DETECTING COMMUNITIES FROM NETWORKS: COMPARISON OF ALGORITHMS ON REAL AND SYNTHETIC NETWORKS

Karen Mkhitaryan, Josiane Mothe, Mariam Haroutunian

Abstract: Communities in real world complex networks correspond to hidden structures that are composed of nodes tightly connected among themselves and weakly connected with other nodes in the network. There are various applications of automatic community detection in computer science, medicine, machine learning, sociology, etc. In this paper, we first present the existing community detection algorithms and evaluation measures used in order to consider the algorithms effectiveness. We then report a deep comparison of the algorithms using both large scale real world complex networks and artificial networks generated from stochastic block model. We found that Louvain algorithm is consistently the best across both the measures and the networks (8 real world and many varied synthetic networks) we tested. Fast Greedy and Leading Eigenvector algorithms are also good alternatives. Moreover, compared to related work, our paper considers both more algorithms and more networks.

Keywords: Complex Networks, Community Detection, Stochastic Block Model, Evaluation Measures.

Introduction

Network science gets more and more attention with the increase of computing power enabling to challenge more complex problems and with the rapid increase in the amount and types of *related* data [Fortunato, 2010]. Network science aims at studying complex relational data and networks both theoretically and on the application point of view with real world networks.

Real world networks are represented as undirected, directed or weighted graphs, composed of nodes and edges where edges connect the nodes. Real

networks and synthetic random graphs differ in their structure. In random graphs the distribution of the node degree is homogeneous which means the probability of having an edge between any two nodes is the same. On the other hand, in real world complex networks the structure is usually heterogeneous which results in groups of nodes that are tightly connected to each other but weakly connected with nodes from other groups.

This property of real world complex networks is known as *community structure.* Communities (or clusters) are defined as groups of nodes that have high inter connectivity (strong links inside the groups) and weak inter-connectivity (weak link between groups) [Fortunato, 2010]. For example in Digital Bibliography and Library Project (DBLP) collaboration network which will be used later, nodes represent the authors and edges represent the connections between authors i.e. published a paper together. Communities in DBLP network can be interpreted as groups of authors with the same research interest. Synthetic networks can also mimic the complexity of real world ones.

The aim of community detection is to identify the groups with high intra-connectivity by analyzing the information that the graph topology encodes.

There are a lot of applications of community extraction in networks in various domains such as in biology (e.g. protein to protein interactions [Mahmoud et al., 2014]), sociology (e.g. social network analysis), [Girvan and Newman, 2001, Bedi and Sharma, 2016, He et al., 2018]), information retrieval (e.g. recommendation systems) [Parimi and Caragea, 2014]), etc.

There is no globally accepted definition of *community* but various definitions are used in the literature [Fortunato, 2010, Hu et al., 2008, Radicchi et al., 2004]. In [Fortunato, 2010], Fortunato defined community using local and global definitions as well as based on vertex similarity. Local definitions consider a community as a separate autonomous entity while in global definitions communities are important parts of a whole graph and considering them separately affects the functionality of the system. In definitions that are based on vertex similarity, a similarity measure is defined for each pair of nodes (e.g., Manhattan distance, cosine similarity, etc.) and they are joined based on the chosen similarity measure [Fortunato, 2010]. Radicchi et al. defined community

in a strong and weak sense using degree of the nodes [Radicchi et al., 2004]. There are also community scoring functions used to quantitatively assess how community-like are the groups of nodes (e.g., conductance, modularity, global clustering coefficient) [Leskovec and Yang, 2015, Newman and Girvan, 2004] that we will detail later in section 3.

Modern real world networks like Facebook and DBLP may contain millions or billions of nodes and edges which obscure the process of community detection due to computational issues. However there are well designed approximation algorithms with low complexity fitting large scale networks [Fortunato, 2010, Lancichinetti and Fortunato, 2011], which we evaluate and compare later in this paper (see section 5). Since the number of easily and openly accessible large real world networks is limited, benchmark models such as Lancichinetti Fortunato Radicchi (LFR) benchmark [Lancichinetti et al., 2008] and the Stochastic Block Model (SBM) [Abbe, 2018] which generate networks with community structure are used to overcome this issue. Although the use of artificial networks is beneficiary to generate random networks with different parameters and properties (e.g., degree distribution, number of communities, number of vertices inside communities etc.), they are far from being real world networks as the structure of real networks can be unpredicted resulting in different topologies.

In the literature of the domain, algorithms have been mostly evaluated and compared either on artificial networks [Lancichinetti and Fortunato, 2011, Lancichinetti et al., 2008, Orman and Labatut, 2009, Yang et al., 2017] or small real world networks [de Sousa and Liang, 2014, Moradi et al., 2012]. These settings are not enough to make strong conclusions. Since there are many reasons why the results may differ from network to network, tests and comparisons should be made on many different networks of various types and properties to make acceptable conclusions. Therefore in our research we included both artificial networks and real networks (see section 4). We generated artificial networks using stochastic block model [Abbe, 2018]. As for real networks, we used SNAP datasets where number of nodes range from 1589 to 334,863 [Leskovec and Krevl, 2014].

The evaluation of a community detection algorithms is a crucial research problem. It implies to consider one or several networks, to apply the algorithm and to analyze how "good" or "bad" the detected communities are. This can be done by comparing the estimated community structure with a reference structure or ground truth using external measures or by assessing the quality of detected communities internally. In this paper, we present the main measures from the literature (See Section 3). We also use them in order to compare popular community detection algorithms (See Section 5).

The aim of this paper is to provide the reader with an overview of the methods and algorithms that have been developed for community detection and to compare them using different measures on different types of networks, both real world and synthetic. We did not study specific cases of community detection and some specific algorithm behavior. A few studies could be mentioned on this topic. For example, Abbe studied the limits for community detection in the SBM, both with respect to information-theoretic and computational thresholds [Abbe, 2018]. Overlapping community detection is also an important topic. In [Xie et al., 2013], Xie et al. considered fourteen algorithms for their ability to detect overlapping nodes; which is a different problem although related to. The authors first analyzed community detection ability using NMI and Omega index. Omega index measures the number of node pairs are together in no clusters, the number of pairs that are together in one cluster only, in two clusters, and so on.

The rest of the paper is organized as follows: In Section 2 we describe the community detection algorithms from the literature. The two next sections present the evaluation framework with regards to measures (Section 3) and benchmark networks (Section 4) including artificial networks generated automatically and real networks. Section 5 presents the evaluation of the algorithms using various measures and considering the different networks. Section 6 presents the related work. Section 7 draws some conclusions.

Community Detection Algorithms

Various community detection algorithms have been developed which differ in terms of complexity and network types they target (e.g., undirected, directed, weighted, etc.). In this paper, we chose to present them briefly and to provide

the link to the papers where details of the algorithms and implementations can be found for the interested reader. Heuristics or approximation algorithms are used to optimize some given objective function (e.g. modularity) to detect communities. Despite these barriers, various algorithms exist in the literature, including those that were initially developed for cluster analysis. These algorithms fall mainly into the following main categories [Fortunato, 2010]: modularity-based algorithms, spectral algorithms, algorithms based on random walks, label propagation and information-theoretical measures that we detail in the next sub-sections.

Algorithms based on modularity optimization. Modularity is a community scoring function which characterizes how community-like are the groups of nodes in the network. Modularity is defined as :

$$\text{Modularity} = \frac{1}{2M}\sum_{x,y}(A_{x,y} - \frac{k_x k_y}{2M})\delta(c_x, c_y) \quad (1)$$

where x and y are nodes, M is the number of edges in the network, k_x and k_y the degrees of x and y respectively and $\delta(c_x, c_y)$ is 1 when x and y fall in the same community and 0 otherwise.

Algorithms based on modularity optimization such as Newman's greedy algorithm [Newman and Girvan, 2004] and its updated version by Clauset et. al, namely *Fast Greedy* [Clauset et al., 2004] join vertices which result in highest increase in modularity. After an iterative process when modularity cannot be maximized anymore, the network is partitioned into communities. Another popular modularity optimization method is *Louvain's* algorithm which initially finds small communities by optimizing modularity locally and then aggregates nodes belonging to the same community and creates a network where nodes represent the communities. This process is iterated until maximum modularity is reached and a hierarchy of communities is produced [Blondel et al., 2008].

Infomap. *Infomap* is a flow based information theoretic method used to reveal community structure in the networks. At the beginning every node is assigned to its own community. Then nodes are moved to neighboring communities that results in the largest decrease of the map equation which is the theoretical limit to describe the path of a random walker on the network for a given partition.

After an iterative process, when map equation is minimized over all possible network partitions and no move results in decrease of it, network splits into communities [Rosvall and Bergstrom, 2007].

Random Walks. In general, communities in networks have more intra connectivity (inside communities) than inter connectivity (between communities). Thus it is expected to have more edges inside those groups than between them. When implementing a short random walk, the probability that both the starting and ending points will be in the same group rather than in different groups is thus higher. Algorithms based on random walks like *Walktrap* [Pons and Latapy, 2005] use this idea to detect communities in networks.

Algorithm based on eigenvectors of modularity matrix. This algorithm proposed by Newman, namely *Leading Eigenvector* [Newman, 2006] uses eigenspectrum of modularity matrix. This algorithm initially creates the modularity matrix $M = A - P$, where A is the adjacency matrix of the network and P is the probability matrix, where $P[i, j]$ is the probability that there is an edge between vertices i and j in a random network which has the same degree distribution as the original network. Then the method takes the eigenvector of the modularity matrix for the largest positive eigenvalue and partitions the network into communities considering the sign of the corresponding element in the eigenvector.

Label Propagation. Unlike other community detection algorithms, *label propagation* does not optimize any given objective function and it does not require to have a priori information about the network structure. Initially every node has its own label (corresponding to its community) and during an iterative process nodes gain the label which is frequent in their neighborhood. When every node has the label that the maximum number of its neighbors have, the algorithm stops, resulting in densely connected groups.

Spinglass. *Spinglass* is an approach from statistical mechanics introduced by Reichardt and Bornholdt [Reichardt and Bornholdt, 200 which is based on the Potts model (model of interacting spins). In the network each vertex can be in one spin state. The number of total spin states equals the number of communities in the network. The edges of the network identify the pairs of

vertices that state in the same spin state and vice versa. At the end, after running the model for a number of steps, spin states represent the community assignments.

Girvan-Newman Algorithm. This algorithm proposed by Girvan and Newman [Girvan and Newman, 2001] uses the concept of betweenness centrality and generalizes it to define *Edge betweenness* centrality which separates tightly connected vertices. Betweenness centrality of a vertex *v* is defined as:

$$B(v) = \sum_{p \neq v \neq q} \frac{S_{p,q}(v)}{S_{p,q}} \tag{2}$$

where $S_{p,g}$ is the total number of shortest paths from vertex p to vertex *q* and $S_{p,g}(v)$ is the number of shortest paths from vertex p to vertex q that pass through node v. The first step of the algorithm is to calculate betweenness of all edges, then remove the edge with highest betweenness score and recalculate betweenness of all edges again. After an iterative process, when no edges remain in the network, result can be shown as a dendrogram. According to dendrogram and modularity score, the proper partition is chosen.

Table 1: Algorithms - the underlying theory used, computing complexity where N is the number of nodes and E is the number of edges in the network and main reference.

Algorithm	Based on	Complexity	Reference
Fast Greedy	Modularity	$O(NElog2E)$	[Clauset et al., 2004]
Louvain	Modularity	$O(E)$	[Blondel et al., 2008]
Infomap	Information theory	$O(N(N + E))$	[Rosvall, Bergstrom, 2007]
Walktrap	Random walk	$O(N^2E)$	[Pons, Latapy, 2005]
Leading Eigenvector	Eigen vectors	$O(N(N + E))$	[Newman, 2006]
Label Propagation	Neighborhood	$O(N + E)$	[Raghavan et al., 2007]
Spinglass	Spin states	$O(N^{3:2})$	[Reichardt, Bornholdt, 2006]
Edge betweenness	Centrality	$O(NE^2)$	[Girvan and Newman, 2001]

Table 1 reports the main algorithms to detect communities along with the underlying theory and their complexity in terms of calculation depending on the

number of nodes *N* and edges *E.* We also provide in this table the main reference where the reader can find the details of the eight algorithms.

Measures to Compare Algorithms

Detecting communities in large networks can sometimes be much complicated due to computational complexity of algorithms specifically for networks with large number of nodes and edges. Indeed, exact detection can be a NP-hard problem (see Table 1). Distinguishing between algorithms and deciding which algorithm works the best on a particular network is not obvious.

Algorithms can be compared in terms of efficiency and effectiveness. In the case of community detection algorithms, in addition to algorithm complexity (see Table 1), efficiency refers to the time taken to partition the network while effectiveness is a qualitative measure describing how "good" the derived communities are.

Measures used to assess the quality of detected communities are in turn divided into two main categories [Halkidi et al., 2002, Chakraborty et al., 2017] as described below.

Internal measures

Internal measures are used to quantitatively assess how community-like is the given set of nodes in the network. As the global definition of community is based on the idea that it has high connectivity within a group and weak connectivity with other groups, scoring functions are also based on this intuition. In the following, we will focus on modularity, conductance, and global clustering coefficient measures. Conductance give optimal results when identifying ground truth communities [Leskovec and Yang, 2015] and modularity is the most widespread evaluation criteria used in the literature.

Modularity. Modularity is a well accepted measure to evaluate a partition of a network into communities that is to say to assess the quality of the network division into communities. The basic idea is that random networks do not contain community structures thus its structure can be compared to a give network to have an idea of the network community-like structure. Modularity is

then defined as "the fraction of the edges that fall within the given groups minus the expected fraction if edges were distributed at random" [da Costa Alves, 2010]

$$\text{Modularity} = \frac{1}{2M}\sum_{xy}(A_{xy} - \frac{k_x k_y}{2M})\delta(c_x c_y). \quad (3)$$

where M is the number of edges in the network, kx and ky the degrees of nodes. x and y respectively and $\delta(c_x c_y)$ is 1 when x and y fall in the same community and 0 otherwise.

Partitions of the network with strong community structure where vertices are densely connected inside communities and sparsely connected between communities have higher modularity score [Newman and Girvan, 2004].

Experiments showed that modularity suffers from resolution limit merging small groups in case of low resolution and splitting large groups in case of high resolution i.e. missing important structures in the network [Lancichinetti and Fortunato, 2011] and often it is not possible to eliminate both biases simultaneously.

Conductance. Conductance measures how strong or dense the community structure of the graph is. Conductance is the fraction of the total edges that goes outside the community and is defined as:

$$Conductance = \frac{O_c}{2I_c + O_c} \quad (4)$$

where O_C is the number of edges pointing outside from the community c and I_c is the number of edges within c.

Using conductance as a community goodness metric Leskovec et al. showed that the best possible communities get less community-like structure when network size grows [Leskovec et al., 2008]. In their other study while experimenting on large networks from 0.33 million up to 117.7 million nodes, conductance and triangle participation ratio gave best results in identifying ground truth communities [Leskovec and Yang, 2015].

Global clustering coefficient. A clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together [Deng et al., 2016]. The global clustering coefficient is designed to give an overall indication of the clustering in the network. Global clustering coefficient is defined as the number of triangles in the network divided by the number of connected triplets (subgraphs having three vertices and two edges) as follows [Wasserman and Faust, 1994]:

$$T = \frac{3 \times \text{number of triangles}}{\text{number of connected triplets of vertices}} \tag{5}$$

In real world networks where the degree distribution is highly heterogeneous, global clustering coefficient shows how much the vertices tend to cluster together to form dense groups [Wasserman and Faust, 1994].

External measures

Unlike internal measures which are used to assess the quality of detected communities, external measures are used to evaluate and compare a given partition X with another or with ground truth if available, denoted Y in what follows.

Normalized Mutual Information Mutual Information (MI) is an information-theoretic measure that quantifies the mutual dependence between two random variables. It is applied in community detection to compare two partitions. MI measures how much information can be obtained about one partition (random variable) from another one.

Normalized Mutual Information (NMI) is the normalized variant of MI [Collignon et al., 1995]. NMI is defined as :

$$\text{NMI}(X, Y) = \frac{2I(X, Y)}{H(X) + H(Y)} \tag{6}$$

where $H(X)$ and $H(Y)$ are the entropies of random variables corresponding to two partitions X and Y respectively. Thus NMI(X, Y) measures the similarity of the two partitions [Cover and Thomas, 2006].

Variation of information Variation of information is also an information-theoretic measure of the distance between two random variables similar to MI and NMI. However unlike these measures, it is a true metric measure defined as:

$$\mathrm{VI}(X,Y) = H(X|Y) + H(Y|X) \tag{7}$$

Considering X and Y as two different partitions of the network, VI(X, Y) measures the variation of information between the two partitions [Meila, 2007].

Adjusted Rand Index Adjusted Rand Index (ARI) is another similarity measure of two different partitions. Given a set of n elements, $S = (d_1, d_2, ..., d_n)$, and two partitions of S, X and Y respectively, ARI is defined as:

$$\mathrm{ARI} = \frac{SS + DD}{SS + SD + DS + DD}, \tag{8}$$

where SS is the number of pairs of elements in S that are in the same subset in X and in the same subset in Y, DD is the number of pairs of elements in S that are in different subsets in X and in different subsets in Y, SD is the number of pairs of elements in S that are in the same subset in X and in different subsets in Y, DS is the number of pairs of elements in S that are in different subsets in X and in the same subset in Y [Rand, 1971, Hubert and Arabie, 1985].

In the evaluation part of this paper we use conductance, global clustering coefficient and modularity to assess the quality of detected communities internally and NMI, VI and ARI to compare partitions externally. We will also measure efficiency considering the processing time of the algorithms in various configurations.

Networks Used for the Evaluation of the Algorithm

In our experiments we use eight real world networks (See Table 2) provided from SNAP Stanford database (http://snap.stanford.edu/data/index.html). However even if large networks are available, the number of networks with pre-known community structure or ground truth is limited. This limitation is over-passed by generating networks similar to real networks using the LFR

benchmark and the SBM. In this paper, we consider both real networks and simulated networks in order to evaluate the algorithms on the same basis.

While real-world networks correspond to real cases from observed links or interactions, generally there is no ground truth associated to them with regard to underlying communities. Synthetic networks offer this possibility and can be built for use as benchmarks [Barrett et al., 2009].

Table 2: Eight real world networks statistics - SNAP Stanford database.

Network	Number of Nodes	Number of Edges	Average Degree	Average Transitivity	Diameter
Net Science	1,589	2,742	3.45	0.878	17
Facebook	4,039	88,234	43.69	0.617	8
Power Grid	4,941	6,594	2.67	0.106	46
Astrophysics	16,706	121,251	14.52	0.726	14
Enron Email	36,692	183,831	10.02	0.715	13
Condensed Matter	40,421	175,692	8.69	0.718	18
DBLP	317,080	1,049,866	6.62	0.732	23
Amazon	334,863	925,872	5.53	0.429	47

Real world networks

One interesting real network collection is the SNAP Stanford database which contains eight real world networks where the number of vertices, edges, Average degree[1], global clustering coefficient and diameter[2] are reported in

[1]The *degree* of a node is the number of edges connected to it. The average degree is computed as $\frac{2E}{N}$)

[2]The *diameter* of a network is the longest of all the calculated shortest paths in a network. In other words, it is the shortest distance between the two most distant nodes in the network.

Table 2. *Facebook* dataset consists of Facebook profiles and friendships circles, *Power Grid* represents the topology of the western states power grid of the United States, *Enron Email* represents an email communication network, *Amazon* data set represents co-purchasing network of products. We also use the four co-authorship networks as follows: *Net Science, Astrophysics, Condensed Matter* and *DBLP,* representing the networks of co-authorship between scientists in science of networks, astrophysics e-print archive, condensed matter e-print archive and in computer science respectively.

From Table 2 we can see that the networks are quite varied. The number of nodes varies from a few thousands to a few hundred thousands while the number of edges varies from a few thousands to about one million. The average degree is also varied (from 2.66 for Power Grid to 43.69 for Facebook). Finally, the diameter of the networks varies from 8 (Facebook) to 47 (Amazon).

Synthetic networks

One of the earlier synthetic based networks is the Girvan-Newman (GN) benchmark [Girvan and Newman, 2002]. It makes possible to define equal size communities with a given degree and a given ratio between internal and outgoing connections. However, the resulting networks are quite homogeneous (equal size communities, same expected degree), the number of nodes is very limited and communities do not overlap. As a result, most of the detection algorithms works well on GN and it is not possible to GN to reveal the specific limitations of a given algorithm. GN is thus considered as providing non realistic networks [Yang et al., 2017]. LFR (Lancichinetti, Fortunato, Radicchi) benchmark [Lancichinetti et al., 2008] and SBM [Karrerand Newman, 2011] overtake these problems and for this reason are preferred to generate test bed and benchmarks to evaluate community detection methods.

LFR benchmark (https://sites.google.com/site/santofortunato/inthepress2) generalizes GN by using power-law distributions of degree and community size in the graph generation. LFR generates networks based on a set of parameters the user specifies. The resulting networks are with pre-known community structure where network size, node degree range and community size distributions are heterogeneous and power-law. Mixing parameter μ is used to

control the fraction of nodes a node shares inside and outside the community [Lancichinetti et al., 2008]. More specifically, for a given node, μ is defined as the fraction of the number of edges connecting it to other nodes belonging to different communities by the total degree of that node. In LFR, the mixing parameter μ is the same for any of the nodes. μ has been shown to be the most influential parameter and thus it makes sense to make it varying when comparing various algorithms [Orman and Labatut, 2009]; moreover, to keep the notion of communities, one should consider $\mu < 0.5$.

SBM (http://igraph.org/r/doc/sample_sbm) is a generative model for random graphs, generating networks with community structure where it is possible to choose the number of nodes in the network, the size of each community, and the community structure considering a CxC Bernoulli matrix, where C is the number of communities and Cij is the probability of an edge between nodes from community i and community j [Abbe, 2018]. The LFR benchmark is a special version of the stochastic block model [Fortunato and Hric, 2016]. SBM is considered as the simplest model of a graph with communities [Abbe, 2018].

LFR and SBM can thus generate an unlimited number of networks with predefined community structure simulating real world networks. In Section 5, when evaluating and comparing the algorithms on synthetic networks we considered SBM. Indeed, SBM is generally preferred as a test bed for algorithms [Goldenberg et al., 2010, McDaid and Hurley, 2010, Abbe and Sandon, 2015, Zhang et al., 2016, Caltagirone et al., 2017, Gulikers et al., 2017]. Synthetic networks and real networks complement well each other. Synthetic networks correspond to insightful models for which ground truth is known by construction, if not fully realistic. For example, *Orman et al.* report that the "distribution of links is not always appropriate" and "the small communities are too dense and clique-like" [Orman and Labatut, 2009].

Comparison of the Algorithms

In this section we evaluate and compare community detection algorithms surveyed in Section 2 on both large real world networks and artificially generated ones using SBM. The same "igraph" package was used to assess

the communities by internal and external measures. Calculation of conductance was implemented by us.

Results on real networks

We applied the eight state of the art community detection algorithms presented in Section 2 on the eight real world networks as presented in Section 4. Modularity, global clustering coefficient and conductance were used to assess the quality of the detected communities.

Modularity and number of detected communities

Table 3 reports the modularity scores of the partition while Table 4 reports the number of communities detected across real networks and algorithms. The reason why some columns in Table 3 and 4 have NA values is that the Spinglass algorithm cannot work with unconnected graphs and Edge betweenness algorithm was computationally extensive for few networks due to its high complexity (See Table 1).

Table 3: **Modularity** of the network when partitioned by each algorithm on real networks. For each network the value in bold highlights the best algorithm. The larger the value, the more the community structure is revealed.

Algorithm	Net Science	Facebook	Power Grid	Astro physics	Enron Email	Condensed Matter	DBLP	Amazon
Fast Greedy	0.955	0.777	**0.933**	0.633	N/A	0.632	0.735	0.880
Louvain	**0.960**	0.835	0.926	**0.727**	**0.605**	**0.722**	**0.820**	**0.926**
Infomap	0.929	0.810	0.817	0.658	0.130	0.631	0.722	0.825
Walktrap	0.957	0.812	0.831	0.636	0.512	0.599	0.672	0.849
Leading Eigenvector	0.951	0.799	0.825	0.595	0.439	0.359	0.024	0
Label Propagation	0.907	0.821	0.806	0.550	0.340	0.425	0.677	0.784
Spinglass	N/A	**0.835**	0.916	N/A	N/A	N/A	0.799	0.863
Edge betweenness	0.958	N/A	0.933	N/A	N/A	N/A	N/A	N/A

Table 4: **Number of communities** detected by each algorithm on real networks.

Algorithm	Net Science	Facebook	Power Grid	Astro physics	Enron Email	Condensed Matter	DBLP	Amazon
Fast Greedy	403	13	41	1,168	N/A	2,253	3,077	1,480
Louvain	406	17	40	1,080	1,319	1,876	275	249
Infomap	442	93	492	1,860	10,291	3,816	17,023	17,286
Walktrap	416	77	364	2,673	3,066	5,252	30,425	14,905
Leading Eigenvector	404	18	35	1,067	1,068	1,801	2	1
Label Propagation	456	53	481	1,506	1,989	3,381	21,040	22,532
Spinglass	N/A	23	25	N/A	N/A	N/A	25	25
Edge betweenness	405	N/A	45	N/A	N/A	N/A	N/A	N/A

Modularity reported in Table 3 varies from 0.024 (Leading Eigenvector on DBLP) to 0.960 (Louvain on Net Science). The various algorithms obtain similar and high modularity (from 0.907 to 0.960) on Net Science network which is the smallest network in terms of number of nodes and edges and one of the networks that have small average degree. Results are also high and similar (0.806 to 0.933) for the algorithms on Power Grid network which is quite similar to Net Science in terms of number of nodes, edges and average degree.

Larger networks such as DBLP and Amazon, which are also not very dense, obtain higher modularity for most of the algorithms. The lowest modularity values occurs on Enron Email, Astrophysics and Condensed Matter networks for which average degree is between 8 and 14, there are a few ten thousand nodes and a few hundred thousand edges.

From the Table 3 we can see that partitions obtained by Louvain have consistently high modularity scores across the networks, indicating that the network partitions are more community-like. Fast Greedy, Infomap and Walktrap algorithms also have high modularity scores.

The algorithms also differ in terms of the number of communities being detected. Table 4 reports this number across collections and algorithms. Fast greedy, Louvain and Leading eigenvector methods detect small number of

communities while Infomap, Walktrap and Label propagation detect more. On the smallest network (Net Science) all the algorithms detect about the same number of communities but not on Power Grid. While the two networks are similar in terms of number of edges, nodes and degree, they differ in terms of diameter (17 for Net Science and 46 for Power Grid). On Power Grid network that has a longer shortest distance between the two most distant nodes than Net Science, Infomap, Walktrap and Label propagation algorithms detect a few hundred communities while the other algorithms detect rather a few tens. This result is not surprising considering the propagation methods used in these three algorithms. On Amazon and DBLP networks which also have a large diameter, the number of communities detected by the algorithms also varies a lot with Infomap, Walktrap and Label propagation, consistently detecting much more communities than the other algorithms. Although the variation is much higher when the diameter is larger, these three algorithms tend to detect more communities than the other algorithms whatever the networks are.

Conductance and global clustering coefficient

In this sub-section, we analyze the results obtained for the two other internal measures. The results are presented in two tables which present the percentage of detected communities for which the global clustering coefficient (T) (Table 5) and conductance (C) (Table 6) satisfy the inequality given in the left part of the tables. Values in bold correspond to the highest percentage of communities in particular intervals. We consider four intervals for the values of C and T in order to make categories related to the quality of the detected communities.

The higher the global clustering coefficient T and the lower the conductance C, the better the community structure. So when $C \in$ (0; 0.01) and/or $T \in$ (0.75; 1), the community structure is good while for $C \in$ (0.1; 1) and/or $T \in$ (0; 0.25), the community structure is weak. Tables 5 and 6 are ordered from the best community structure to the weakest according to the measure used. In these tables, we did not report Spinglass nor Edge betweenness because they had too many NA values.

For example, we can read in Table 5 that on Net Science network, Fast Greedy algorithm gets 69.9% of the communities it detects having a global clustering coefficient T higher than 0.75,18.4% of the detected communities with clustering coefficient in between 0.5 and 0.75, 5.2% in between 0.25 and 0.5 and 6.3% lower than 0.25. Fast Greedy is a good algorithm when considering global clustering coefficient and Net Science network. Reversely, on Amazon, Leading Eigenvector gets 100% of the communities it detects with global clustering coefficient lower than 0.25; this algorithm fails detecting the communities on Amazon network.

The algorithms all provide good structure communities with low conductance and high global clustering coefficient on Net Science network (the smallest network). Reversely, most of them fail on Power Grid (high conductance, low global clustering coefficient). Power Grid is the only network that has a very low average transitivity; although we cannot make a clear correlation with this feature, it might be a reason.

In Table 6, we also can see that Fast Greedy, Louvain, and Leading EigenVector algorithms get low conductance for most of the communities they detected and this across most of the real networks we used. These algorithms also have the highest global clustering coefficient (see Table 6). These algorithms are thus the ones that are considered the best with regard to conductance and global clustering coefficient.

When considering global clustering coefficient only, the next best algorithms are Walktrap and Label propagation (but they are in the worth when considering conductance). On the other hand, Infomap, Walktrap and label propagation have the worth (high) conductance and Infomap is also in the worth category when considering global clustering coefficient. When $T \in$ (0.75; 1) we have a good community structure while for $T \in$ (0;0.25) community structure is weak.

As we can see in Table 5 and Table 6 based on global clustering coefficient and conductance, Louvain, Fast greedy, Leading eigenvector and Label propagation algorithms were able to detect mostly good communities (i.e. highest global clustering coefficient and lowest conductance).

Table 5: Percentages of communities for which the **conductance** score is in the respective interval on real networks. The top part of the table indicates good community structure according to conductance, while the conductance is lower as we go on the bottom of the table. Low conductance (top part of the table) shows a good community structure while high conductance reflects a weak community structure (bottom part of the table).

C	Algorithm	Net Science	Facebook	Power Grid	Astro physics	Enron Email	Condensed Matter	DBLP	Amazon
$C \in [0; 0.01)$	Fast Greedy	**97.4**	23	7.3	**72.6**	NA	**67.6**	0.2	7.3
	Louvain	**96.4**	23.5	10	**88**	**80.8**	**92.4**	1.8	24.8
	Infomap	**83.4**	0	0	30.4	8.5	32	0.06	0.5
	Walktrap	**91.6**	0	0.2	17.2	33.4	20.4	0.02	0.5
	Leading Eig.	**97.1**	16.6	2.8	**90.4**	**99.6**	**99.5**	**50**	**100**
	Label Prop.	**77.2**	0	0	32.6	**52.6**	27.4	0.02	0.2
$C \in [0.01; 0.02)$	Fast Greedy	0	0	7.3	0.3	NA	0.2	1.3	10.9
	Louvain	0	0	10	0.4	0.3	0	4.7	11.6
	Infomap	0	0	0	0.3	0.01	0.2	0.4	1.8
	Walktrap	0.3	0	0	0.1	0.1	0.04	0.09	1.8
	Leading Eig.	0	5.5	2.8	0.2	0	0	0	0
	Label Prop.	0.3	0	0	0.5	0.1	0.1	0.1	0.9
$C \in [0.02; 0.1)$	Fast Greedy	2.5	**61.5**	**82.9**	6.2	NA	5.4	33.4	59.1
	Louvain	3.5	**58.8**	**80**	3.5	9.7	2.2	**50.9**	**59.4**
	Infomap	6.6	17.7	12	4.4	0.03	5.2	14	25.4
	Walktrap	6.5	18.1	16.2	1.6	4.5	3	5.2	28.7
	Leading Eig.	2.1	**38.8**	**57.1**	0.9	0	0.1	0	0
	Label Prop.	4.8	28.8	8.1	3.7	7.5	3.9	7.6	15.9
$C \in [0.1; 1]$	Fast Greedy	0	15.3	2.4	20.6	NA	26.6	**64.9**	22.5
	Louvain	0	17.6	0	7.8	9	5.3	42.5	4
	Infomap	9.8	**82.2**	**87.9**	**64.7**	**91.4**	**62.4**	**85.4**	**72**
	Walktrap	1.3	**81.8**	**83.5**	**81**	**61.8**	**76.4**	**94.5**	**68.9**
	Leading Eig.	0.7	**38.8**	37.1	8.3	0.3	0.3	**50**	0
	Label Prop.	17.5	**71.1**	**91.8**	**63**	39.6	**68.5**	**92.1**	**82.8**

Table 6: Percentage of communities for which the global clustering coefficient is in the respective interval on real networks. High global clustering coefficient shows a good community structure (top part of the table) while low global clustering coefficient reflects a weak community structure (bottom part of the figure).

T	Algorithm	Net Science	Facebook	Power Grid	Astro physics	Enron Email	Condensed Matter	DBLP	Amazon
$T \in [0.75;1]$	Fast Greedy	**69.9**	38.4	0	**63.7**	NA	**57.2**	**36.9**	18.4
	Louvain	**69.3**	29.4	0	**67**	**57**	**61.8**	19.2	0.4
	Infomap	65	**40.6**	0.4	**39.6**	23	29.4	17	9.8
	Walktrap	**67.2**	**40.5**	0.5	**58**	**59.9**	**46.6**	**34.8**	17.9
	Leading Eig.	**69.5**	23.5	0	**68.7**	**61.2**	**68.5**	0	0
	Label Prop.	**69.3**	**43.1**	3	**68.2**	**66.7**	**55.9**	**43.4**	20.4
$T \in [0.5;0.75)$	Fast Greedy	18.4	15.3	0	19.4	NA	23	31.7	36.7
	Louvain	19.3	**35.2**	0	20.1	15.3	17.3	29.8	12.8
	Infomap	**23.1**	27.9	2	33.9	2.8	**35**	30.5	27
	Walktrap	21.5	31.8	1.6	20.4	14.2	25.2	26.1	30.8
	Label Prop.	20.2	39.6	3.8	19.6	14.6	24.6	26.4	31.8
	Leading Eig.	18.9	23.5	2.8	18.1	7.9	15.3	**50**	0
$T \in [0.25;0.5)$	Fast Greedy	5.2	**46.1**	9.7	7.5	NA	9.3	20.9	**36.8**
	Louvain	5.1	**35.2**	5	4.6	5.2	10.1	**48**	**68.2**
	Infomap	6.6	22	10	20.1	1	29.1	**39**	**45.3**
	Walktrap	5.3	21.7	6.7	8.9	6.2	15.3	20.3	**33**
	Leading Eig.	5.1	**52.9**	8.5	4.5	1.1	4.4	**50**	0
	Label Prop.	4.9	17.2	10.1	6.9	4.5	13.4	20.9	**35.7**
$T \in [0;0.25)$	Fast Greedy	6.3	0	**90.2**	9.2	NA	10.3	10.2	7.9
	Louvain	6.2	0	**95**	8.1	22.2	10.6	2.9	18.4
	Infomap	5.1	9.3	**87.4**	6.2	**73**	6.3	13.4	17.7
	Walktrap	5.9	5.7	**90.9**	12.4	19.5	12.8	18.5	18.1
	Leading Eig.	6.3	0	**88.5**	8.6	29.6	11.6	0	**100**
	Label Prop.	5.4	0	**83**	5	14	5.9	9.1	11.9

External Measures

Finally, the external measures NMI, VI and ARI were used to compare partitions obtained by different algorithms. We calculated the values for the partitions obtained by each pair of algorithms. In Table 7, we report the highest values only.

Pairs of algorithms which partitions resulted in the highest NMI, VI and ARI scores for each network are reported in Table 7. When considering NMI (first row in the Table), we can see that on 4 networks, namely Astrophysics networks, Enron email, Condensed matter, and DBLP, the partitions obtained by Infomap and Walktrap algorithms are the most similar. Infomap and Label propagation obtain similar results on two other networks (Power Grid and Amazon networks). On 5 over 8 networks, the same couples of algorithms give the highest NMI and VI, this is on Net science, Facebook, Astrophysics, Condensed matter and Amazon networks. Similar results indicate the analogy of NMI and VI.

Table 7: Pairs of algorithms for which the score is the highest for a given measure and a network - Using real networks and the three measures: normalized mutual information, variation of information and adjusted rand index.

External Measure	Net Science	Facebook	Power Grid	Astrophysics	Enron Email	Condensed Matter	DBLP	Amazon
NMI	0.992 Louvain & Fast Greedy	0.937 Louvain & Spinglass	0.903 Infomap & Label prop.	0.848 Infomap & Walktrap	0.743 Infomap & Walktrap	0.835 Infomap & Walktrap	0.855 Infomap & Walktrap	0.942 Louvain & Label prop.
VI	0.083 Louvain & Fast Greedy	0.344 Louvain & Spinglass	0.997 Louvain & Fast Greedy	1.965 Infomap & Walktrap	2.298 Louvain & Label prop.	2.361 Infomap& Walktrap	1.789 Infomap& Label prop.	1.102 Infomap& Label prop.
ARI	0.898 Louvain & Edge betweenness	0.914 Louvain & Spinglass	0.680 Louvain & Fast Greedy	0.151 Walktrap & Fast Greedy	0.238 Louvain & Walktrap	0.216 Walktrap & Fast Greedy	0.238 Fast Greedy& Label prop.	0.587 Infomap & Label prop.

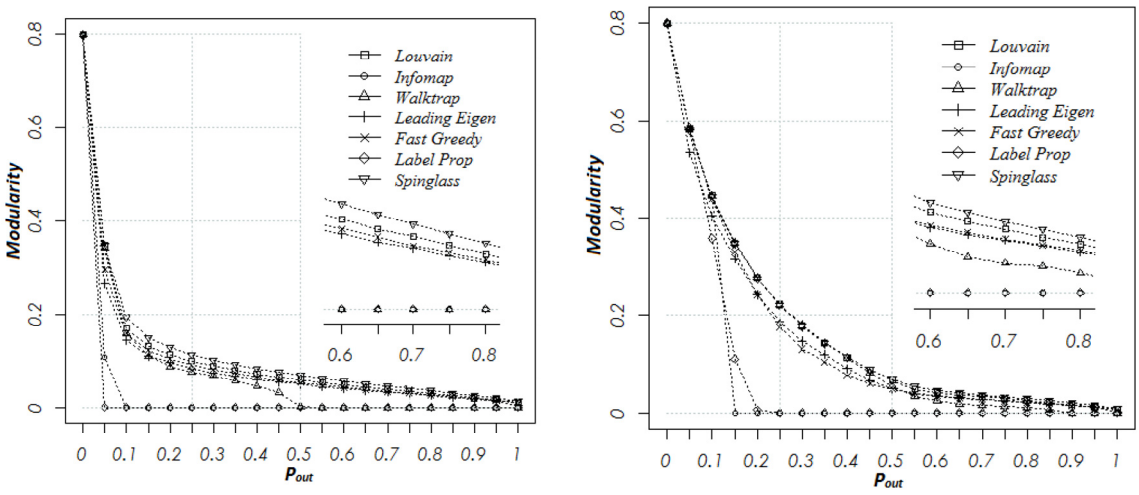
Results on artificial networks

In this section, we used SBM to generate random networks with community structure, where the number of communities, community sizes and probabilities of edges between communities and inside communities are known *a priori.*

Using such networks help in analyzing the impact of the network properties on the results of algorithms or vice versa. In our experiments we generated more than 400 SBMs, where the number of nodes is 200 and they are divided into 5 equally sized communities. We did our experiments for P_{in} = 0.25 (weak connectivity inside communities) and P_{in} = 0.75 (strong connectivity inside communities).

Modularity

Fig. 1 (a) displays the modularity scores of the partitions based on the probability of edges between communities (P_{out}) for P_{in} = 0.25 (weak connectivity inside communities) while Fig. 1 (b) is for P_{in} = 0.75 (strong connectivity inside communities).



(a) For $P_{in} = 0{:}25$ (low connectivity inside communities).

(b) For $P_{in} = 0{:}75$ (strong connectivity inside communities).

Figure 1: Probability of edges between communities (P_{out}) versus modularity for a fixed P_{in}. A zoom on curves when $P_{out} \in [0{:}6; 0{:}8]$ is provided in the right bottom part.

In both Figures (Fig.1 (a), Fig. 1 (b)) for each P_{out} = 0.05n, where $n \in$ {0,1,20}, we generated 100 SBMs and averaged the values of modularity of the partitions. As the pattern of the majority of the algorithms are close to each other, a zoom is provided on the denser interval $P_{out} \in$ [0.6,0.8] to distinguish the curves for the different algorithms easily.

Results show that when the probability of edges between communities (P_{out}) increases, the partitions become less community-like resulting in smaller modularity values. However for any P_{out} value, Louvain and Spinglass methods still detect community structure with highest modularity compared with other methods. With an increase of P_{out}, Infomap, Walktrap and Label Propagation algorithms fail to detect communities (i.e. modularity of the partition becomes

zero at $P_{out} \geq 0.1$ for Infomap, P_{out} = 0.5 for Walktrap and P_{out} = 0.05 for Label Propagation).

Processing time

We also evaluated algorithms based on the processing time and the number of nodes in the network; results are displayed in Fig. 2. From Fig. 2 that reports the relationship between P_{out} and processing times of the algorithms, we can see that when Pout increases, the processing time of the majority of the algorithms increases as the network structure becomes more dense. Moreover, in the same Fig. 2, we see that Infomap is the most time consuming algorithm whatever the probability of edges between communities (P_{out}) is. On the contrary, processing time for Label propagation is constant with regard to the P_{out} value. The processing time for Spinglass algorithm is not displayed in the figure and is much longer than for other methods.

Finally Fig. 3 gives the relationship between the number of nodes in the network and the processing time of the algorithms. We can see that when the number of vertices in the network increases, processing time for Spinglass, Fast Greedy and Walktrap increases the most, while the other methods detect community structure in a relatively constant time.

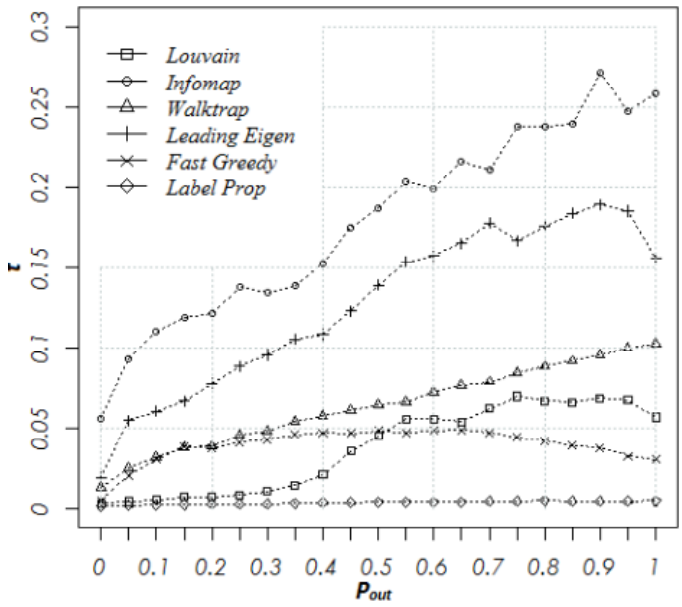


Figure 2: Probability of edges between communities (P_{out}) versus processing time in seconds (t) for P_{in} = 1 (Strong community structure).

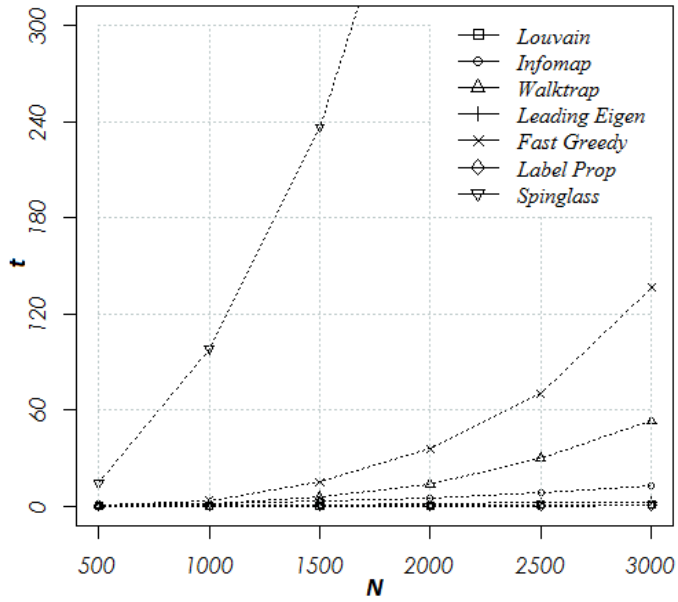


Figure 3: Number of vertices (N) versus time in seconds (t)

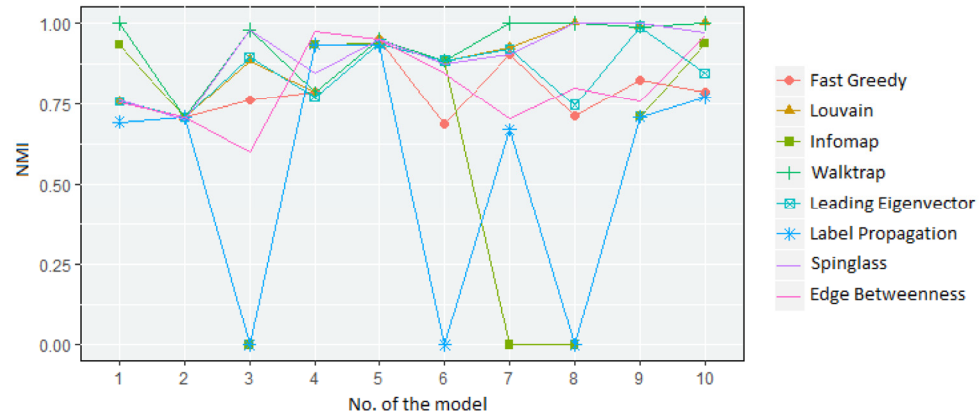
Comparison with ground truth

To complete the comparison of the algorithms, we evaluated and compared the community structures the algorithms detected with pre-known ground truth using stochastic block model.

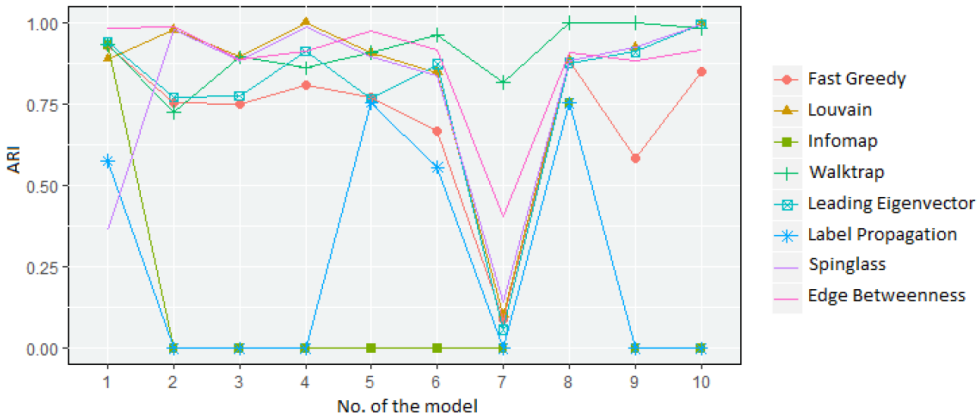
From the definition of SBM we already know that networks are generated based on the CxC Bernoulli matrix, where C_{ij} is the probability of an edge between nodes from community i and community j and relative sizes of communities. Thus specifying higher probabilities for the edges inside communities ($C_{ij} \in (0.5,1)$) and lower probabilities for the edges between communities (($C_{ij} \in (0,0.25)$), where $i = j$) and taking random numbers for relative community sizes which sum up to 200, we can generate an unlimited number of ground truth community structures.

The results obtained on a sample of 10 random networks are presented in Fig.4. We report NMI, ARI, and VI scores.

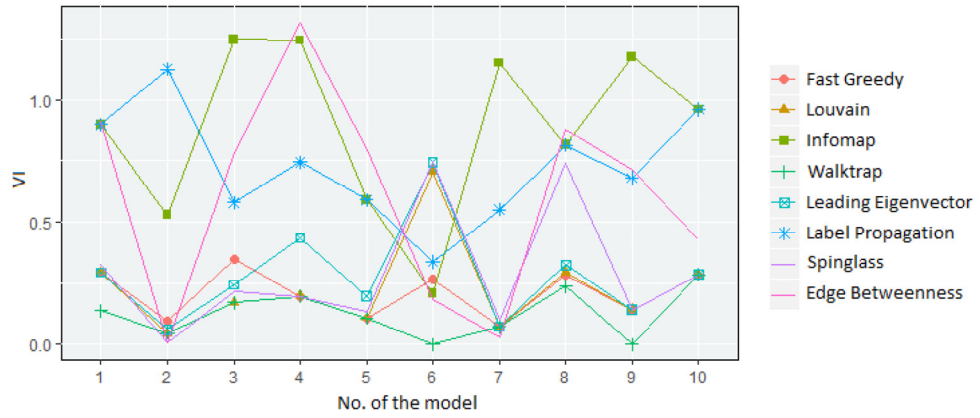
On Fig. 4 (a), we can see that the results regarding normalized mutual information are unstable for Label propagation and Infomap algorithms; while they can achieve good results when applied to some network structures, they can fail on others. Moreover, they never clearly overtake the other algorithms.



(a) Normalized mutual information



(b) Adjusted rand index



(c) Variation of information

Figure 4: NMI, ARI, and VI scores on 10 sample networks using the 8 algorithms when comparing detected community structures and ground truth.

Other algorithms are more stable. We can mention the high performance of Louvain, Walktrap and Spinglass algorithms and slightly lower performance for Fast Greedy. With regard to adjusted rand index displayed on Fig. 4 (b), the results are similar with some instability for Label propagation and Infomap algorithms, good performance for the other algorithms, but slightly better for Louvain, Walktrap and Spinglass, while Fast Greedy is slightly lower. Finally the results of variation of information are displayed on Fig. 4 (c). We can again see the instability among some algorithms (Edge betweenness, Infomap and Label propagation) while in the majority of the cases Walktrap, Spinglass and Louvain algorithms outperform the others. The similarity in the results obtained by NMI and VI comes from their analogy as they both have mutual information component in common.

We also report the numerical values obtained when averaged over the 100 random networks in Table 8. From Table 8 we entail that Walktrap algorithm overtakes all other algorithms considering the three measures when identifying ground truth community structure, although Louvain and Spinglass are very close, followed by Leading Eigenvector.

Table 8: Normalized mutual information, adjusted rand index and variation of information averaged over 100 trials (random networks) when comparing the community structures detected by the eight algorithms and the ground truth. Best results are in bold.

Agorithm Measure	Fast Greedy	**Louvain**	InfoMap	**Walktrap**	Leading Eig.	Label Prop	**Spinglass**	Edge between
NMI	0.791	0.879	0.471	**0.939**	0.839	0.444	0.883	0.807
ARI	0.794	0.883	0.449	**0.937**	0.849	0.404	0.889	0.891
VI	0.356	0.216	0.641	**0.114**	0.291	0.657	0.220	0.527

Summary of the results

On real world networks, Louvain algorithm achieves the detection of communities which are densely connected inside communities and sparsely connected between communities (almost always the highest modularity score), although it leads to detecting much less communities on some networks compared to other algorithms, specifically on the large networks and on networks with large diameter (Power Grid, DBLP, and Amazon). Louvain algorithm remains both effective and efficient also when the probability of edges between communities increases (results on artificial networks). Our experiments vote in for Louvain algorithm. Fast Greedy, Leading Eigenvector and Label propagation are also good algorithms. Spinglass and Edge betweenness suffer from their complexity. Infomap and Walktrap are weak on too many measures, while Walktrap was the most effective one when detecting ground truth communities on synthetic networks.

Table 9: Overview of the results. ++ indicates that the algorithm has good properties according to the measure while -that it has low performance.

Algorithm	Complexity	Modularity	Number of communities	Conductance	Clusteringcoefficient	High P_{out} Modularity	High P_{out} Time	Comparisonground truth
Fast Greedy	+	++	Small	++	+			+
Louvain	++	++	Small	++	++	++		++
Infomap	+	++	Large	--	-	--	--	--
Walktrap	-	++	Large	-	+	--		++
Leading Eigenvector	+	+	Small	++	++			+
Label propagation	++	+	Large	+	++	--	++	--
Spinglass	--	NA	Small	NA	NA	++		++
Edge betweenness	-	NA	NA	NA	NA			+

Related Work

Several papers consists in community detection survey but many do not compare them [Porter et al., 2009, Coscia et al., 2011, Fortunato and Hric,

2016, Bedi and Sharma, 2016] while a few others made comparison but either on fewer measures than our survey, or on either real or synthetic networks [Danon et al., 2005, Fortunato, 2010, Leskovec et al., 2010, Orman et al., 2011, de Sousa and Liang, 2014, Yang et al., 2017].

In these studies, the authors did not aim at deeply comparing the algorithms. For this reason, we do not mention them in this section nor in the Table 10 where an overview of survey papers is reported.

Table 10: Overview of the literature. For each of the reviewed papers (first column), we report the number of algorithms that are reviewed in that paper (2nd column), the number of real networks used (3rd column), the type or number of synthetic networks used (4th column), the measures used to compare algorithms (5th column), and the outcome in terms of the best algorithms mentioned by the paper (6th column).

Survey	Algorithms	Real networks	Synthetic net.	Measures	Best
[Danon et al., 2005]	16 algorithms	None	GN	Modularity	
[Fortunato et al., 2010]	12 algorithms	None	LFR various μ	NMI, Precision	
[Leskovec et al., 2010]	2 algorithms	Yes	None	Conductance & Modularity	
[Orman et al., 2011]	5 algorithms	None	LFR,10k & 100k nodes	NMI, Topology	Infomap & Walktrap
Moradi et al., 2014 [de Sousa and Liang, 2014]	8 igraph algo.	3 real	Yes (4 own)	Modularity & time	Walktrap, Spinglass
[Yang et al., 2017]	8 igraph algo.	None	LFR, 200 to 32k nodes various μ	NMI, $\overline{C}/C$, Time	Louvain
[Chakrabortyet al., 2017]	6 algorithms	6 real	LFR	VI, NMI, ARI, F, Purity	
Our contribution	8 algorithms	8 real	SBM various P_{in} and P_{out}	Modularity Conductance & NMI	Louvain, Leading Eig. Fast Greedy

One of the earlier studies worth mentioning is Danon et al. [Danon et al., 2005]. The authors compare 16 algorithms mainly on modularity and fraction of nodes correctly identified on GN network. The authors conclude that most of the methods are very good at detecting the structure, but the synthetic network used to evaluate the algorithm was quite homogeneous and small (state of the art test bed set at that time).

In [Leskovec et al., 2010], Leskovec et al. first compare two graph partitioning algorithms namely the Local Spectral Partitioning algorithm [Andersen et al., 2006] and the flow-based Metis+MQI algorithm[Karypis and Kumar, 1998] considering conductance evaluation measure using several real networks including DBLP, Enron email and Astrophysics networks. The authors also briefly report the comparison of other algorithms still on conductance plus community score.

The survey by Fortunato [Fortunato, 2010] is probably still the largest reported study (100 pages long). This survey also suggests a classification of algorithms based on the underlying type of method they use; classes are in agreement with the ones previously mentioned.

Orman et al. uses LFR benchmark and generated 3 different network structures [Orman et al., 2011]. The authors first analyzed the properties of the synthetic network generated as compared to real networks. They found out that while it is possible to generate networks that are realistic in terms of size, the distribution of links is not always appropriate. For example, the small communities seem to be too dense and clique-like. They evaluated 4 of the 8 algorithms we also evaluated (namely Louvain, Fast Greedy, Infomap, and Walktrap) and added MarkovCluster. When evaluated on networks either composed of 10,000 and 100,000 nodes and using NMI, the authors found out that Infomap got the best results while Fast Greedy got the lowest. When considering the topology of the network as extracted by the algorithm and when compared to the ground truth, Waltrap seemed to reflect it better than the other algorithms.

Moradi et al., 2014 [de Sousa and Liang, 2014] evaluated the 8 algorithms from igraph package on 3 small real networks (ZacharyaAZs Karate Club, Football

Network, and NetScience) and 4 small synthetic networks they built (up to 100 nodes). They found that Infomap and Walktrap are the best when considering NMI and the topology of the extracted structure when compare to the ground truth structure. However, the size of the networks used are not fully realistic when compare to current real networks.

Yang et al. is one of the most recent survey [Yang et al., 2017]. In their paper, they compared the same 8 algorithms that we used in this paper and consider LFR synthetic networks making the networks varying from 200 to 32,000 nodes and also making the structure of the network varying through the μ parameter. To evaluate the algorithms, [Yang et al., 2017] considered NMI, $\overline{C}/C$ where $\overline{C}$ is the average number of detected communities the algorithm extracted (the experiment was repeated using 100 different networks) and C is the average number of communities in the same networks as given by the LFR benchmark, and processing time. The authors report that the number of nodes has little impact on the NMI apart for leading eigenvector algorithm. They also found that Label propagation and Louvain algorithms are much faster than the others. Finally, they concluded that Louvain algorithm is the best when considering the various aspects. While our study is more complete in terms of the measures we used as well as the type of networks we used since we used both real and synthetic networks, we could make the same conclusion on that algorithm.

Conclusion

In this paper we surveyed state of the art traditional community detection algorithms, internal evaluation measures to assess the quality of community structure and external evaluation measures to compare partitions. We applied the algorithms on eight real world networks and on artificially generated stochastic block model.

Thereafter we used internal evaluation measures such as conductance, global clustering coefficient and modularity to quantitatively assess the detected community structure and external evaluation measures such as normalized mutual information, variation of information, and adjusted rand index to compare network partitions of each algorithm. We also compared the algorithms based on performance (processing time). Overall, Louvain algorithm appears to be the

most robust algorithm across the real and synthetic networks and across measures. Fast Greedy and Leading Eigenvector algorithms are also interesting alternatives.

Acknowledgment

The work benefited from the support of the European program Erasmus+ Convention KA107 No036342.

Bibliography

[Abbe, 2018] Abbe, E. (2018). Community detection and stochastic block models. Foundations and Trends in Communications and Information Theory, 14(1-2):1-162.

[Abbe and Sandon, 2015] Abbe, E. and Sandon, C. (2015). Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on, pages 670-688. IEEE.

[Andersen et al., 2006] Andersen, R., Chung, F., and Lang, K. (2006). Local graph partitioning using pagerank vectors. In Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, pages 475-486.

[Barrettetal.,2009] Barrett, C. L., Beckman, R. J., Khan, M., Kumar, V. A., Marathe, M. V., Stretz, P. E., Dutta, T., and Lewis, B. (2009). Generation and analysis of large synthetic social contact networks. In Simulation Conference (WSC), Proceedings of the 2009 Winter, pages 1003-1014. IEEE.

[Bedi and Sharma, 2016] Bedi, P. and Sharma, C. (2016). Community detection in social networks. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 6(3):115-135.

[Blondel et al., 2008] Blondel, V., Guillaume, J., and Lambiotte, R. (2008). Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 2008.

[Caltagirone et al., 2017] Caltagirone, F., Lelarge, M., and Miolane, L. (2017). Recovering asymmetric communities in the stochastic block model. IEEE Transactions on Network Science and Engineering.

[Chakrabortyetal., 2017] Chakraborty, T., Dalmia, A., Mukherjee, A., and Ganguly, N. (2017). Metrics for community analysis: A survey. ACM Comput. Surv., 50:54:1-54:37.

[Clauset et al., 2004] Clauset, A., Newman, M., and Moore, C. (2004). Finding community structure in very large networks. Phys. Rev. E 70,70(066111).

[Collignon et al., 1995] Collignon, A., Maes, F., Delaere, D., Vandermeulen, D., Suetens, P., and Marchal, G. (1995). Automated multi-modality image registration based on information theory. volume 3, pages 263-274.

[Coscia etal., 2011] Coscia, M., Giannotti, F., and Pedreschi, D. (2011). A classification for community discovery methods in complex networks. Statistical Analysis and Data Mining: The ASA Data Science Journal, 4(5):512-546.

[Cover and Thomas, 2006] Cover, T. and Thomas, J. (2006). In Elements of Information Theory, 2nd Edition. Wiley.

[da Costa Alves, 2010] da Costa Alves, C. S. (2010). Social network analysis for business process discovery. The Technical University of Lisbon.

[Danon et al., 2005] Danon, L., Diaz-Guilera, A., Duch, J., and Arenas, A. (2005). Comparing community structure identification. Journal of Statistical Mechanics: Theory and Experiment, 2005(09):P09008.

[de Sousa and Liang, 2014] de Sousa, F. B. and Liang, Z. (2014). Evaluating and comparing the igraph community detection algorithms. In Intelligent Systems (BRACIS), 2014 Brazilian Conference on, pages 408-413. IEEE.

[Deng et al., 2016] Deng, S.-P., Zhu, L., and Huang, D.-S. (2016). Predicting hub genes associated with cervical cancer through gene co-expression networks. IEEEACM Transactions on Computational Biology and Bioinformatics (TCBB), 13:27-35.

[Fortunato, 2010] Fortunato, S. (2010). Community detection in graphs. Physics Reports, 486:75-174.

[Fortunato and Hric, 2016] Fortunato, S. and Hric, D. (2016). Community detection in networks: A user guide. Physics Reports, 659:1-44.

[Girvan and Newman, 2002] Girvan, M. and Newman, M. E. (2002). Community structure in social and biological networks. Proceedings of the national academy of sciences, 99(12):7821-7826.

[Girvan and Newman, 2001] Girvan, M. and Newman, M. E. J. (2001). Community structure in social and biological networks. PNAS, 99(12):7821-7826.

[Goldenberg et al., 2010] Goldenberg, A., Zheng, A. X., Fienberg, S. E., Airoldi, E. M., et al. (2010). A survey of statistical network models. Foundations and Trends® in Machine Learning, 2(2):129-233.

[Gulikersetal., 2017] Gulikers, L., Lelarge, M., and Massoulié, L. (2017). A spectral method for community detection in moderately sparse degree-corrected stochastic block models. Advances in Applied Probability, 49(3):686-721.

[Halkidi etal., 2002] Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2002). Clustering validity checking methods: part ii. ACM SIGMOD Record, 31(3):19-27.

[He et al., 2018] He, K., Li, Y., Soundarajan, S., and Hopcroft, J. E. (2018). Hidden community detection in social networks. Information SciencesaATInformatics and Computer Science, Intelligent Systems, Applications: An International Journal, 425(C).

[Hu et al., 2008] Hu, Y., Chen, H., Zhang, P., Li, M., Di, Z., and Fan, Y. (2008). Comparative definition of community and corresponding identifying algorithm. Phys. Rev. E, 78(026121).

[Hubert and Arabie, 1985] Hubert, L. and Arabie, P. (1985). Comparing partitions. Journal of Classification, 2(1):193-218.

[Karrer and Newman, 2011] Karrer, B. and Newman, M. E. (2011). Stochastic blockmodels and community structure in networks. Physical review E, 83(1):016107.

[Karypis and Kumar, 1998] Karypis, G. and Kumar, V. (1998). A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM Journal on scientific Computing, 20(1):359-392.

[Lancichinetti and Fortunato, 2011] Lancichinetti, A. and Fortunato, S. (2011). Limits of modularity maximization in community detection. Physical Review E, 84(066122).

[Lancichinetti et al., 2008] Lancichinetti, A., Fortunato, S., and Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms. Physical review E, 78(4):046110.

[Leskovec and Krevl, 2014] Leskovec, J. and Krevl, A. (2014). SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data.

[Leskovec et al., 2008] Leskovec, J., Lang, K., Dasgupta, A., and Mahoney, M. (2008). Statistical properties of community structure in large social and information networks. Proceedings of the 17th international conference on World Wide Web, 3733(5):695-704.

[Leskovec et al., 2010] Leskovec, J., Lang, K. J., and Mahoney, M. (2010). Empirical comparison of algorithms for network community detection. In Proceedings of the 19th international conference on World wide web, pages 631-640. ACM.

[Leskovec and Yang, 2015] Leskovec, J. and Yang, J. (2015). Defining and evaluating network communities based on ground-truth. Knowledge and Information Systems, 42(1):181-213.

[Mahmoud et al., 2014] Mahmoud, H., Masulli, F., Rovetta, S., and Russo, G. (2014). Community detection in protein-protein interaction networks using spectral and graph approaches. Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB), 8452.

[McDaid and Hurley, 2010] McDaid, A. and Hurley, N. (2010). Detecting highly overlapping communities with model-based overlapping seed expansion. In Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on, pages 112-119. IEEE.

[Meila, 2007] Meila, M. (2007). Comparing clusteringsaAfan information based distance. Journal of Multivariate Analysis, 98(5):873-895.

[Moradi et al., 2012] Moradi, F., Olovsson, f., and fsigas, P. (2012). An evaluation of community detection algorithms on large-scale email traffic. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 7276:283-294.

[Newman, 2006] Newman, M. (2006). Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E, 74(036104).

[Newman and Girvan, 2004] Newman, M. and Girvan, M. (2004). Finding and evaluating community structure in networks. Phys. Rev. E, 69(026113).

[Orman and Labatut, 2009] Orman, G. and Labatut, V. (2009). A comparison of community detection algorithms on artificial networks. Lecture Notes in Computer Science (LNCS), pages 242-256.

[Orman et al., 2011] Orman, G., Labatut, V., and Cherifi, H. (2011). Qualitative comparison of community detection algorithms. Communications in Computer and Information Science, Springer, Berlin, Heidelberg, 167:265-279.

[Parimi and Caragea, 2014] Parimi, R. and Caragea, D. (2014). Community detection on large graph datasets for recommender systems. 2014 IEEE International Conference on Data Mining Workshop, pages 589-596.

[Pons and Latapy, 2005] Pons, P. and Latapy, M. (2005). Computing communities in large networks using random walks. Computer and Information Sciences - ISCIS 2005, pages 284-293.

[Porter et al., 2009] Porter, M. A., Onnela, J.-P., and Mucha, P. J. (2009). Communities in networks. Notices of the AMS, 56(9):1082-1097.

[Radicchi et al., 2004] Radicchi, F., C.Castellano, Cecconi, F., Loreto, V., and Parisi, D. (2004). Defining and identifying communities in networks. PNAS, 101(9):2658aAS2663.

[Raghavan et al., 2007] Raghavan, U., Albert, R., and Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. Physical Review E, 76(036106).

[Rand, 1971] Rand, W. (1971). Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association, 66(336):846-850.

[Reichardt and Bornholdt, 2006] Reichardt, J. and Bornholdt, S. (2006). Statistical mechanics of community detection. Phys. Rev. E, 74(016110).

[Rosvall and Bergstrom, 2007] Rosvall, M. and Bergstrom, C. (2007). Maps of random walks on complex networks reveal community structure. PNAS, 105(4):1118-1123.

[Wasserman and Faust, 1994] Wasserman, S. and Faust, K. (1994). Social Network Analysis: Methods and Applications. Cambridge: Cambridge University Press.

[Xie et al., 2013] Xie, J., Kelley, S., and Szymanski, B. K. (2013). Overlapping community detection in networks: The state-of-the-art and comparative study. Acm computing surveys (csur), 45(4):43.

[Yang et al., 2017] Yang, Z., Algesheimer, R., and Tessone, C. J. (2017). A comparative analysis of community detection algorithms on artificial networks. Scientific Reports, 6(30750):Jun.

[Zhang et al., 2016] Zhang, A. Y., Zhou, H. H., et al. (2016). Minimax rates of community detection in stochastic block models. The Annals of Statistics, 44(5):2252-2280.

Authors' Information

Karen Mkhitaryan *- PhD student, Institute for Informatics and Automation Problem,*

0014, Yerevan, Armenia;

e-mail*: karenmkhitaryan@gmail.com*

Major Fields of Scientific Research*: Network Science, Community Detection*

Josiane Mothe *- Professor, Institutde Recherche en Informatique de Toulouse, URM5505 CNRS, Universite de Toulouse, Toulouse;*

e-mail*: josiane.Mothe@irit.fr*

Major Fields of Scientific Research*: Information Systems, Information Retrieval, Data analytics, Big Data*

Mariam Haroutunian *- Professor, Institute for Informatics and Automation Problems, 0014, Yerevan, Armenia;*

e-mail: *armar@ipia.sci.am*

Major Fields of Scientific Researc*h: Information Theory, Information Security, Probability Theory and Statistics*