Perspective on Measurement Metrics for Community Detection Algorithms

Yang Yang, Yizhou Sun, Saurav Pandit, Nitesh V. Chawla, and Jiawei Han

Abstract Community detection or cluster detection in networks is often at the core of mining network data. Whereas the problem is well-studied, given the scale and complexity of modern day social networks, detecting "reasonable" communities is often a hard problem. Since the first use of k-means algorithm in 1960s, many community detection algorithms have been presented—most of which are developed with specific goals in mind and the idea of detecting meaningful communities varies widely from one algorithm to another.

As the number of clustering algorithms grows, so does the number of metrics on how to measure them. Algorithms are often reduced to optimizing the value of an objective function such as modularity and internal density. Some of these metrics rely on ground-truth, some do not. In this chapter we study these algorithms and aim to find whether these optimization based measurements are consistent with the real performance of community detection algorithm. Seven representative algorithms are compared under various performance metrics, and on various real world social networks.

The difficulties of measuring community detection algorithms are mostly due to the unavailability of ground-truth information, and then objective functions, such

Y. Yang (✉) · S. Pandit · N.V. Chawla
Department of Computer Science & Engineering, University of Notre Dame, Notre Dame, IN 46556, USA
e-mail: yyang1@nd.edu

S. Pandit
e-mail: spandit@nd.edu

N.V. Chawla
e-mail: nchawla@nd.edu

Y. Sun · J. Han
Department of Computer Science, University of Illinois, Urbana and Champaign, Urbana, IL 61801, USA

Y. Sun
e-mail: sun22@uiuc.edu

J. Han
e-mail: hanj@uiuc.edu

T. Özyer et al. (eds.), *Mining Social Networks and Security Informatics*,
Lecture Notes in Social Networks, DOI 10.1007/978-94-007-6359-3_12,

as modularity, are used as substitutes. The benchmark networks that simulate real world networks with planted community structure are introduced to tackle the unavailability of ground-truth information, however whether the simulation is precise and useful has not been verified. In this chapter we present the performance of community detection algorithms on real world networks and their corresponding benchmark networks, which are designed to demonstrate the differences between real world networks and benchmark networks.

Keywords Social network · Community detection · Objective functions · Benchmark network · Measurements

1 Introduction

Community detection algorithms attract a great deal of attention from researchers in computer science [17, 18], especially in the area of data mining, and becoming more and more important due to the rapid proliferation of social networks, such as Facebook, the "blogosphere" or even cellular phone communication networks. However, how to effectively measure the performance of community detection algorithms remains a problem without consensus. Currently many objective functions are used to evaluate the quality of detected communities, but whether these objective functions are good approximations of performance are not yet clear. We propose to compare community detection algorithms under various performance metrics, and on several social networks to explore whether current objective functions are consistent with the "ground-truth" of social network datasets. Another important purpose of our survey is to take a closer look at whether a consensus can at all be reached or whether different community detection algorithms are effective on different networks.

In order to conduct appropriate experiments, we divide the community detection algorithms into two categories based on whether the social network is heterogeneous or homogeneous. Additionally considering the heuristics or philosophy employed by community detection algorithms, some of the heuristics could be formalized into objective functions, e.g. modularity [16, 22] and partition density [1]. Then the clustering problem virtually reduces to by maximizing or minimizing these objective functions. However in some other algorithms, heuristics are hard to be abstracted by objective functions, such as RankClus [21].

As for performance metrics, they can also be classified into two categories according to whether their evaluations rely on ground-truth or not. We are using the metrics listed in Table 1, which are used frequently as performance metrics.

Besides those performance metrics we discussed above, Andrea Lancichinetti et al. [10] proposed to use benchmark networks with built-in communities to evaluate the performance of community detection algorithms, this method is also involved in our comparisons. Our objective of experiments on benchmark networks is to an-

Table 1 Performance metrics of community detection algorithms

Metrics	Based on ground-truth	Not based on ground-truth			
	Rand index	Internal density	Conductance	Cut ratio	Modularity
Equation	$\frac{SS+DD}{SS+SD+DS+DD}$	$\frac{2m_k}{n_k(n_k-1)}$	$1-\frac{l_k}{(2m_k+l_k)}$	$1-\frac{l_k}{n_k(n_k-1)}$	$\sum_{k=1}^{K} \text{mod}_k/2M$
Comments	The first character of each variable states whether two nodes are from the same ground-truth class, and similarly the second character of each variable represents whether they are classified together by the algorithm.	This is the internal density of links within the community C_k [11].	This is the fraction of total edge number pointing outside the community [11].	This is the fraction of all possible edges leaving the community structure [11].	This states the quality of communities [11].

For all these metrics, high score indicates better quality

swer the question that whether these simulated benchmark networks are reliable to measure the performance of community detection algorithms.

The remainder of the chapter is organized as follows. We introduce the preliminary information and related work in Sect. 2. Section 3 discusses our observations on experimental results collected from small networks. The discussion of large-scale networks results are presented in Sect. 4. We provide analysis of benchmark networks in topological perspective, and present associated experimental results in Sect. 5. The conclusion of our study is drawn in Sect. 6.

2 Related Work

Here we survey related work and discuss preliminary information for our work.

2.1 Community Detection Algorithms

A great deal of work has been devoted to finding communities in networks, and much of this has been designed to formalizing heuristic that a community is a set of nodes that has more intra connections than inter connections. The algorithms used in our survey are selected according to categories described in Sect. 1. We try to select a set of community detection algorithms, which are comprehensive and representa-

Table 2 Community detection algorithms

Algorithm	RankClus [21]			LinkCommunity [1]			LineGraph [6]		
	Formalization	Heter	Homo	Formalization	Heter	Homo	Formalization	Heter	Homo
Properties	No	Yes	N/A	Yes	Yes	Yes	Depends	Yes	Yes

Algorithm	Walktrap [16]			SPICi [9]			Betweenness [8]		
	Formalization	Heter	Homo	Formalization	Heter	Homo	Formalization	Heter	Homo
Properties	Yes	N/A	Yes	Yes	N/A	Yes	No	Yes	Yes

Algorithm	K-means [4]		
	Formalization	Heter	Homo
Properties	Yes	No	Yes

tive. In our work there are algorithms which can work in heterogeneous networks (RankClus [21]) and algorithms which are applicable in homogeneous networks (Betweenness [8]); we also include algorithms that employ objective functions to guide their clustering (Walktrap [16]) and algorithms that do not use objective functions (RankClus [21]). Some of algorithms are agglomerative (Walktrap), and some of them are divisive (Betweenness); some of algorithms can give overlapping community partition (LinkCommunity [1] and LineGraph [6]), and some of them can only give non-overlapping results (SPICi [9]).

2.2 Social Network Datasets

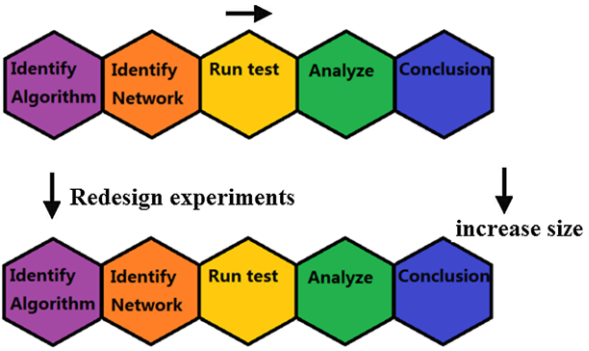
The social networks datasets are listed in Table 3. Due to the obstacle of collecting community ground-truth information, in our work we can only provide 9 datasets listed in Table 3. Our efforts are made to ensure that these datasets are representative. In order to observe the behaviors of algorithms in different sized networks, we make the networks sizes range from 34 nodes to 80,513 nodes. Besides homogeneous networks, we also include heterogeneous networks, such as Cities and Services [24].

The network datasets (or their sources) used for experimentation are: Zachary Karate Club [25], Mexican Political Power [7], Sawmill [12], Cities and Services [24], MIT Reality Mining [5], Flickr [23], Youtube [13], and LiveJournal [13]. These real world networks are selected as our datasets because they have well defined communities ground-truth information. The first five of them are small social networks and are valuable at the startup stage of our survey, by using which we can have a more intuitive and clear view of social networks and community detection algorithms. The last three are large-scale social networks, they are used to verify

Table 3 Social network datasets

Datasets	Size	Ground-truth communities	Is heterogeneous
Karate Club	34 nodes	2	No
Mexican	35 nodes	2	No
Sawmill	36 nodes	3	No
Reality Mining	79 nodes	2	No
Cities&Services	101 nodes	4	Yes
Benchmark	45 nodes	3	No
Flickr	80,513 nodes	195	No
LiveJournal	3,986 nodes	113	No
Youtube	8,202 nodes	168	No

Fig. 1 Methodology



whether the conclusions made in small social networks still hold for large-scale social networks.

2.3 Methodology

The methodology (Fig. 1) employed in our paper is something like ”black box” approach by measuring algorithms under different objective functions, because whether these objective functions are reliable or not is unknown to us, by employing this methodology we can simplify our work of experiments and achieve more concise comparisons of these performance metrics for community detection algorithms. In the first step only small size social networks with ground-truth information are chosen, which is easy for us to conclude the initial observations. With these findings we can increase the social networks size and see whether these conclusions still hold with the increment of social network size. The last three networks (Table 3) are selected to verify our conclusions when community detection algorithms work on large-scale datasets. Using this methodology we can explore more precise conclusions and unveil the relationship between network sizes and performance metrics.

3 Community Detection on Small Networks

Among our selected datasets there are heterogeneous networks and homogeneous networks, while in the set of chosen clustering algorithms there are algorithms designed for heterogeneous networks, homogeneous networks or both (Table 2). Such that if an algorithm is designed for homogeneous networks and we apply it on heterogeneous networks, it may have unreasonable results. However there is possibility that it still has high scores in some objective functions, in this way bias between objective function and ground-truth information could be identified.

3.1 Experiments on Small Networks

We firstly apply six selected community detection algorithms on the first six datasets listed in Table 3 which have small sizes. The communities detected are evaluated by performance metrics presented in Table 1.

We study these data according to two dimensions: algorithm dimension and objective function dimension. Algorithm dimension means we only study related behaviors of one specified algorithm, while objective function dimension refers that we analyze information related to specific objective function. Generally speaking the ground-truth based *rand index* is much more precise to differentiate qualities of algorithms on networks, we will compare the *rand index* scores with other metrics to unfold the bias between them.

If we focus on the RankClus algorithm we can see that metrics, such as *internal density*, *conductance*, *cut ratio* and *modularity*, to some extent can reveal the algorithm's performance over different datasets. For example RankClus has the worst performance on Mexican Political dataset when comparing with its performance on other networks in terms of *rand index*; similarly *internal density*, *modularity* and *conductance* also suggest that the detected communities are of poor quality. However there is also bias, for instance RankClus has the best performance on cities and services dataset when comparing with other algorithms in terms of *rand index*; however its related *conductance*, *cut ratio* and *modularity* are very low. Another example is, RankClus correctly clustered all nodes in the Karate Club dataset, however the *internal density* has a lower score when comparing with other algorithms.

Another interesting observation is that Walktrap algorithm and LinkCommunity algorithm have the worst performance on the same social networks (they cluster nodes of Mexican dataset and cities dataset into one single community). And more interesting thing is that while they have the worst performance (fail to partition the network) their *conductance* and *cut ratio* scores are perfect, which gives diametrically opposed evaluations. The reason for this is that Walktrap and LinkCommunity are both designed to optimizing *modularity* objective functions, Mexican dataset and cities dataset happen to have larger *modularity* than any partitions of themselves. In contrast optimization of criteria does not always lead to qualified communities. Additionally we can see that although RankClus is designed for heterogeneous networks, it also has surprisingly high scores on specific homogeneous networks. This

Table 4 Small networks experiment results

Dataset	GT	RankClus						Walktrap					
		RI	ID	C	CR	M	Co	RI	ID	C	CR	M	Co
Karate	2	**1.000**	0.275	**0.875**	**0.965**	0.211	2	0.745	0.308	0.810	0.953	0.188	2
Mexican	2	0.489	0.168	0.434	0.778	0.003	2	0.536	0.197	**1.000**	**1.000**	0.018	1
Sawmill	3	0.530	0.048	0.138	0.877	0.003	3	0.560	0.307	**0.845**	**0.981**	0.178	2
Cities	4	**0.668**	**0.988**	0.168	0.018	0.004	4	0.348	0.266	**1.000**	**1.000**	0.006	1
Reality	2	**0.575**	**1.000**	**0.987**	**0.988**	0.099	2	0.561	**1.000**	0.968	0.969	0.100	2
Bench	3	0.718	0.208	0.625	0.937	0.107	3	**1.0**	0.310	**0.874**	**0.981**	0.289	3

Dataset	GT	K-means						LinkCommunity					
		RI	ID	C	CR	M	Co	RI	ID	C	CR	M	Co
Karate	2	0.941	0.168	0.503	0.897	0.057	2	0.743	0.499	0.468	0.907	**0.284**	8
Mexican	2	0.536	0.218	0.606	0.847	0.066	2	0.536	0.197	**1.000**	**1.000**	0.018	1
Sawmill	3	0.527	0.309	0.761	0.961	0.232	3	0.560	0.328	0.731	0.902	**0.314**	5
Cities	4	0.604	0.310	0.282	0.807	**0.032**	4	0.348	0.266	**1.000**	**1.000**	0.006	1
Reality	2	0.523	0.433	0.742	0.944	**0.189**	2	0.574	0.964	0.898	0.828	0.109	3
Bench	3	**1.000**	0.143	0.411	0.888	0.015	3	0.826	0.397	0.598	0.931	**0.406**	11

Dataset	GT	SPICi						Betweenness					
		RI	ID	C	CR	M	Co	RI	ID	C	CR	M	Co
Karate	2	0.586	**0.729**	0.524	0.898	0.136	5	0.913	0.210	0.630	0.933	0.199	3
Mexican	2	0.553	**0.600**	0.648	0.903	**0.155**	3	**0.605**	0.079	0.100	0.790	0.036	7
Sawmill	3	**0.629**	**0.633**	0.547	0.947	0.192	7	0.570	0.028	0.110	0.908	0.022	6
Cities	4	0.636	0.513	0.110	0.799	0.022	12	0.267	0.000	0.000	0.729	0.007	12
Reality	2	0.573	0.88	0.844	0.900	0.098	2	0.563	0.000	0.110	0.322	0.079	9
Bench	3	0.865	**0.521**	0.731	0.965	0.260	5	0.943	0.399	0.721	0.964	0.284	4

In these tables GT states the number of classes of ground-truth, RI is the rand index score, ID is the internal density, C is the conductance, CR is the cut ratio, M represents the modularity, and Co is the number of communities detected by corresponding algorithms. As for these metrics, *higher score indicates higher quality*

is an interesting phenomenon we need to look deep into for our future work. From Table 1 we can observe that the behaviors of community detection algorithms vary in different networks.

When we concentrate on a single objective function, for instance, *internal density*, trivially we can find that SPICi algorithm has the best *internal density* on Karate dataset; however RankClus has the best performance (in terms of *rand index*) on Karate dataset. Another example is, LinkCommunity has the best *modularity* on Sawmill dataset while SPICi has the best performance (in terms of *rand index*) on Sawmill dataset. Among the data in Table 4 there are a lot of such examples, based

on current experiments results and observations, we can see that the correlation between the *rand index* and objective functions that are not based on ground-truth, is not strong.

Here we conclude our findings as below:

1. Heuristics are native reasons for behavioral differences or similarities of algorithms, similar heuristics lead to similar performance. A good example is Walktrap and LinkCommunity, although one of them generates overlapping communities while another does not, they have very similar behaviors in our selected datasets.
2. Different heuristics fit in different circumstances, inappropriate heuristics lead to damages on performance. RankClus' heuristic is applicable for most of social networks (Ranking and Clustering mutually enhance each other), however it has worst performance on the benchmark network in terms of *rand index* because the benchmark network is significantly different from social networks in most topological properties.
3. Community structure of networks depends on many factors, topological properties of networks are only parts of them. In some circumstances when topologies do not prevail, use of objective functions (highly related to topological properties) may lead to inappropriate evaluations of communities. This is the reason that *rand index* does not always agree with other metrics in our work.

3.2 Correlations Between Objective Function and Ground-Truth Measurement

Actually we can take a closer look at the bias between ground-truth measurement and objective functions by presenting their correlations quantitatively. Assume there are a set of datasets $D = \{D_1, D_2, D_3, \ldots, D_M\}$, a set of algorithms $A = \{A_1, A_2, A_3, \ldots, A_N\}$ and a set of objective functions $F = \{F_1, F_2, F_3, \ldots, F_K\}$, we apply algorithms on the given datasets and calculate corresponding objective functions scores; in this way for each pair of dataset D_i and objective function F_k there is a vector $v_{D_i,F_k} = (F_k(A_1, D_i), F_k(A_2, D_i), F_k(A_3, D_i), \ldots, F_k(A_M, D_i))$, and for each dataset there is also a vector for ground-truth $v_{D_i,G} = (G(A_1, D_i), G(A_2, D_i), G(A_3, D_i), \ldots, G(A_M, D_i))$.

By computing the correlation coefficient between v_{D_i,F_k} and $v_{D_i,G}$, we can quantitatively identify whether an objective function is a good measurement of community detection algorithm. In Fig. 2 we can observe most of objective functions on most of datasets have little correlation with ground-truth scores and some of them even have negative correlationship, such as *internal density* on karate dataset and *modularity* on reality dataset. And additionally we can see *internal density* is much more correlated with ground-truth measurement than other objective functions. From this plot we can conclude that these objective functions are not reliable enough to determine whether a community detection algorithm performs well or

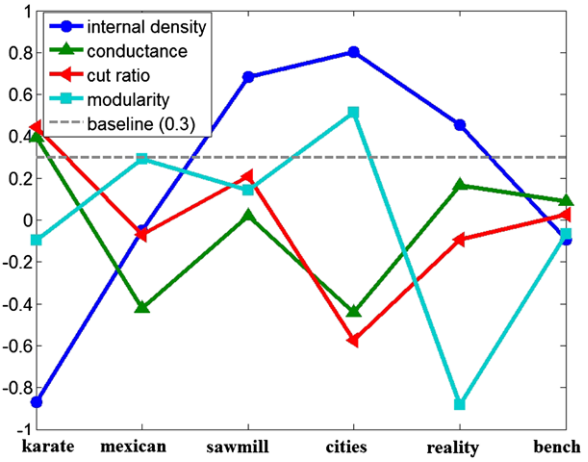


Fig. 2 Correlation coefficients between objective functions and ground-truth, small networks

not on a given network. An interesting observation is that *cut ratio* and *conductance* share the same behavior in Fig. 2. This is intuitive because both metrics are designed to capture the inter-cluster interactions.

Andrea Lancichinetti et al. proposed to use generated benchmark networks to measure the performance of community detection algorithms. However in Table 4 we can see that these six algorithms listed above all have very high *rand index* scores; however the performance on other datasets is not so promising. There are two possible reasons, the first one is that these generated benchmark networks are easy to be "mined", another one is that the generated benchmark networks are not good simulations of "real world" networks. In order to know which reason contributes to this phenomenon, we conducted more experiments and present the results in Sect. 4.1. We would like to note that, the quality of communities detected by algorithms is hard to evaluate, for example, in Fig. 3 even with ground-truth information it is still difficult for us to tell which method performs better on Karate dataset (LinkCommunity and Line Graph give almost the same *rand index* scores). Performance metrics can help our evaluations but cannot completely define the quality.

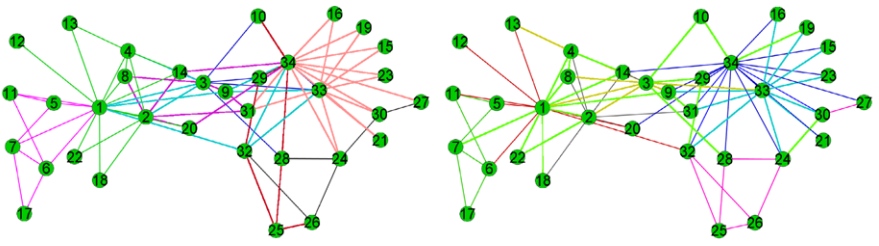


Fig. 3 LinkCommunity (*left*) and line graph (*right*) clustering on the Karate Club dataset

Table 5 Large-scale networks experiment results

Dataset	GT	RankClus						Walktrap					
		RI	ID	C	CR	M	Co	RI	ID	C	CR	M	Co
Flickr	195	0.950	**0.967**	0.108	0.998	0.079	195	0.680	0.298	**0.286**	**0.999**	**0.287**	344
Youtube	168	0.979	**0.984**	0.131	0.998	0.141	168	0.801	0.416	**0.710**	**0.999**	0.354	152
LiveJournal	113	0.977	**0.953**	0.434	0.990	0.275	114	0.981	0.487	0.780	0.990	0.365	216

Dataset	GT	K-means						LinkCommunity					
		RI	ID	C	CR	M	Co	RI	ID	C	CR	M	Co
Flickr	195	0.950	0.910	0.213	0.998	0.100	195	*	*	*	*	*	*
Youtube	168	0.979	0.931	0.396	0.990	0.128	168	0.983	0.415	0.152	0.996	**0.710**	6,701
LiveJournal	113	0.983	0.908	**0.802**	0.990	0.366	114	**0.988**	0.364	0.563	**0.999**	**0.780**	1,430

Dataset	GT	SPICi						Betweenness					
		RI	ID	C	CR	M	Co	RI	ID	C	CR	M	Co
Flickr	195	**0.960**	0.437	0.045	0.998	0.065	10,267	*	*	*	*	*	*
Youtube	168	**0.984**	0.297	0.380	0.900	0.122	816	*	*	*	*	*	*
LiveJournal	113	**0.988**	0.212	0.678	**0.999**	0.250	746	*	*	*	*	*	*

In these tables GT states the number of classes of ground-truth, RI is the rand index score, ID is the internal density, C is the conductance, CR is the cut ratio, M represents the modularity, and Co is the number of communities detected by corresponding algorithms. As for these metrics, *higher score indicates higher quality*. Results for Betweenness algorithm and partial results of LinkCommunity are not available due to their expensive computational cost and memory requirement (marked with *)

4 Community Detection on Large-Scale Networks

4.1 Experiments on Large-Scale Networks

In the above section we apply six representative community detection algorithms on six small "real world" datasets and unfold several interesting phenomenons of objective functions. In this section we increase sizes of networks and perform the same algorithms on these large-scale networks to verify whether the experimental results will be different from those on small networks.

The experimental results will also be analyzed in two dimensions. Due to computation complexity and memory requirement, results for Betweenness algorithm and part of results for LinkCommunity are not available. In Table 5 we still can see bias between ground-truth information and objective functions; for example *conductance*, *cut ratio* and *modularity* all suggest that Walktrap algorithm works better than other algorithms on Flickr dataset, however the truth is Walktrap algorithm has the lowest *rand index* score. In our observation the reliability of objective functions

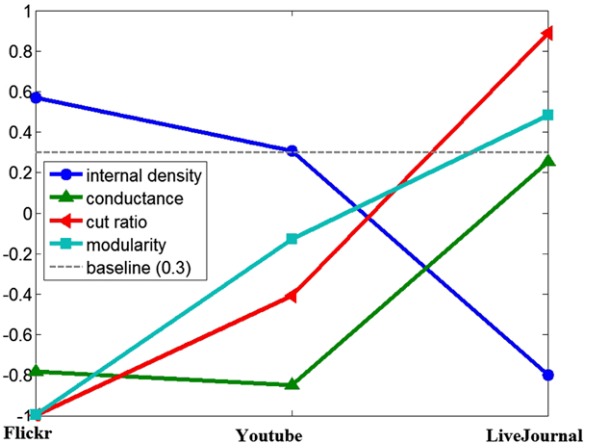


Fig. 4 Correlation coefficients between objective functions and ground-truth, large-scale networks

does not improve when the network size is increased, the bias between *rand index* and other metrics is still apparent as discussed in small size networks results. However some of them have consistent measurements with ground-truth information of some datasets, for example, *cut ratio* and *rand index* both suggest SPICi and LinkCommunity work best in LiveJournal dataset. In the same way we compute the correlation coefficients between ground-truth measurement and objective functions and plot them to verify the reliability of objective functions on large-scale networks.

In Fig. 4 we can see that our conclusions on small size networks still hold for large-scale networks, most of objective functions on most datasets have none significant correlation or even have anti-correlation. Interestingly *internal density* still performs better than other objective functions on large-scale networks.

5 Benchmark Networks

The benchmark network generator [10] can take the parameters such as, node number, average degree, and mixing parameter (Table 6) to simulate an existing "real world" network, and ground-truth information for communities is also generated. Its objective is to simulate an existing "real world" network and provide an estimated information of communities structure for this network, in this way community detection algorithms' performance can be trivially evaluated using the generated ground-truth information.

5.1 Network Model Discussion

In the work of benchmark network [10] nodes of network are partitioned into l (given by input) groups, the sizes of groups can follow some distribution specified

Table 6 Parameters for benchmark network generator [10]

Parameter	Description
n	nodes number
k	average degree
maxk	max degree
μ_t	mixing parameter
minc	minimum community size
maxc	maximum community size

in the input. Nodes of the same group are linked with a probability p_{in}, whereas nodes from different groups are connected with a probability p_{out}. Each subgraph corresponding to a group is then a random Erdős Rényi graph with connection probability $p = p_{in}$. If $p_{in} > p_{out}$ the intra-cluster density exceeds the inter-cluster density, and then the community structure of the simulated network is formed. However the Erdős Rényi graph is different from real world networks in many aspects, which are presented in Table 7.

From Table 7 we can see the Erdős Rényi network differs from real world networks in two important properties: degree distribution and clustering coefficient. The Erdős Rényi network is not a good simulation of the real world network, thus the benchmark network (based on Erdős Rényi model) discussed in [10] is unlikely to precisely simulate the real world community structures. From the network model perspective the performance of algorithms on benchmark networks cannot lead to an accurate estimation of the performance of algorithms on real world networks. We conduct several experiments in Sect. 5.2 to demonstrate the correctness of our opinion.

5.2 Experimental Results

Our experiment is to verify whether these generated networks can simulate the "real world" network precisely. For the first four datasets listed in Table 3 there are ground-truth information for communities, thus we can compare the performance of algorithm on these networks and their corresponding simulated networks to identify whether benchmark networks generator is feasible to evaluate community detection algorithms' performance. Most of the parameters requires by the benchmark network generator can be calculated trivially, such as degree, minimum community size and maximum community size; however the mixing parameter μ_t, which is set to define the proportion of each nodes links which link outside its community, is hard to computed when the community structure is not available. Leto Peel [15] proposed a novel method to estimate the mixing parameter μ_t by network structure information. In our experiments we employ Leto Peel's method to calculate the mixing parameter and use the value as the input for benchmark network generator.

Table 7 Comparison between Erdős Rényi networks and real world networks

Properties	Degree distribution	Clustering coefficient	Average diameter
Real World Networks	Power Law	High	Small
Erdős Rényi Networks	Poison	Low	Small

Table 8 Benchmark networks

Dataset	GT	RankClus		Walktrap		K-means		LinkCom		SPICi		Betweenness	
		RI	Co	RI	Co	RI	Co	RI	Co	RI	Co	RI	Co
Karate	2	1.000	2	0.745	2	0.941	2	0.743	8	0.586	5	0.913	3
Mexican	2	0.489	2	0.536	1	0.536	2	0.536	1	0.553	3	0.605	7
Sawmill	3	0.530	3	0.560	2	0.527	3	0.560	5	0.629	7	0.570	6
Reality	2	0.575	2	0.561	2	0.523	2	0.574	3	0.573	2	0.563	9

Dataset	GT	RankClus		Walktrap		K-means		LinkCom		SPICi		Betweenness	
		RI	Co	RI	Co	RI	Co	RI	Co	RI	Co	RI	Co
sim-Karate	2	0.510	2	0.520	4	0.510	2	0.520	28	0.500	6	0.510	3
sim-Mexican	2	0.510	2	0.510	8	0.510	2	0.500	1	0.510	3	0.500	6
sim-Sawmill	3	0.630	3	0.610	7	0.610	3	0.630	1	0.630	9	0.630	2
sim-Reality	2	0.490	2	0.500	2	0.490	2	0.500	1	0.500	2	0.490	2

In these tables GT states the number of classes of ground-truth, RI is the rand index score, and Co is the number of communities detected by corresponding algorithms. As for these metrics, *higher score indicates higher quality*

By the information listed in Table 8 we simulate 100 networks for first four datasets listed in Table 3, cities and services dataset can not be simulated because benchmark network generator is not able to simulate heterogeneous network. We apply selected algorithms on these 100 networks for each dataset, compute the rand index scores and then calculate the average rand index score for each algorithm on 100 simulated networks. The results are listed in Table 8. The top phase of the table shows the rand index scores of each algorithm on each dataset, the bottom phase presents the performance of each algorithm on each dataset's simulated network. Trivially we can see the "real world" dataset can easily differentiate algorithms based on their performance, for example, RankClus outperforms others on karate dataset and SPICi performs much better than others on sawmill dataset, while for the simulated networks algorithms almost have the same scores on the same dataset. In conclusion, 1) it is hard to differentiate the performance of community detection algorithms on benchmark networks; 2) the behaviors of community detection algorithms on real world networks are different from their behaviors on corresponding benchmark networks; 3) benchmark networks are not promising substitutes of real world datasets for algorithms measurements.

6 Conclusion

Seven representative algorithms were compared under various performance metrics, and on various "real world" social networks, from small size networks to large-scale networks. Based on our current observations of experiments results, we can conclude that performance metrics based on the ground-truth information are more reliable than objective functions that are not based on ground-truth, such as *internal density* and *modularity*. And the reliability of non ground-truth based objective functions does not improve with the increment of network size. Characteristics of different algorithms are unfold in our experiments, RankClus has best or comparable performance on most datasets due to the reason that it employs a more general heuristic instead of using objective functions to guide clustering process.

In our work we also discuss the benchmark networks with built-in communities structures. We analyzed the differences between benchmark networks and real world networks, such as degree distribution and clustering coefficient. These differences lead to the invulnerability of benchmark networks as they are used to measure the performance of community detection algorithms designed for real world networks. By experiments we conclude that the networks created by the benchmark network generator [10] are not qualified enough to differentiate the performance of community detection algorithms; algorithms tend to have similar scores in given simulated networks.

7 Future Work

Our current work has included the experiments on small networks, large-scale networks and benchmark networks and draw several conclusions. For example objective functions not based on ground-truth information are not reliable to accurately reveal the performance of algorithms on social networks we have studied. In the future work more performance metrics are to be involved, and more algorithms and datasets will be selected to reinforce the robustness of our conclusions. With the gradual increment of dataset size the relation between social network volume and objective functions is estimated to be unfolded. Next, the networks studied will be expanded into other genres than social networks, such as biological networks and telecommunication networks, and algorithms and performance metrics will be compared independently in each category of networks. Much more objective functions will be included into our future study, which is designed to conduct an empirical comparison of algorithms for network community detection. In this way we can expand our work into other areas than social networks, the different behaviors of objective functions in different types of networks may be unfold in such experiments.

Acknowledgements This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or

References

1. Ahn Y, Bagrow JP, Lehmann S (2010) Link communities reveal multiscale complexity in networks. arXiv:0903.3178v3 [physics.soc-ph]
2. Chen J, Zaïane OR, Goebel R (2009) Detecting communities in social networks using max-min modularity. In: International conference on data mining (SDM 09)
3. de Nooy W, Mrvar A, Batagelj V (2004) Exploratory social network analysis with Pajek, Chapter 12. Cambridge University Press, Cambridge
4. Dhillon I, Guan Y, Kulis B (2005) A fast kernel-based multilevel algorithm for graph clustering. In: Proceedings of the 11th ACM SIGKDD, Chicago, IL, August 21–24
5. Eagle N, Pentland A (2006) Reality mining: sensing complex social systems. Pers Ubiquitous Comput 10(4):255–268
6. Evans TS, Lambiotte R (2009) Line graphs, link partitions, and overlapping communities. Phys Rev E 80(1):016105
7. Gil-Mendieta J, Schmidt S (1996) The political network in Mexico. Soc Netw 18(4): 355–381
8. Girvan M, Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci USA 99(12):7821–7826
9. Jiang P, Singh M (2010) SPICi: a fast clustering algorithm for large biological networks. Bioinformatics 26(8):1105–1111
10. Lancichinetti A, Fortunato S, Kertész J (2009) Detecting the overlapping and hierarchical community structure in complex networks. New J Phys 11(3):033015
11. Leskovec J, Lang KJ, Mahoney MW (2010) Empirical comparison of algorithms for network community detection. In: WWW 2010, April 26–30, Raleigh, North Carolina, USA
12. Michael JH, Massey JG (1997) Modeling the communication network in a sawmill. For Prod J 47:25–30
13. Mislove A (2009) Online social networks: measurement, analysis, and applications to distributed information systems. Ph.D Thesis, Rice University, Department of Computer Science
14. Pandit S, Kawadia V, Yang Y, Chawla NV, Sreenivasan S (2011) Detecting communities in time-evolving proximity networks. In: IEEE first international workshop on network science (submitted)
15. Peel L (2010) Estimating network parameters for selecting community detection algorithms. In: 13th international conference on information fusion
16. Pons P, Latapy M (2006) Computing communities in large networks using random walks. J Graph Algorithms Appl 10(2):191–218
17. Radicchi F, Castellano C, Cecconi F, Loreto V, Parisi D (2004) Defining and identifying communities in networks. Proc Natl Acad Sci USA 101(9):2658–2663
18. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905
19. Steinhaeuser K, Chawla NV (2010) Identifying and evaluating community structure in complex networks. Pattern Recognit Lett 31(5):413–421
20. Steinhaeuser K, Chawla NV Is modularity the answer to evaluating community structure in networks? In: International conference on network science (NetSci), Norwich, UK
21. Sun Y, Han J, Zhao P, Yin Z, Cheng H, Wu T RankClus: integrating clustering with ranking for heterogeneous information network analysis. In: EDBT 2009, March 24–26, 2009, Saint Petersburg, Russia
22. Sun Y, Han J (2010) Integrating clustering and ranking for heterogeneous information network analysis. In: Yu PS, Han J, Faloutsos C (eds) Link mining: models, algorithms and applications. Springer, New York, pp 439–474

23. Tang L, Liu H (2009) Scalable learning of collective behavior based on sparse social dimensions. In: Proceedings of the 18th ACM conference on information and knowledge management (CIKM'09)
24. World Cities and Global Firms dataset was created by Taylor PJ, Walker DRF as part of their project "World city network: data matrix construction and analysis" and is based on primary data collected by Beaverstock JV, Smith RG, Taylor PJ (ESRC project "The geographical scope of London as a world city" (R000222050))
25. Zachary WW (1977) An information flow model for conflict and fission in small groups. J Anthropol Res 33:452–473