties at the hyperedges. Alternatively, the properties can be considered as vertices and the individuals as hyperedges. An example of a simple hypergraph is shown in Figure 2.

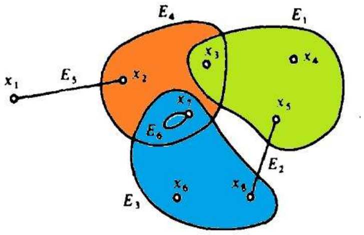

Fig. 2 example of a Hypergraph (a colored version of example in [4] page 2)

2.3 Galois lattice

Freeman [19] was the first to use Galois lattices in order to represent network data. The underlying assumption is that individuals sharing the same subset of properties define a community. The approach adopted consists of the following: objects, attributes and the set of relations between objects and attributes form a "context", in accordance with Formal Concept Analysis [20]. This set of relations can then be represented by a binary bi-adjacency matrix, whereby objects o are the columns, attributes a are the rows and a "1" is placed at the cell corresponding to a pair (o_i, a_j) if o_i possesses a_j. A maximum subset of objects that contain a subset of attributes is defined as a "concept", i.e. a group of objects for which the addition or removal of an attribute changes its constitution. All objects of a concept then form the "extent" and all attributes of a concept give rise to the "intent". A partial order is applied to concepts and serves to establish a hierarchy. According to the definition of Galois hierarchies, an object can appear in all the concepts where it can share the same set of attributes with other objects belonging to other concepts. Figure 3 illustrates a simple example of a Galois lattice in which several individuals are sharing several photos.

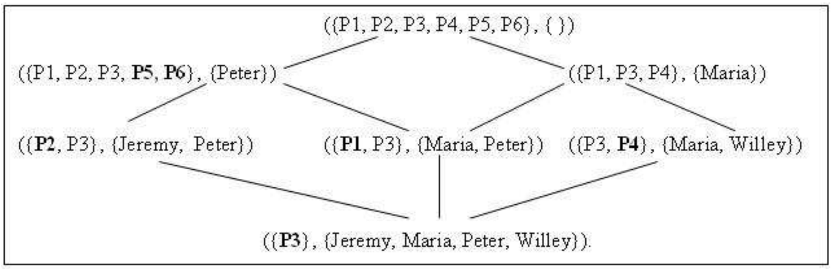
({P1, P2, P3, P4, P5, P6}, { })
({P1, P2, P3, **P5**, **P6**}, {Peter})
({P1, P3, P4}, {Maria})
({**P2**, P3}, {Jeremy, Peter})
({**P1**, P3}, {Maria, Peter})
({P3, **P4**}, {Maria, Willey})
({**P3**}, {Jeremy, Maria, Peter, Willey}).

Fig. 3 example of a Galois lattice (with photos Pi and individuals)

2.4 The concept of modularity

Modularity has been introduced to measure the quality of community algorithms. Newman [48] proceeded with the initial introduction, in providing the following formula:

$$Q = \sum_i (e_{ii} - a_i^2) \qquad (1)$$

where

- e_{ij}: number of edges having one end in group i and the other end in group j.
- $a_i = \sum_j e_{ij}$: number of edges having one end in group i.

This quantity Q measures the fraction of edges in the network that connect vertices of the same type (i.e. “intra-community” edges) minus the expected value of the same quantity in a network with the same community divisions yet with random connections between vertices.

Modularity measures the capacity of a given graph partition to yield the densest groups. This formula has mainly been used by researchers in order to measure the ability of a community detection algorithm to obtain a satisfactory partition of a given graph. Moreover, the formula may be adapted to weighted graphs (i.e. graphs whose edges display different weights or lengths), like in [5].

3 State-of-the- art assessment: existing surveys

This section will address the body of existing surveys on community detection. Most surveys primarily focus on the graph structure aspect of communities. Seven of the main existing surveys have in fact been recently conducted. Since the concept of community may differ, this existing body of surveys defines communities before classifying the methods employed according to various classification systems. This study will start by determining which definitions are provided for these different community detection methods before presenting a state-of-the-art on existing surveys.

3.1 Community definitions

Defining a community is quite a challenging task. Definitions vary from author to author and from algorithm to algorithm. The most commonly used definition is that of Yang [75]: “a community as a group of network nodes, within which the links connecting nodes are dense but between which they are sparse”. This definition is applicable for graphs and could be extended to bipartite graphs.

Fortunato [18] identifies three levels to define a community: local definitions, global definitions, and definition based on vertex similarity. In the local definition group, a definition of communities consists of: "parts of the graph with few ties to the rest of the system". In this partition, communities are studied from their inner structure independently of the remaining part of the graph. In the global definition group, a global criterion associated with the graph is used to compute communities. This global criterion is dependent on the algorithm implemented to locate communities. Either a clustering criterion or a distance-based criterion may be introduced; more often, the criterion commonly used shows that the graph contains a community structure different from that of a random graph. In the vertex similarity-based community definition group, communities are considered as groups of vertices similar to one another.

Fortunato [18] further defines communities, in also calling them clusters or modules, as "groups of vertices that probably share common properties and/or play similar roles within the graph". His assigned definition depends on the algorithm employed, resulting in the identification of at least eight different definitions:

- Clique: subgroups whose members are all "friends" (i.e. connected with an edge) to each other [37],
- n-clique with two variants : maximal sub-graphs such that the distance of each pair from its vertices is not greater than n [36],
- k-plex: maximal subgraph in which each vertex is adjacent to all other vertices of the subgraph except at most k of them [65],
- LS-set (weak community) : subgraph such that the internal degree is greater than the external degrees[35],
- lambda set: subgraph where each pair of vertices has a greater edge connectivity than any pair formed by one vertex of the subgraph and one outside the subgraph [6],
- communities based on either a fitness measure or a quality measure,
- communities determined by means of modularity-based algorithms,
- clusters: communities derived using well-known clustering methods [64].

Porter [57] recalls the origins of community study in the fields of sociology and anthropology. He defines communities as: "cohesive groups of nodes that are connected more densely to each other than to the nodes in other communities". The difference in methods highlighted in his survey relies on a definition of the expression "more densely", which is identified with five types of algorithms, namely: clustering techniques [64], quality function algorithms [31], centrality based community detection algorithms [48] and other similar ones, clique percolation algorithms [15] and lastly modularity optimization algorithms [46].

N. Gulbahce and S. Lehmann [26] define a community as "a densely connected subset of nodes that is only sparsely linked to the remaining network".

Papadopoulos [54]ultimately defines communities as: “groups of vertices that are more densely connected to each other than to the rest of the network”.

It can be seen that all definitions are quite similar yet may still differ in their associated formal mathematical definition. Communities may also be considered from a different perspective. The initial approach partitions the underlying graph, i.e. by dividing the existing graph or network structure into distinct communities using the optimal algorithms. The second approach then detects overlapping communities and seeks the best community arrangement.

In [61] Roth defines a new type of community: “epistemic communities”, which are “knowledge communities or groups of agents sharing common knowledge concerns”, for instance a group of researchers investigating a single precise topic. This new type of community concept requires new kinds of structures to proceed with their description. Roth has opted to use Galois lattices.

3.2 State-of-the-art in community detection surveys

Most surveys classify research papers and methods according to the type of community detection algorithm.

The first of these seven main surveys by S. Fortunato [18] is exhaustive with respect to many community detection methods and has been based on a graphic representation. This survey provides an effective overview of the field and describes the methodological foundations of community detection, adopting a statistical physics perspective and specifically focusing on techniques designed by statistical physicists. His discussion also includes critical issues like: the significance of clustering, the procedure by which methods should be tested and compared to one another, and applications to real networks. Methods are classified into eight families, i.e.:

- traditional methods based on clustering like k-means [39] and others applications [31],
- divisive algorithms mainly based on hierarchical clustering [52],
- modularity-based algorithms [48, 8, 12] and other similar algorithms,
- spectral algorithms [58],
- dynamic algorithms [73, 13] and other similar algorithms,
- statistical inference-based methods [2],
- multi-resolution methods [56],
- and lastly methods to find overlapping communities [53, 15] and other miscellaneous methods.

The second survey, conducted by Porter[57] only includes graph partitioning approaches and offers insight into graphical techniques through citing the first survey. An extensive set of techniques is highlighted, as are some of the most important unresolved issues remaining. Application examples are also given on some of the largest social networks in addition to grouping community detection into five main techniques:

- centrality based techniques built around the Newman algorithm [48],
- local methods around the k-Clique percolation method [53],
- modularity optimization methods around the Newman algorithm[46],
- spectral partitioning methods around Simon's algorithm [58]
- and lastly physics-based methods inspired by Potts law [73].

The third survey by B. Yang [75] is quite exhaustive relative to all techniques relying on graphical representation and produces a good overview of the field through classifying all techniques in a tree structure, according to three categories:

- optimization based algorithms [31, 46, 25],
- heuristic algorithms [21, 69],
- similarity based algorithms and hybrid methods [56].

The fourth one from N. Gulbahce and S. Lehmann [26] is a partial survey analyzing hierarchical type community detection methods and provides a number of leads for future community detection approaches.

The fifth survey by Pons [56] incorporates several community detection methods and classifies them into five different families:

- Classical approaches including classical graph partitioning, e.g. spectral bisection from [58], Kernighan & Lin[31], clustering [28] and hierarchical clustering like [72].
- Separative approaches attempting to split a graph into several communities by deleting the edges connecting distinct communities. In this group, Pons places the well-known Girvan-Newman algorithm[48], and other variant approaches.
- agglomerative approaches quite similar to their hierarchical counterparts and include a method based on optimized modularity by Newman[46] and others algorithms.
- Random walk type algorithms[27] and others based on the mean time required to reach a vertex [79].
- Lastly, a broad group of miscellaneous approaches.

The Pons survey has only examined graph partitioning approaches.

The sixth survey by Papadopoulos [54] classifies community detection techniques in five methodological categories:

- cohesive subgraph discovery [53],
- vertex clustering [56],
- community quality optimization [48, 11],
- divisive [48]
- and model-based methods [23].

The seventh and last survey was undertaken by Danon [14] and primarily focuses on the performance of each type of algorithm.

Among these seven surveys, only one offers leads on overlapping communities, while none make use of hypergraph structures. Another domain overlooked by these surveys is Galois lattice structures. As will be described below, Galois lattices are more complex structures than regular graphs, yet they provide more semantics to network structures.

4 Social Network Community detection Methods

4.1 Approach classification

All community detection methods will be classified in a grid divided into six categories based on the types of input data and output data; it will then be shown how each author can be placed in this grid.

In order to detect communities, the initial input data were considered in network form (whether a social network or biological network, etc.) and represented by different mathematical structures relative to graph structures, which may be of three distinct types:

- unipartite graph: this is a normal graph whose vertices are individuals and whose edges are links connecting the individuals (these links may be of various types: friend, family, club, sport, university, etc.);
- a bipartite graph: this type of graph may be generated whenever individuals share tags (i.e. terms assigned by users), web pages or links. The first set contains individuals, and the second is a set of tags, web links or documents, etc.;
- a multipartite graph: very similar to bipartite graphs, this graph however is composed of several disjoint sets. In this survey, this type of graph has not been directly taken into account since it may be reduced to a bipartite graph, as shown in [44] and elsewhere.

The output data of a community detection method consists mainly of a set of node groups representing communities. The following merit consideration:

- Graph partition, where each node is associated with just one group of nodes and where no overlap exists between groups. Partitions are the primary result of most community detection algorithms.
- Hypergraph with overlapping communities.
- Concept graphs or Galois lattices where nodes share several common properties.

The following table (1) explains which type of input and output data each method is capable of accepting.

Most existing surveys included in our state-of-the-art evaluation describe community detection methods that may be classified within the “A1” cell of

Table 1 Methods according to representations

$\frac{Input\Rightarrow}{Output\Downarrow}$	**1**: Graph	**2**: Bipartite Graph / HyperGraph
A: Partition	input: unipartite graph (see 2.1) output: partition	input: bipartite graph (see 2.1) output : partition
B: Hyper Graph (see 2.2)	input: unipartite graph (see 2.1) output: overlapping communities represented by a hyper-graph	input data: bipartite graph (see 2.1) output: overlapping communities represented by a hypergraph (see2.2)
C: Galois Hierarchy (see 2.3)	no method eligible except [40] with partial results	input: bipartite graph (see 2.1) output: Galois lattice of communities (see 2.3)

table 1. Table 2 below summarizes the major methods according to the above classification.

Table 2 Papers classified according to their Community detection Methods

$\frac{Input\Rightarrow}{Output\Downarrow}$	**1:** graph	**2:** bipartite graph/hypergraph
A: Partition	[78, 71, 70, 7, 47, 76] **S[18, 57, 75, 26, 54]:** *[10, 64, 48, 56, 37, 3, 12, 21]* *[33, 74, 49, 51, 47, 6, 5, 46]*	[66, 42, 61, 59, 67]
B: hypergraph (overlapping communities)	**S[18, 57]:***[50, 16, 1, 33, 53].* [17]**, S[54]**: *[23, 53, 15]*	[44, 9, 34, 41] **S[54]**:*[34]*
C: Galois Hierarchy	[40]: partial results	[29, 68, 62, 19]

Table Legend :
Letter **S** preceding a publication number indicates that this paper is a survey.
The expression: **S**[2]:[1], indicates that the survey referenced in paper number 2 cites and comments on the community detection method described by paper number 1, and moreover paper number 1 can be classified according to the table cell where it has been placed.

4.2 From graphs to partitions (cell A1)

Most community detection algorithms lie in this class of methods. The input data is a normal graph (i.e. a set of vertices representing individuals, who are connected by edges), and the output is a list of node groups representing the communities on the initial graph. Each individual belongs to one and only one community. All surveys mentioned in Section 3 describe these algorithms in full detail; the following algorithm classification has been adopted: top-down (separate) methods in S[18],[56]; bottom-up (agglomerative) and/or clustering methods in S[18, 54, 75, 57],[56]; optimization-based algorithms [75] and heuristic algorithms [75]. The three most popular algorithms will be discussed hereafter, i.e. the Girvan-Newman algorithm [21] based on intermediate centrality, the Newman algorithm [46, 48] based on modularity and the Louvain Algorithm [5].

4.2.1 Girvan and Newman Algorithm

According to survey [18] this algorithm belongs to the category of divisive algorithms. Its underlying principle calls for removing the edges that connect different communities. In the algorithm described in [48], several measures of edge centrality are computed, in particular the so-called intermediate centrality, whereby edges are selected by estimating the level of edge importance based on these measures. As an illustration, intermediate centrality is defined as the number of shortest paths using the edge under analysis. The steps involved are as follows:

1. Compute centrality for all edges,
2. Remove edges with with the greatest centrality (when ties exist with other edges, one edge is to be chosen at random),
3. Recalculate centralities on the remaining graph,
4. Iterate beginning at step 2.

This work has exerted great influence on research and, consequently, edge centrality has been a key field of study for many scientists, resulting in the proposal of several measures [69].

4.2.2 Modularity-based algorithm

This algorithm introduced by Girvan and Newman [48] and then improved in [12] is based on modularity (see Section 2.4). The "glutton" type algorithm maximizes modularity by merging communities at each step in order to get the greatest value increase. Only those communities sharing one or more edges are allowed to merge at each strep. This method is performed in linear time; however, the community quality is less than that of other more costly methods.

4.2.3 Louvain Algorithm

The main benefit of the Louvain algorithm [5] lies in its capacity to operate very quickly on extremely large weighted graphs. This property however does not guarantee an optimal graph partition; an adapted modularity formula, derived from the initial formula presented in Section 2.4, is used for weighted graphs. Initially, all vertices are placed in different communities. At first, all vertices are taken into consideration. For each node i, the algorithm computes the gain in weighted modularity when placing i in the community of its neighbor node j and then chooses the community offering maximal gain. At the end of this first loop, the algorithm yields the first partitioning scheme before repeating the same step while already considering formed communities as new nodes. The algorithm stops once additional increases in modularity are no longer possible. This method has been used to process very large social networks extracted from phone companies, for example, with over 2.6 million customers (see [5] for more details). Its processing time is very short.

4.3 From graphs to overlapping communities, hypergraphs (B1)

TThis class of methods remains atypical, yet a number of authors have attempted its implementation. The input data is a normal graph, while the output is a list of node groups representing the communities of the initial graph; these communities may indeed overlap. The result could be represented as a hypergraph. The survey [57] mentions several methods without providing any description along with two surveys [18, 54] that describe methods found in this class. The most famous of such methods is "Clique percolation"; this algorithm devised by Palla *et al.* [15] speculates that the internal edges of a community are likely to form cliques due to their high density. On the other hand, it is unlikely that intercommunity edges form cliques. Palla *et al.* use the term k-clique in reference to a complete graph with k vertices. Two k-cliques are adjacent if they share k-1 vertices. The union of adjacent k-cliques constitutes s a k-clique chain. Two k-cliques are connected if they form part of a k-clique chain. Moreover, a k-clique community is the largest connected subgraph obtained by uniting a k-clique with all k-cliques connected to it. Several authors have proposed improvements to this method given that the computing time may increase exponentially with the number of nodes or edges in the graph. This method has been determined to provide good results.

Other authors have found alternative ways to extract overlapping communities, such as [22, 23, 17]. As an example, [23] enhanced the Girvan-Newman algorithm (see above) in its ability to detect overlapping communities.

4.4 From bipartite graphs to partitions (A2)

In this class, the inputs are bipartite graphs, representing for example individuals sharing common properties (photos, tags, etc.). The output contains a list of communities from the initial graph. Each node belongs to just one community. No surveys directly describe this particular case; however, [67]], which is based on work performed by Murata [43], adapted a new modularity measure for bipartite graphs in order to build separate communities. Another effort in [59] uses cluster-type local density to extract communities from bipartite graphs modeled as hypergraphs..

4.5 From bipartite graphs to overlapping communities, hypergraphs (B2)

In this class, inputs are bipartite graphs, while outputs are either hypergraphs or lists of node groups representing communities that may or may not overlap. One survey [54] describes this case, in citing [34]. By defining "Epistemic communities" in [61] Roth depicts a partial example of this class and then takes it one step further with Galois hierarchies (see below).

4.6 From bipartite graphs to Galois hierarchies (C2)

This class extends another a step by attempting to extract communities while preserving knowledge shared in each community. No survey has described this type of method. The inputs are bipartite graphs, and the outputs a Galois hierarchy that reveals communities semantically defined with their common properties or shared knowledge. Communities are non-empty lattice extents, and the result is a hypergraph whose hyperedges are labeled by lattice intents (i.e. shared knowledge).

However, a Galois hierarchy, which is roughly computed from the hypergraph input (as in the case of Freeman [19]), is not a satisfactory scheme since a significant number of groups may be obtained. Under ideal conditions, reduction methods should be introduced, which at one level cause the loss of some semantic precision, yet on another level add precision, i.e. cohesion and reliability inside the extracted communities. Only very few authors have actually addressed this difficulty. Roth [61] found the epistemic communities described above before proposing to retain communities of significant size (extent) and semantics (intent), though with weak justification and validation for the proposed heuristics. The authors in [63] then proposed well-known Galois lattice reduction methods based on the so-called iceberg method as well as the stability method.

The iceberg method from [29] identifies concepts with frequent intents above a set threshold. The authors however point out that some important concepts may be overlooked with this method. Stability methods, as used in [29, 32], rely on concept stability. The fewer the number of extent subsets present in child concepts, the greater the concept stability. In [7] it is argued that combining both the iceberg and stability methods yields good results for extracting pertinent communities based on concepts. Two thresholds still need to be set however, as the algorithms computation time may be exponential in the number of objects and attributes (NP complete) and lastly the result presented in the form of a Galois lattice is not easily comprehensible.

4.7 Discussions

In this section, all the major community detection methods have been classified and described. The majority of methods examined lie in cell A1 of Table 2, thus producing a partition scheme of communities, which is a configuration not so well adapted to social networks since individuals may belong to several interest groups. The methods in cell B1 of Table 2 allow extracted communities to overlap. This hypergraph model is better adapted to representing social communities. The methods in cell A2 address the case where individuals are represented with their property knowledge (bipartite graph) yet still provide a partition community scheme. The approaches in cell B2 are more realistic when it comes to representing property sharing communities, although they do require some abstraction. Lastly, the methods in cell C2 are the most accurate, because they extract communities using their precise semantics. Nonetheless, they fall short of giving simple and practical results. It is easy to conclude the lack of perfect methods, as each one presents its pros and cons depending on what the experimenters are seeking. Many methods have been proposed to extract partitioned communities from simple graphs, and this availability of methods is certainly due to the ease of describing this type of problem and drawing a partition in comparison with hypergraphs or Galois lattices.

5 Evaluation methods for community detection

Validation is a key issue: How is it possible to verify that the communities identified are actually the appropriate ones? How is it possible to compare results between two distinct algorithms and declare one better than the other? Several methods may be proposed. One simple sentence from a survey [57] reminds us of the great difficulty involved in evaluating community detection methods: “Now that we have all these ways of detecting communities, what do we do with them?” No evaluation methods are actually given. In his survey, Fortunato [18] provides an effective analysis of evaluation methods in proposing three steps: benchmarks, evaluation measures, and comparative evaluation results. Papadopoulos’ survey [54] merges evaluation with various community definitions; like other surveys available, it does not pay great attention to evaluation, except for presenting applications on well-known cases and extensive practical social networks. Most validation methods have been designed for the methods in cell A1 of Tables 1 and 2. As a matter of fact, most detection methods may be compared to ”clustering” algorithms as regards evaluation. It is well known that clustering yields results that are difficult to evaluate. This same situation is encountered in the area of community detection. Some standard evaluation methods do however emerge. This section will start by presenting several measures of potential use in evaluating the results of community detection methods. Afterwards, evaluation benchmarks will be introduced before reviewing a number of evaluation methods.

5.1 Community extraction results' measures.

5.1.1 Referent graph versus Expert panel

One type of evaluation is based on expert validation. Two validation modes may be considered. Either the result is presented to an expert or a panel of experts, who visually decide whether or not each community is valid. Alternatively, referent community allocation schemes, which are perhaps manually designed by an expert or else a users' panel, are available; according to such a scheme, an expert uses the measures described below in order to decide whether or not the computed result is adequate with a given confidence criterion. This method relies on expertise and may vary depending on the assigned expert; it may also depend on the actual viewpoint adopted. Each community can be positively evaluated provided a justifying angle of interpretation can be found. As such, the tasks of expert work and reliability are rendered quite difficult. Some types of referent graphs however do not introduce any doubt, e.g. a graph showing civil relations between individuals in a wedding. The Karate Club example in [77] is famous because it was known that at one time the club divided into two subgroups and moreover any partitioning method would show this result.

5.1.2 F-Measure based on recall and precision measures

Gregory [22] adapted the measures of recall and precision formulas:

- recall: the fraction of vertex pairs belonging to the same community within the referent benchmark graph that are also members of the same community in the resulting partition;
- precision: the fraction of vertex pairs that are members of the same community in the resulting partition while also belonging to the same community in the referent benchmark graph.

The F-measure is used in this context, i.e. the harmonic average of recall and precision. The F-measure provides a useful balanced vision of the community detection algorithm. The dual recall and precision measures may also be introduced, in applying these formulas on the edges instead of the vertices.

These kinds of measures are very practical yet remain difficult to adapt if the community number in the resulting partition scheme is not the same as that in the referent benchmark graph. Certain adapted measures may be implemented for added flexibility on resulting communities.

Girvan-Newman [21] proposed a similar measure: the fraction of correctly assigned community vertices, divided by the total size of the graph.

5.1.3 Automatic graph and community generation

As will be seen in greater detail in the following subsections, a number of authors have generated automatic graphs and graph community schemes using several methods. They then compared their community detection algorithms to the communities actually generated. Some generation methods produce random communities, in which case the goal consists of proving that their algorithms are better than randomness. Other generation methods offer a community scheme according to a previously defined goal (e.g. generate 5 communities), with the authors then attempting to obtain a similar result with their algorithm. This method will be discussed in more precise terms below.

5.1.4 quality measure

A community detection scheme may be considered effective or ineffective by using a so-called quality function. This specific measure provides a means for comparing quality across several community detection schemes.

One of these quality measures, ”Modularity”, which was introduced by Newman [48] and has already been mentioned in Section 2.4, is very famous and widely used. Modularity expresses the fact that a community has a high density ratio as compared to the same graph without any community structure. This high-density criterion is considered to offer good community detection quality. Some authors have combined a local quality measure (based on modularity) along with the potential of community communication in order to produce an overall quality ratio (see for example [11]).

Several authors cited in the Fortunato survey [18] (in his Section C2) note that a community detection method yielding strong modularity results is not always the best choice, in arguing that low modularity values could provide greater stability in communities.

5.1.5 Discussion and more global measures

Some measures may prove to be contradictory, as indicated in [30]. Moreover, most measures focus on the mathematical properties of a graph. In a social network however, an individual may have different intentions regarding group membership. Real communities may not be optimal with respect to collective modularity. In the real world, communities are determined by their history, which in turn is driven by individual personalities and contingent events. The social optimum at present is not easily able to manage and explain everything. Other methods are needed to take into account these dynamic and human dimensions.

5.2 Standard Benchmarks, random generated graphs

Considered by many researchers as references, several popular real networks are often used as benchmarks. Some of these are cited along with their vertex number v and edge number e: Zachary Karate Club [77](v=34, e=78); a social network of dolphins living in Doubtful Sound (New Zealand) [38] (v=62, e=159); college football team games [21] (v=115, e=613); University e-mail network [24] (v=1133, e=5451); and scientific co-authorship in condensed matter physics [45] (v=27519, e=116181). For some of these, a final community list for their graphs exists, while for others only the graph structure is present. Girvan-Newman [21] designed a random computer-generated graph and community partition scheme with 4 groups and 32 vertices in each group. This set-up has since become a standard benchmark. A new enhanced version of this graph has been designed by Brandes [8]. These standard graphs and benchmarks can then be used to compare community results with a particular community detection algorithm by applying the previously defined measures.

5.3 Community detection method evaluation

From an evaluation standpoint, community detection is a complex problem. Each method evaluation turns out to be different, and comparing each method's performance proves to be a real challenge. By using the set of measures presented in Section 5.1, a performance evaluation is derived for a given algorithm. A good solution consists of obtaining results from several methods and maintaining the community detection schemes generated by several methods. This option guarantees good stability of the community detection scheme. Testing a method against the Girvan-Newman benchmark entails calculating the similarity between partitions determined by the method and the natural partition of the graph within the four equal-sized groups. Regarding the two popular real networks with a known community structure, i.e. the social networks of Zachary's Karate Club and bottlenose dolphins, the question raised is whether the actual separation into two social groups could have been predicted from the graph topology. Zachary's Karate Club is by far the most widely investigated system. Several algorithms are in fact able to identify the two classes, notwithstanding a few intermediate vertices, which could potentially be misclassified. For example, the so-called Louvain Algorithm finds 5 karate club communities instead of 2. This process however does not guarantee the performance on real networks. Other dimensions, such as semantics and pragmatics, need to be considered as we have argued above. Community detection methods that keep knowledge embedded in the original network, i.e. based on Galois hierarchies or hypergraphs, may lead to more accurate results. This evaluation process enhancement has yet to be introduced. Some of the authors from the C2 and C3 cells in Table 2 have addressed this difficult issue, though semantics and pragmatics considerations must still be developed in community detection evaluation methods.

6 Conclusion

The study of networked communities is, in some respects, now quite old, with its origins traced back to sociology, computer science, statistics and other disciplines. Nevertheless, the expanding field of social networks has focused greater attention on this topic. The present survey has provided a state-of-the-art on existing methods with a new angle: classifying algorithms according to their input and output data schemes. Graphs, hypergraphs and Galois lattices are proven to be useful in representing the growing complexity of community detection methods. This development has allowed demonstrating how to share knowledge and information among social network users. Further research is still required in this field given the organizing power and commercial interest inherent in knowledge. More methods should be developed and their associated software tools are expected to follow.

References

1. Yong-yeol Ahn, James Bagrow, and Sune Lehmann. Communities and Hierarchical Organization of Links in Complex Networks. *eprint physics*, 1:1–8, 2009.
2. Alex M Andrew. *INFORMATION THEORY, INFERENCE, AND LEARNING ALGORITHMS, by David J. C. MacKay, Cambridge University Press, Cambridge, 2003, hardback, xii + 628 pp., ISBN 0-521-64298-1 (30.00)*, volume 22. Cambridge University Press, 2004.
3. James P Bagrow. Evaluating Local Community Methods in Networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(05):8, 2007.
4. Claude Berge. Hypergraphes, Combinatoires des ensembles finis. *Gauthier-Villars*, 1987.
5. Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, October 2008.
6. Stephen P Borgatti, Martin G Everett, and Paul R Shirey. LS sets, lambda sets and other cohesive subsets. *Social Networks*, 12(4):337–357, 1990.
7. U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner. On Modularity Clustering. *Knowledge and Data Engineering, IEEE Transactions on*, 20(2):172 – 188, 2008.
8. U. Brandes, M. Gaertler, and D. Wagner. Experiments on Graph Clustering Algorithms. In *Proc. 11th Europ. Symp. Algorithms (ESA '03)*, pages 568–579, 2003.
9. Camille Roth. Compact, evolving community taxonomies using concept lattices. In Pascal Hitzler, Henrik Schaerfe, and Peter Ohrstrom, editors, *Contributions to 14th International Conference on Conceptual Structures*, pages 172–187, Aalborg, 2006. Aalborg University Press.
10. Andrea Capocci, Vito D P Servedio, Guido Caldarelli, and Francesca Colaiori. Detecting communities in large networks. *Physica A: Statistical Mechanics and its Applications*, 352:669–676, 2005.
11. Aaron Clauset. Finding local community structure in networks. *Physical Review E - Statistical, Nonlinear and Soft Matter Physics*, 72(2 Pt 2):7, 2005.
12. Aaron Clauset, Mark Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6):1–6, 2004.
13. D. Hughes. Random walks and random environments. Volume 1: Random walks. *Bulletin of Mathematical Biology*, 58(3):598–599, 1996.
14. Leon Danon, Jordi Duch, Albert Diaz-Guilera, and Alex Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(09):10, 2005.
15. Imre Derényi, Gergely Palla, and Tamas Vicsek. Clique percolation in random networks. *Physical Review Letters*, 94(16):160202, 2005.

16. T S Evans and R Lambiotte. Line Graphs, Link Partitions and Overlapping Communities. *Physical Review E*, 80(1):9, 2009.
17. L Falzon. Determining groups from the clique structure in large social networks. *Social Networks*, 22(2):159–172, 2000.
18. Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):103, June 2009.
19. Linton C Freeman and D R White. Using Galois Lattices to Represent Network Data. *Sociological Methodology,23*, pages 127–146, 1993.
20. Ganter B. and Wille R. *Formal concept analysis: foundations and applications*. 1999.
21. M. Girvan and Mark Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99(12):7821–7826, 2002.
22. Steve Gregory. A Fast Algorithm to Find Overlapping Communities in Networks. *Machine Learning and Knowledge Discovery in Databases*, 5211:408–423, 2008.
23. Steve Gregory. Finding overlapping communities in networks by label propagation. *New Journal of Physics*, 12(10):103018, 2009.
24. R Guimerà, L Danon, A Díaz-Guilera, F Giralt, and A Arenas. Self-similar community structure in a network of human interactions. *Physical Review E*, 68(6):1–4, 2003.
25. Roger Guimerà, Marta Sales-Pardo, and Luís Amaral. Module identification in bipartite and directed networks. *Physical Review E*, 76(3), September 2007.
26. Natali Gulbahce and Sune Lehmann. The art of community detection. *BioEssays news and reviews in molecular cellular and developmental biology*, 30(10), October 2008.
27. Harel David and Koren Yehuda. On Clustering Using Random Walks. In R. Hariharan, M. Mukund, and and V. Vinay, editors, *FSTTCS 2001, LNCS 2245*, pages 18–41, Berlin Heidelberg, 2001. Springer-Verlag.
28. A K Jain, M N Murty, and P J Flynn. Data clustering : A review. *ACM Comput. Surv.*, 31(3):264–323, 1999.
29. Nicolas Jay, François Kohler, and Amedeo Napoli. Analysis of social communities with iceberg and stability-based concept lattices. pages 258–272, February 2008.
30. Jon Kleinberg. An Impossibility Theorem for Clustering. In Klaus Obermayer Suzanna Becker, Sebastian Thrun, editor, *Advances in Neural Information Processing Systems 15*. Suzanna Becker,Sebastian Thrun,Klaus Obermayer, August 2002.
31. B W Kernighan and S Lin. An Efficient Heuristic Procedure for Partitioning Graphs. *Bell Sys. Tech. J.*, 49(2):291–308, 1970.
32. Sergei O. Kuznetsov. On stability of a formal concept. *Annals of Mathematics and Artificial Intelligence*, 49(1-4):101–115, June 2007.
33. Andrea Lancichinetti, Santo Fortunato, and János Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, March 2009.
34. Yu-Ru Lin, Jimeng Sun, Paul Castro, Ravi Konuru, Hari Sundaram, and Aisling Kelliher. MetaFac: community discovery via relational hypergraph factorization. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 527–536, New York, NY, USA, 2009. ACM.
35. F Luccio and M Sami. On the decomposition of networks in minimally interconnected subnetworks. *Ieee Transactions On Circuit Theory*, 16:184–188, 1969.
36. R D Luce. Connectivity and generalized cliques in sociometric group structure. *Psychometrika*, 15(2):169–190, 1950.
37. R D Luce and A D Perry. A method of matrix analysis of group structure. *Psychometrika*, 14(1):95–116, 1949.
38. David Lusseau, Karsten Schneider, Oliver J Boisseau, Patti Haase, Elisabeth Slooten, and Steve M Dawson. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003.

39. J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In Cam LML. and Neyman L., editors, *Proc. of the fth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, Berkeley, USA, 1967. University of California Press.
40. Michel Crampes, Michel Plantié, and Bidault Julien. Cliques maximales d'un graphe et treillis de Galois. In *MARAMI conférence sur les Modèles et l'Analyse des Réseaux : Approches Mathématiques et Informatique*, GRENOBLE, 2011.
41. Hiroyuki Miyagawa. Community Extraction in Hypergraphs Based on Adjacent Numbers. *Operations Research*, 50:309–316, 2010.
42. Tsuyoshi Murata. Detecting communities from tripartite networks. WWW '10. ACM Press, 2010.
43. Tsuyoshi Murata. Modularity for heterogeneous networks. In *Proceedings of the 21st ACM conference on Hypertext and hypermedia - HT '10*, page 129, New York, New York, USA, June 2010. ACM Press.
44. Neubauer Nicolas and Obermayer Klaus. Towards Community Detection in k-Partite k-Uniform Hypergraphs. In *Proceedings NIPS 2009 . . .*.
45. Mark Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences of the United States of America*, 98(2):7, 2000.
46. Mark Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69(6), June 2004.
47. Mark Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E - Statistical, Nonlinear and Soft Matter Physics*, 74(3 Pt 2):036104, 2006.
48. Mark Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2), February 2004.
49. Mark Newman and Juyong Park. Why social networks are different from other types of networks. *Physical Review E - Statistical, Nonlinear and Soft Matter Physics*, 68(3 Pt 2):036122, 2003.
50. V Nicosia, G Mangioni, V Carchiolo, and M Malgeri. Extending the definition of modularity to directed graphs with overlapping communities. *Journal of Statistical Mechanics: Theory and Experiment*, 2009(03):P03024, March 2009.
51. Andreas Noack and Randolf Rotta. Multi-level algorithms for modularity clustering. page 12, December 2008.
52. Klaus Nordhausen. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition by Trevor Hastie, Robert Tibshirani, Jerome Friedman. *International Statistical Review*, 77(3):482–482, 2009.
53. Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–8, June 2005.
54. S Papadopoulos, Y Kompatsiaris, A Vakali, and P Spyridonos. Community detection in Social Media. *Data Mining and Knowledge Discovery*, (June):1–40, 2011.
55. Michel Plantié and Michel Crampes. From Photo Networks to Social Networks, Creation and Use of a Social Network Derived with Photos. *Proceedings of the ACM international conference on Multimedia , Firenze, Italy, October*, 2010.
56. Pascal Pons. *Détection de communautés dans les grands graphes de terrain.* PhD thesis, Paris 7, 2007.
57. M.A. Porter, J.P. Onnela, and P.J. Mucha. Communities in Networks, 2009.
58. Alex Pothen, Horst D. Simon, and Kang-Pu Liou. Partitioning Sparse Matrices with Eigenvectors of Graphs. *SIAM Journal on Matrix Analysis and Applications*, 11(3):430, May 1990.
59. Rong Qian, Wei Zhang, and Bingru Yang. Community detection in scale-free networks based on hypergraph model. In *Proceedings of the 2007 Pacific Asia conference on Intelligence and security informatics*, PAISI'07, pages 226–231, Berlin, Heidelberg, 2007. Springer-Verlag.

60. Camille Roth and Paul Bourgine. Binding Social and Cultural Networks: A Model. *Networks*, nlin.AO(February):8, 2003.
61. Camille Roth and Paul Bourgine. Epistemic Communities: Description and Hierarchic Categorization. *Mathematical Population Studies: An International Journal of Mathematical Demography*, 12(2):107–130, 2005.
62. Camille Roth and Paul Bourgine. Lattice-based dynamic and overlapping taxonomies: The case of epistemic communities. *Scientometrics*, 69(2):429–447, 2006.
63. Camille Roth, Sergei Obiedkoy, and Derrick G Kourie. On succinct representation of knowledge community taxonomies with formal concept analysis. *International Journal of Foundations of Computer Science*, 19(2):383, 2008.
64. S SCHAEFFER. Graph clustering. *Computer Science Review*, 1(1):27–64, August 2007.
65. Stephen B Seidman and Brian L Foster. A graph-theoretic generalization of the clique concept. *Journal of Mathematical Sociology*, 6(1):139–154, 1978.
66. N. Selvakkumaran and G. Karypis. Multiobjective hypergraph-partitioning algorithms for cut and maximum subdomain-degree minimization . *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 25(3):504 – 517, 2006.
67. Kenta Suzuki and Ken Wakita. Extracting Multi-facet Community Structure from Bipartite Networks. *2009 International Conference on Computational Science and Engineering*, 4:312–319, 2009.
68. Carla Taramasco, Jean-Philippe Cointet, and Camille Roth. Academic team formation as evolving hypergraphs. *Scientometrics*, 85(3):721–740, 2010.
69. Joshua R Tyler, Dennis M Wilkinson, and Bernardo A Huberman. Email as Spectroscopy: Automated discovery of Community Structure within Organizations. In *Communities and technologies*, pages 81–96. Kluwer, 2003.
70. Li Wan, Jianxin Liao, Chun Wang, and Xiaomin Zhu. JCCM: Joint Cluster Communities on Attribute and Relationship Data in Social Networks. *Advanced Data Mining and Applications*, (60525110):671–679, 2009.
71. Li Wan, Jianxin Liao, and Xiaomin Zhu. CDPM: Finding and Evaluating Community Structure in Social Networks. In *Proceedings of the 4th international conference on Advanced Data Mining and Applications*, ADMA '08, pages 620–627, Berlin, Heidelberg, 2008. Springer-Verlag.
72. Joe H Ward. Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association*, 58(301):236–244, 1963.
73. F. Y. Wu. The Potts model. *Reviews of Modern Physics*, 54(1), January 1982.
74. Fang Wu and Bernardo A Huberman. Finding Communities in Linear Time: A Physics Approach. *The European Physical Journal B Condensed Matter*, 38(2):331–338, 2003.
75. Bo Yang, Dayou Liu, Jiming Liu, and Borko Furht. *Discovering communities from Social Networks: Methodologies and Applications* . Springer US, Boston, MA, 2010.
76. Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. A Bayesian Approach Toward Finding Communities and Their Evolutions in Dynamic Social Networks. *Work*, pages 990–1001, 2009.
77. W W Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.
78. F. Zaidi, A. Sallaberry, and G. Melancon. Revealing Hidden Community Structures and Identifying Bridges in Complex Networks: An Application to Analyzing Contents of Web Pages. In *IEEE/WIC/ACM Int Conf Web*, pages 198–205. IEEE, 2009.
79. Haijun Zhou and Reinhard Lipowsky. Network Brownian Motion: A New Method to Measure Vertex-Vertex Proximity and to Identify Communities and Subcommunities. In *Int Conf Computational Science*, volume 3038 of *Lecture Notes in Computer Science*, pages 1062–1069. Springer, 2004.