CS-ComDet: A Compressive Sensing Approach for Inter-Community Detection in Social Networks

Hamidreza Mahyar*, Hamid R. Rabiee†, Ali Movaghar†, Elaheh Ghalebi*, Ali Nazemian*
Department of Computer Engineering, Sharif University of Technology (SUT)
*e-mail: {hmahyar, ghalebi, nazemian}@ce.sharif.edu , †e-mail: {rabiee, movaghar}@sharif.edu

***Abstract*—One of the most relevant characteristics of social networks is community structure, in which network nodes are joined together in densely connected groups between which there are only sparser links. Uncovering these sparse links (*i.e.* inter-community links) has a significant role in community detection problem which has been of great importance in sociology, biology, and computer science. In this paper, we propose a novel approach, called CS-ComDet, to efficiently detect the inter-community links based on a newly emerged paradigm in sparse signal recovery, called compressive sensing. We test our method on real-world networks of various kinds whose community structures are already known, and illustrate that the proposed method detects the inter-community links accurately even with low number of measurements (*i.e.* when the number of measurements is less than half of the number of existing links in the network).**

***Index Terms*—Compressive Sensing, Inter-Community Detection, Social Networks.**

I. Introduction

Many real-world systems can be modeled as networks of nodes interactions. A few examples of these networks are Internet, World Wide Web, social interactions, information systems, and biological systems. Actually, we live in a world of networks. A network is often represented by a graph with a set of nodes joined in pairs by links. For instance, there is the fact that social users as nodes are, by definition, connected to others via some relations as links (*i.e.*, friendship, participation in the same event, and membership in the same group), referred as *social graph*. In recent years, a wide rang of research has been done on characteristics of social networks in various domains, from measurement of structural properties to extraction of functional properties [1]. It has been revealed that one of the most common properties in many real-world social networks is *community structure* [2]. A network is said to have community structure if there exist densely connected groups of nodes, with only sparser connections between groups [3]. These sparse connections are usually called "*inter-community links*". Communities, also called *clusters* or *modules* or *groups*, correspond to real social groups, similarity, or a common function which is significant structure in the networks. A figurative sketch of a network with its communities is shown in Fig. 1. The communities are the groups of more intensely interconnected nodes, while there are just small number of connections (*i.e.* sparse links) between these communities.

Within the past decade, social networks (especially online social networks) have emerged as the most popular complex networks. For example, according to Nielsen [4], worldwide users spend over 110 billion minutes on social media sites per month, which accounts for 22% of all the time spent online, surpassing even the time spent on email. Despite their attractions, extraction of useful knowledge from the network leads to collection and analysis of network data. However, there are two main constraints which make it difficult or impossible to obtain direct measurement of each individual node/link in the network: (1) Today, with the growth of technology, we are faced with very large scale networks. For instance, `Facebook` as the most popular online social networks, has attracted more than 1.4 billion monthly active users worldwide as of March 2015 (Source: Facebook Inc.); (2) The global structure for many networks is initially unknown. For instance, there are access limitations in most social services such as login requirements, topological constraints, API query limits, and treatment of user data as proprietary. In the analysis of social networks, the existence of missing data is almost inevitable because the aforementioned constraints may prevent access to entire data of the networks. Mostly, direct measurement of each individual node can be difficult, costly, and sometimes impossible due to massive scale, distributed management, and access limitation of real social networks. Therefore, an efficient method for indirect measurement and estimation of network internal characteristics seems to be more essential.

In this paper, we want to efficiently detect inter-community links (the links that connect nodes of different communities) in social networks with high community structure in an indirect manner. In social interacting networks, these links are of great importance and identifying them is an essential issue in network monitoring. *For example*, the links that have more participation in cascading the diffusion event of an arbitrary type of information throughout the social graph; the links that represent the amount of friendship relation between communities; and the links that show the activity rate between membership groups. According to the definition of community structure in social networks, the inter-community links are often sparse in the network structure such that the number of these links are often much smaller than the set of all links in the network. In this paper, we introduce a novel approach to efficiently identify the inter-community links of the social networks using *compressive sensing theory*. Compressive Sensing (also known as Compressive Sampling or Compressed Sensing) [5–10] is a recently emerged paradigm in signal processing and information theory which tries to

ASONAM '15, August 25-28, 2015, Paris, France
© 2015 ACM. ISBN 978-1-4503-3854-7/15/08 $15.00
DOI: http://dx.doi.org/10.1145/2808797.2808856

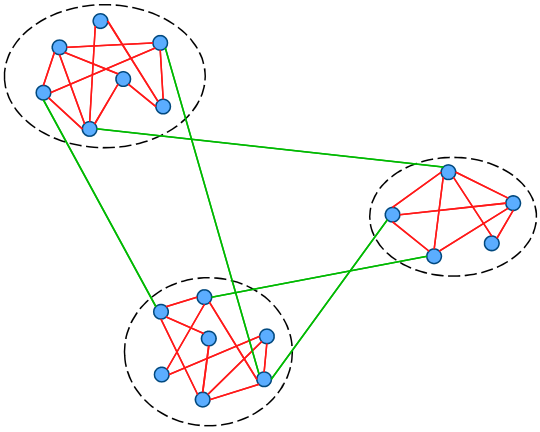


Fig. 1: A schematic representation of a network with community structure. In this network, there are three communities of densely connected groups of nodes (enclosed by the dashed circles), with only sparse connections between groups.

recover sparse signals from small number of non-adaptive measurements or incomplete observations. Its main goal is to sample and compress sparse signals, simultaneously. The fundamental idea behind compressive sensing (CS) is that in an appropriate lower dimensional representation (*e.g.* sparse vector, low-rank matrix, etc.), the under-sampled data of a signal have all the information needed about that signal [9].

The developments in compressive sensing started with the seminal works in [5] and [8]. The authors noted that the combination of ℓ_1-minimization and random matrices can lead to efficient recovery of sparse vectors and also has strong potential to be used in many applications. For the last couple of years, CS has been considered in signal processing, but its role in network applications is still in its early stages due to some challenging issues. One of the most restrictive challenges is the construction of *feasible* measurement matrix. Many existing results for designing measurement matrices depend critically on assumptions that do not hold for network applications. *For example*, in networks, a measurement matrix is in a more limiting class taking only non-negative integers, while random Gaussian measurement matrices are usually used in current CS literature. More significantly in networks, measurements are restricted by network topological constraints which is again absent in existing CS research. In other words, for every measurement, only links that induce a path or connected subgraph can be aggregated together in the same measurement. As a result, compressive sensing for network applications is quite different from other CS problems, although it is interesting in its own right because we can represent many real-world systems by their graphs/networks.

There have just been a few recent works that consider network topological constraints in order to design a feasible measurement matrix with non-negative integer entries over networks (graphs) using compressive sensing [11–17]. To the best of our knowledge, there is no previous work on inter-community detection in social networks using compressive sensing. Our motivation for using CS is that it can provide a concrete mathematical framework for sparse recovery problem in networks and a sparse signal can be recovered from a relatively small number of measurements or incomplete observations. In social networks, the existing sparsity in the network structure (*e.g.* the number of inter-community links are much smaller than the set of all links) helps to make this technique more applicable.

As a beneficial application to our approach, a simple way for community detection in a graph is to detect the inter-community links and remove them, so that the communities get disconnected from each other [3]. In addition to community detection, our method has potential applications in predicting or recommending social connections for a user as well as in understanding global diffusion of information. The rest of this paper is organized as follows. First, we introduce some preliminaries such as basic notations, problem statement, and problem formulation. Then, we propose our novel approach for inter-community detection in social networks using compressive sensing. Finally, we experimentally evaluate the performance of the proposed method via simulation results.

II. Preliminaries

A. Basic Notations and Definitions

We consider a social Network, expressed by an undirected static graph $\mathcal{G} = (V, E)$, where $V = \{v_1, v_2, ..., v_n\}$ denotes the set of nodes (vertices) with cardinality $|V| = n$, and $E = \{e_1, e_2, ..., e_N\}$ is the set of unweighted links (edges) with cardinality $|E| = N$. Let Adj be the adjacency matrix of $\mathcal{G}$, where $Adj(u, v) = 1$ if and only if there exist a link between u and v, otherwise $Adj(u, v) = 0$. For a node $v \in V$, we denote its degree by $deg(v)$ and the list of its neighbors by $Nbr(v) \subset V$. A graph $\mathcal{G}$ can be weighted. $W(u, v)$ denotes the weight of link $(u, v) \in E$, and the weight of node $u \in V$ is given by:

$$W(u) = s(u) = \sum_{v \in Nbr(u)} W(u, v), \tag{1}$$

and $W(\mathcal{G}) = [W(u, v)]_N$ is the vector of weights of links in graph $\mathcal{G}$. One of the most significant measure for actual weights in a weighted network is obtained by considering the node strength $s(u)$ which is defined in Eq. (1).

B. Problem Statement and Importance

Community is a group of densely connected nodes in which there are more links between nodes within the community (intra-community links) and only sparser links between communities (inter-community links) [18]. Nodes in the same community probably share a common properties and/or play similar roles throughout the network. Communities happen in many networked systems such as social, biological, information, and technological systems. For example, (1) Society often has a wide range of possible community organization like families, working, friendship, towns, nations. Social communities have been studied for a long time [19]. (2) In protein-protein interacting networks, a group of proteins is organized as a community to have the same specific function within the

cell [20]. (3) The group of pages in World Wide Web with the same or related topics corresponds to a community [21]. (4) The diffusion of Internet also leads to the creation of online virtual communities. Therefore, our proposed method may have potential applications in many networks besides social networks.

Communities can have concrete applications. Web clients can partitioned into the different communities based on their interests similarity and geographically near to each other. In order to improve the performance of provided services on the World Wide Web, each community of clients could be served by a dedicated mirror server [22]. In the network of purchase relationships between customers and online retailers products (*i.e.*, `www.amazon.com`), detecting the community of customers with similar interests helps to set up an efficient recommendation system [23]. In order to efficiently store the graph of big data and handle navigational queries, clustering large graphs can be used to create proper data structures [24].

Detecting communities and their boundaries in networks is an essential task to understand the structure, function, and evolution in various areas for complex networks specifically social networks. As stated in [3], inter-community links which connect boundary nodes of different communities, play an important role in mediation, relationship, and exchanges between communities. Moreover, a simple way to identify and separate communities in a graph is to detect inter-community links and remove them [25]. As a consequence, proposing an efficient algorithm for detection of inter-community links in a network is an essential task with a wide range of applications.

According to the definition of community structure in the social networks, inter-community links are sparse in the sense that number of these links are much smaller than those inside the communities. Our main goal in this paper is to propose an efficient sparse recovery method based on compressive sensing framework for accurately detecting the inter-community links in social networks. We, for the first time, consider the community structure of the networks for the proposed approach in the context of compressive sensing.

C. Model and Problem Formulation

Consider the graph $\mathcal{G} = (V, E)$. Suppose every link i has a real value x_i, and vector $\mathrm{x} = (\mathrm{x}_i, i = 1, 2, ..., |E|)$ is associated with $E(\mathcal{G})$. ℓ_p-norm of vector x defines as the following [8],

$$\|\mathrm{x}\|_p = (\sum_{i=1}^{n} |\mathrm{x}|^p)^{1/p}. \tag{2}$$

Note that for $p = 0$, $\|\mathrm{x}\|_0$ is the number of non-zero elements in x; for $p = 1$, $\|\mathrm{x}\|_1$ is the summation of the absolute values of elements in x; for $p = 2$, $\|\mathrm{x}\|_2$ is the usual Euclidean norm; and for $p = \infty$, $\|\mathrm{x}\|_\infty$ is the maximum of the absolute values in x. x is a k-sparse link vector if $\|\mathrm{x}\|_0 = k$, namely x has only k non-zero elements. In other words, the sparsity of the signal x is k. For example, inter-community links have sparsity property in the social networks, so that the number of these links are much smaller than all links in the network.

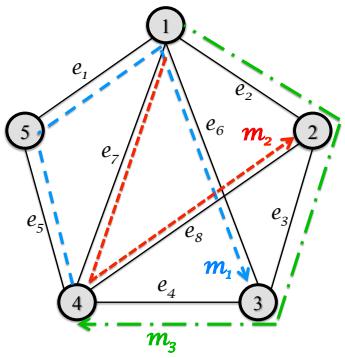


Fig. 2: An example network with three measurements

Let $\mathrm{x} \in \mathcal{R}^N$ be a non-negative vector whose p-th entry is the value over link p, and $\mathrm{y} \in \mathcal{R}^m$ denotes the vector of m measurements whose q-th entry represents the total values of links in a connected sub-graph over the network. Let $\mathcal{A}$ be an $m \times N$ measurement matrix with its i-th row corresponds to the i-th measurement. $\mathcal{A}_{ij} = 1$ $(i = 1, ..., m, j = 1, ..., N)$ if and only if the i-th measurement includes link j and zero otherwise. For example, for a network with $|V| = 5$ nodes, $|E| = 8$ links and $m = 3$ path measurements in Fig. 2, the measurement matrix $\mathcal{A}$ is:

$$\mathcal{A} = \begin{array}{c} \\ m_1: v_4 \rightsquigarrow v_3 \\ m_2: v_1 \rightsquigarrow v_2 \\ m_3: v_1 \rightsquigarrow v_4 \end{array} \begin{array}{c} \begin{array}{cccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

In the compact form, we can write the linear system as

$$\mathrm{y}_{m\times 1} = \mathcal{A}_{m\times N}\ \mathrm{x}_{N\times 1}, \tag{3}$$

where $m \ll N$. In sparse recovery, specially compressive sensing, the set of sparse solutions to this system are of interest. Thus, we need to add a constraint to limit the solution space. Now, the main question is how to estimate the link vector x from the path measurement y in the case of an underdetermined system $(m \ll N)$. This is still possible if we add a constraint that the vectors x are sufficiently sparse (*e.g.*, inter-community links are often much smaller than all links), which is often a reasonable assumption in networks $(k \ll N)$.

In the theory of compressive sensing, it is stated that the sparsest solution can be obtained by solving the following optimization problem [5; 8]:

$$\min_{\mathrm{x}} \|\mathrm{x}\|_0 \quad \text{subject to} \quad \mathrm{y} = \mathcal{A}\mathrm{x}. \tag{4}$$

It is known that solving Eq. (4) is NP-hard. Fortunately, it was shown that one can replace the ℓ_0-norm by ℓ_1-norm, and formulate the following problem instead [5; 8]:

$$\min_{\mathrm{x}} \|\mathrm{x}\|_1 \quad \text{subject to} \quad \mathrm{y} = \mathcal{A}\mathrm{x}. \tag{5}$$

With the combination of least squares and Eq. (5), we can change the objective function to have a possible solution for solving the linear system, even in presence of noise or truncated values in the matrix $\mathcal{A}$ and vector y, by:

$$\min_{\mathrm{x}} \|\mathrm{x}\|_1 + \|\mathcal{A}\mathrm{x} - \mathrm{y}\|_2^2. \tag{6}$$

This formulation is also known as LASSO [26; 27]. Consider that we have m measurements over the network. We, in this paper, would like to identify inter-community links from these measurements, with the knowledge that these links are sparse in social networks. It is important that sparse recovery over networks using compressive sensing has a closely related field called graph constrained group testing [28–30]. Note that compressive sensing can perform better than group testing based on the required number of measurements, as it is stated in [16]. Hence, we have used compressive sensing throughout this paper. In addition, CS may abstractly model complex systems even when the measurements from certain elements are not available. Therefore, our approach can be potentially used in other applications besides social networks.

III. Proposed Method

In this section, we propose a Compressive Sensing approach for inter-Community Detection (called CS-ComDet) in social networks. In this approach, we construct a *feasible* measurement matrix $\mathcal{A}$ to infer social networks and identify the inter-community links inside a network via *indirect* measurements. The constructed measurement matrix $\mathcal{A}$ from CS-ComDet should satisfy the condition of sparse recovery with network topological constraints, in which every measurement with non-negative integer entries has to be feasible in the sense that the links of the same measurement should correspond to a path or connected sub-graph.

The pseudo code of proposed method is shown in Algorithm 1. In this method, every row of the measurement matrix $\mathcal{A}$ is constructed from a measurement based on the CS-ComDet approach. As clearly depicted in Algorithm 1, this algorithm generally includes 7 steps:

- (i) Links weight $W(u,v)$ are calculated for all the links $(u,v) \in E$ in the graph $\mathcal{G}$ in a distributed fashion by letting each node-pair locally computes the local clustering coefficient using nodes degree and the degree of their neighbors, in lines (7)-(10).
- (ii) A first node is selected relative to $\mathrm{P}_{first}(v)$ which is calculated for all the nodes $v \in V$ in the graph $\mathcal{G}$, in lines (12)-(15).
- (iii) The transition matrix is constructed based on the transition probabilities P_{trans} in lines (18)-(21), such that $P_{trans}(v,u)$ is the probability of moving from node v to node u.
- (iv) The next node is selected under two different options proportional to the nodes probabilities in lines (17)-(28). Then, the traversed link removes.
- (v) The $update$ function is called in line (29) and performed according to the Algorithm 2.
- (vi) The steps (iii), (iv), and (v) are fulfilled 's' times which is the length of a measurement, to generate a new row for the matrix $\mathcal{A}$ in lines (16)-(31).
- (vii) All the previous steps are repeated 'm' times to construct a feasible measurement matrix with 'm' measurements in lines (11)-(33).

Algorithm 1 The Proposed Algorithm: CS-ComDet

Input: $\mathcal{G}(V,E), m, s$

1: $\mathcal{G}(V,E)$: Graph of the network
2: m: number of measurements
3: s: number of measurement steps
4: W = NULL /*Initializing Weight Matrix*/
5: $\mathcal{A}$ = NULL /*Initializing Measurement Matrix*/
6: P_{trans} = NULL /*Initializing Transition Matrix*/
——— /*Calculating Weights*/ ———
7: **Foreach** $(u,v) \in E$ **do**
8: $T(u,v)$ = Number of triangles with (u,v)
9: $W(u,v) = \frac{T(u,v)+1}{\min[(deg(u)-1),\ (deg(v)-1)]}$
10: **end for**
——— /*Constructing 'm' measurements*/ ———
11: **for** $i = 1 \rightarrow m$ **do**
12: **Foreach** $v \in V$ **do** /*First Node Selection*/
13: $P_{first}(v) = \frac{1}{|V|-1}\left(1 - \frac{W(v)}{\sum_{v \in V} W(v)}\right)$
14: **end for**
15: v_c = Select first node relative to P_{first}
—— /*Performing measurements with step size 's' */ ——
16: **for** $j = 1 \rightarrow s$ **do**
17: **if** $\exists\ u \in Nbr(v_c)$ **then** /*Next Node Selection*/
18: **Foreach** $u \in Nbr(v_c)$ **do**
19: $Score_u = \frac{W(v_c)}{W(v_c,u) \times \min(1, \frac{W(v_c)}{W(u)})}$
20: $P_{trans}(v_c,u) = \frac{Score_u}{\sum_u Score_u}$
21: **end for**
22: v_{next} = Select next node relative to $P_{trans}(v_c,u)$
23: Remove the link between v_c and v_{next}
24: $Nbr(v_c) = Nbr(v_c) - \{v_{next}\}$
25: $Nbr(v_{next}) = Nbr(v_{next}) - \{v_c\}$
26: **else**
27: v_{next} = Trace back to the previous node
28: **end if**
29: CALL $update(\mathrm{P}_{trans}, W, v_c, v_{next})$
30: $v_c = v_{next}$
31: **end for**
32: Add the measurement to the matrix $\mathcal{A}$ as a new row
33: **end for**

Output: measurement matrix $\mathcal{A}$

As we want to recover the inter-community links as a sparse property in social networks, we try to traverse these links more than intra-community links in the constructed measurement by CS-ComDet algorithm. To achieve this, every node-pair locally computes a weight for the link connecting them based on the *edge clustering coefficient* [18]. It is in analogy with the usual node clustering coefficient, as the number of triangles to which a given edge belongs, divided by the number of triangles that might potentially include it, given the degrees of the adjacent nodes. More formally, for the link $(u,v) \in E$, the edge clustering coefficient is [18]

$$C(u,v) = \frac{T(u,v)+1}{\min\left[(deg(u)-1),\ (deg(v)-1)\right]} \qquad (7)$$

where $T(u,v)$ is the number of triangles built on that link and $\min\left[(deg(u)-1),\ (deg(v)-1)\right]$ is the maximal possible number of them. The main idea behind this criterion is that

Algorithm 2 update($\mathrm{P}_{trans}, W, v_c, v_{next}$)

Input: W, $\mathrm{P}_{trans}, v_c, v_{next}$
1: W: Weight Matrix
2: P_{trans}: Transition Matrix
3: v_c: Current node
4: v_{next}: Next node
5: $\mathrm{P}_{trans}(v_c, v_{next}) = 0$
6: **Foreach** $u \in Nbr(v_c)$ **do**
7: Recalculate $\mathrm{P}_{trans}(v_c, u)$
8: Recalculate $\mathrm{P}_{trans}(u, v_c)$
9: **end for**
10: **Foreach** $u \in Nbr(v_{next})$ **do**
11: Recalculate $\mathrm{P}_{trans}(v_{next}, u)$
12: Recalculate $\mathrm{P}_{trans}(u, v_{next})$
13: **end for**

Output: P_{trans}

the links connecting nodes in different communities (inter-community links) are included in few or no triangles and tend to have small values for this coefficient. On the other hand, many triangles exist within communities. Therefore in step (i), we want each node-pair computes the weight $W(u, v) = C(u, v)$ for every link $(u, v) \in E$, and conduct the measurements based on the CS-ComDet method to visit inter-community links more with higher probability. In the sense that the inter-community links mostly tend to present a small value for the edge clustering coefficient that locally calculated for all links of the network.

In CS-ComDet approach, to efficiently recover sparse link vector, the maximum element in the measurement matrix $\mathcal{A}$ is upper bounded by 2. Therefore, in order to construct each measurement of $\mathcal{A}$, three situations for a link in the network $\mathcal{G}$ may be happened: (1) A link is not selected by that measurement, (2) it is visited once by that measurement and then removed (never visited again by that measurement), and (3) it is visited once and if there needs back tracking to the previous node, it is visited for the second time. Note that after a link removal, we need to update the transition matrix, So the *update* function calls in step (v). As shown in Algorithm 2, after removing the link $(v_c, v_{next}) \in E$, we recalculate the transition probabilities for v_c, v_{next}, and all their neighbors. We expect to have a more accurate method by this update function.

In the proposed method, we can avoid biasedness towards high-weighted links by selecting a "good start" node for every m measurements, and also assigning proper probabilities to the neighbors of nodes for selecting the best next node, according to steps (ii), (iii), and (iv). For every measurements, we first select a good start node proportional to the probabilities P_{first}, and then select the next node relative to the probabilities P_{trans}. The next node is chosen s times which is the length of a measurement, in step (vi). To calculate the transition probability, there are two steps: Scoring and Normalization, in step (iii). Because of link removal, it is possible that a node do not have any neighbor to select as a next node, thus, in this case we track back to the previous visited node, shown in line (27).

Overall, we construct a feasible measurement matrix with non-negative integer entries by using m measurements with the step size of s based on the CS-ComDet method, as stated in steps (vi) and (vii). In the proposed approach, each measurement go through a connected path which evidences feasibility of the measurement matrix $\mathcal{A}$. Therefore, for detecting the inter-community links, our method satisfies sparse recovery with network topological constraints. After generation of measurement matrix $\mathcal{A}$ via the CS-ComDet method and adding the accumulative sum of values of the visited links to the vector y for each measurement, we form the linear system of $\mathrm{y}_{m\times 1} = \mathcal{A}_{m\times N}\ \mathrm{x}_{N\times 1}$ and find the sparse solution for this system using Eq. (6).

Therefore, construction of measurement matrix $\mathcal{A}$ based on the CS-ComDet seems to be efficient for investigating social networks and identifying inter-community links. We severely offer the CS-ComDet approach for analysing complex networks, such as social interactions, biological networks, and technological networks. We will experimentally evaluate the performance of our approach with extensive simulations on various networks in the next section.

IV. Experimental Evaluation

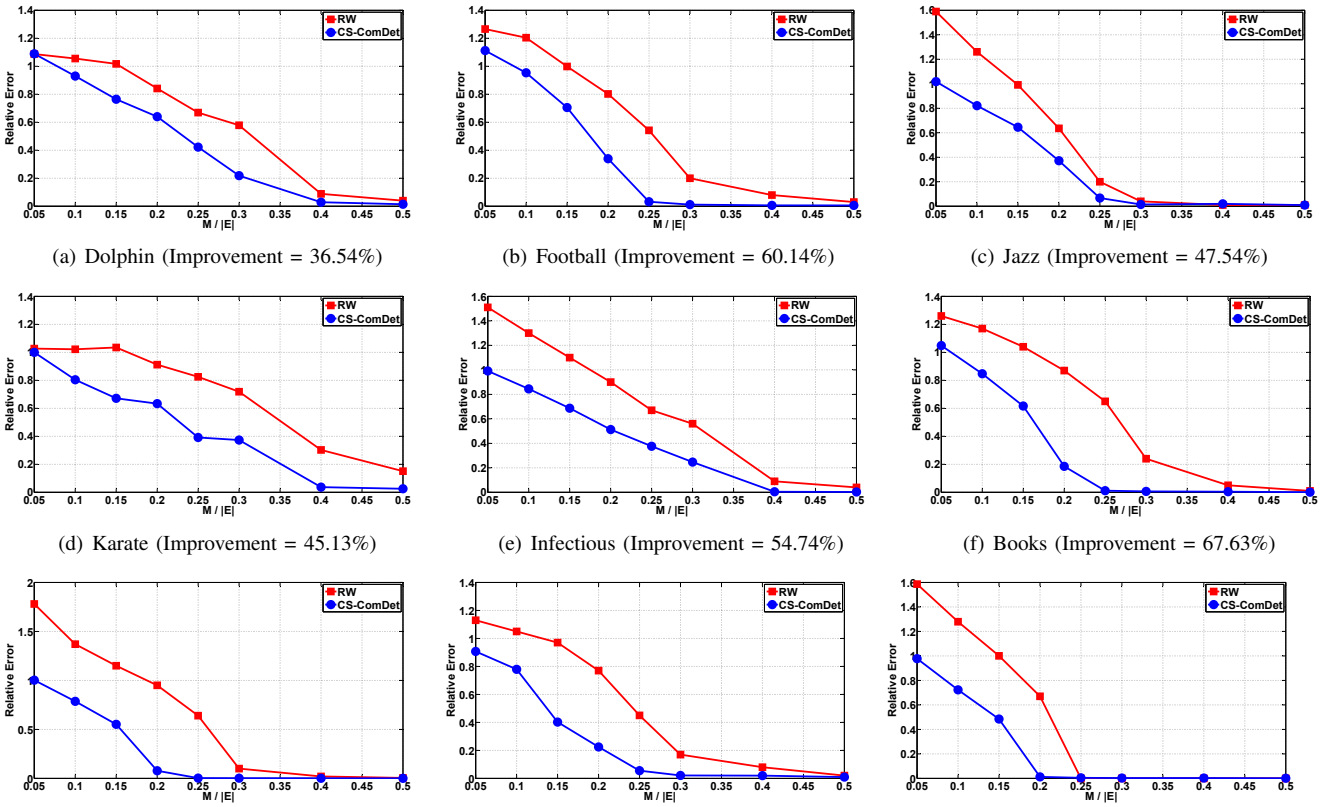
In this section, we evaluate the performance of the CS-ComDet method under various configurations. First, we introduce the real datasets we use for the evaluation. Next, we explain settings of the tests. Finally, the achieved results and their analysis are shown.

A. Datasets

To study performance of the proposed method, we consider some well-known real-world social networks as test data: (1) Zachary's Karate Club [31], (2) American College Football [32], (3) Jazz Musicians Network [33], (4) Dolphin Social Network [34], (5) Les Miserables - Coappearance Network [35], (6) Books about US Politics [36], (7) Infectious SocioPatterns [37], (8) Primary School - Cumulative Network [37]. We also consider a well-known real-world technological network to generalize our method in other complex networks: (9) The network of 500 busiest commercial airports in the United States (USTop500) [38]. Table I summarizes the details of these networks.

TABLE I: The details of real-world networks. From left to right: name of the network, the number of links ($|E|$) and nodes ($|V|$), density of the network (D), the average degree ($< deg >$), the number of communities (N_c), and modularity of the network (ϕ).

Networks	$\|E\|$	$\|V\|$	D	$< deg >$	N_c	ϕ
Karate	78	34	0.139	4.588	4	0.405
Dolphin	159	62	0.084	5.129	4	0.521
LesMis	254	77	0.087	6.597	6	0.565
Books	441	105	0.081	8.4	4	0.526
Football	613	115	0.094	10.661	9	0.601
Infectious	1781	332	0.032	10.729	27	0.83
Jazz	2742	198	0.141	27.697	4	0.444
USTop500	2980	500	0.024	11.92	11	0.282
School	5539	238	0.196	46.546	5	0.395



(a) Dolphin (Improvement = 36.54%) (b) Football (Improvement = 60.14%) (c) Jazz (Improvement = 47.54%)

(d) Karate (Improvement = 45.13%) (e) Infectious (Improvement = 54.74%) (f) Books (Improvement = 67.63%)

(g) School (Improvement = 75.57%) (h) LesMis (Improvement = 59.76%) (i) USTop500 (Improvement = 57.9%)

Fig. 3: Experiment 1: Recovery error for measurements of length $\frac{|E|}{5}$ with Sparsity $\frac{|E|}{10}$

B. settings

In each of the test cases for the real-world datasets, we generated 10 set of measurements. For each network and each set of measurements, we performed the experiments. The denoted points in the figures, represent the median value of the tests for all sets. For recovery error, we consider the relative error, specifically $\frac{\|x-x'\|_2}{\|x\|_2}$, where x and x′ are the original and predicted vectors, respectively. For the optimization step, we use SPAMS package on MATLAB [39]. We choose the LASSO model [26], [27] for the minimization that is explained in section II-C. In all of the test cases, we compare our CS-ComDet method with the work in [17] which we call RW in short. This work is one of the state-of-the-art methods for sparse recovery in networks which uses random walk to construct a feasible measurement matrix.

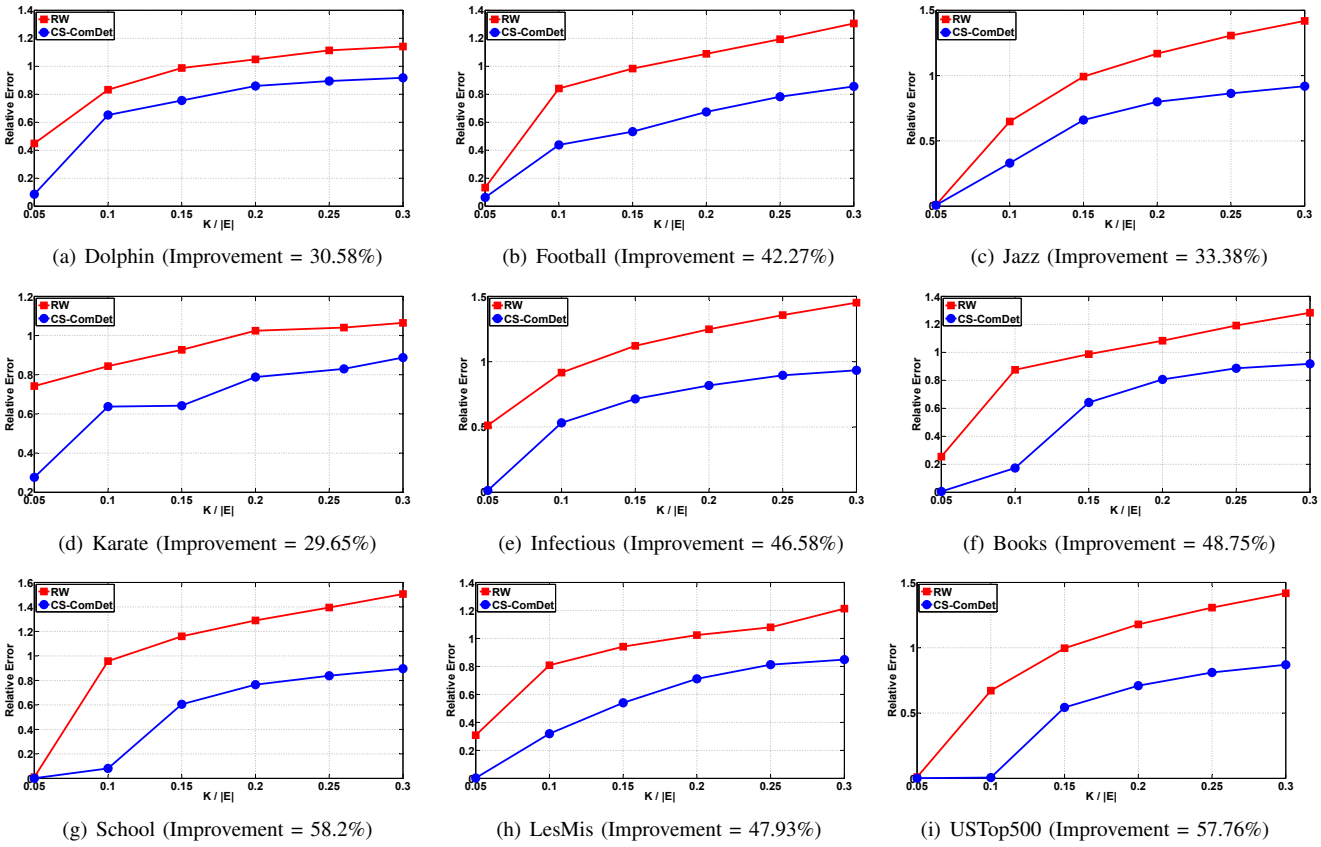
C. Evaluation Results

Experiment 1 (Effectiveness of Number of Measurements): Fig. 3 shows the performance evaluation of our method in comparison with the RW method, in terms of recovery error for different number of measurements. We set the sparsity (the number of non-zero elements) of the unknown vector to 10% of the number of links in each network. The length of each measurement in this test is $\frac{|E|}{5}$. Each point in the horizontal axis is proportional to the number of required measurements divided by the number of all links.

As it is shown, in all test cases, our CS-ComDet approach outperforms RW in terms of having lower recovery error for all number of measurements. In addition, our method gets lower error even in small number of measurements (*i.e.* when the number of measurements is less than half of the number of existing links in the network) compared to RW. This improvement can be very important in the situations where performing measurements has a high cost and the goal is to do an acceptable recovery on a reasonable cost.

The reason for the better results in recovery can be explored in many ways. First, in our approach we avoid traversing links repeatedly more than twice by the cases defined in the Algorithm 1. This leads to coverage of a greater part of the network, comparing to RW in which no particular measure is explicitly taken to avoid this issue. Second, an efficient neighbor selection method in the CS-ComDet measurements on the network, leads to have a fair coverage of all links. Hence, in our method, we cover more links and the end-to-end measurements will include more non-zero values in each measurement. Third, after each transition we call the *update* function, shown in Algorithm 2, to consider all changes and have a more accurate solution. Overall, we see around 56% improvement in average on all networks.

Experiment 2 (Effectiveness of Sparsity Percentage): In this experiment for all networks and for each percentage of recovery, we ran a set of measurements containing $\frac{|E|}{5}$



(a) Dolphin (Improvement = 30.58%) (b) Football (Improvement = 42.27%) (c) Jazz (Improvement = 33.38%)

(d) Karate (Improvement = 29.65%) (e) Infectious (Improvement = 46.58%) (f) Books (Improvement = 48.75%)

(g) School (Improvement = 58.2%) (h) LesMis (Improvement = 47.93%) (i) USTop500 (Improvement = 57.76%)

Fig. 4: Experiment 2: Sparsity effect for $\frac{|E|}{5}$ measurements of length $\frac{|E|}{5}$

measurements of length $\frac{|E|}{5}$. Fig. 4 shows the performance comparison for different sparsity in the unknown vector. It can be observed that even on high sparsity, we have the lower recovery error by our method. As it is shown, the CS-ComDet approach outperforms the RW method by around 44% improvement in average on all networks. The reasons for this improvement are the same as experiment 1: (1) The efficient neighbor selection method, (2) More coverage of the network, (3) Restricting the measurements by walking on the links no more than twice, and (4) Updating the transition matrix for each step of measurements via Algorithm 2.
As the final result, it can be seen that the CS-ComDet fulfils the requirements that we mentioned for a compressive sensing approach over social networks. Therefore, the CS-ComDet approach is an accurate solution to efficiently recover any k-sparse link vectors especially for identifying top-k inter-community links.

Experiment 3 (Visualizing Accuracy of The Method): As clearly depicted in Fig. 5, the detected links based on the proposed CS-ComDet method seems to be the inter-community links that accurately connect nodes of different communities in two different networks. According to the definition of community structure, there exist densely connected groups of nodes with only sparser connections (*i.e.* inter-community links) between groups. We only consider two networks for this experiment because of space limitation. Identifying these links are important and have several essential applications such as community detection, understanding global diffusion of information, predicting and recommending social connections for a user, measuring amount of friendship relation between communities, measuring activity rate between membership groups, and so on.

V. Conclusion

To the best of our knowledge, this is the first work that consider the community structure of the social networks in the context of compressive sensing which is an efficient tool for sparse recovery problem. A network is said to have community structure if there exist subsets of nodes within which node-to-node connections are dense, but between which connections are sparse. These sparse links are named inter-community links which are the sparse specification of link vectors in the networks. In this paper, we introduced a novel approach, called CS-ComDet, for the problem of recovering inter-community links in social networks. We used this method in the context of compressive sensing to construct a feasible measurement matrix under network topological constraints. We empirically evaluated the performance of our proposed method on several real-world networks in various aspects. Simulation results indicated that this approach can be employed to efficiently detect inter-community links accurately even on low number of measurements.

(a) Books (b) Karate

Fig. 5: Inter-community links detected by CS-ComDet method (denoted by bold lines) in two different datasets

REFERENCES

[1] M. E. J. Newman, A. L. Barabasi, and D. J. Watts, "The structure and dynamics of networks," *Princeton University Press*, 2006.

[2] M. E. Newman, "Scientific collaboration networks. i. network construction and fundamental results," *Phys. Rev. E-Stat, Nonlin, Soft Matter Phys*, vol. 64, no. 1, Jul. 2001.

[3] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75–174, Feb. 2010.

[4] Nielsen statistics and measurements, june 2010. [Online]. Available: http://blog.nielsen.com/nielsenwire/online_ mobile/social-media-accounts-for-22-percent-of-time-online.

[5] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[6] R. Berinde, A. Gilbert, P. Indyk, H. Karloff, and M. Strauss, "Combining geometry and combinatorics: a unified approach to sparse signal recovery," in *46th Annual Allerton Conference on Communication, Control, and Computing*, Sep. 2008, pp. 798–805.

[7] E. J. Candes, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[8] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.

[9] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[10] D. Donoho and J. Tanner, "Sparse nonnegative solution of underdetermined linear equations by linear programming," *Natl. Acad. Sci. U.S.A.*, vol. 102, no. 27, pp. 9446–9451, Mar. 2005.

[11] M. Coates, Y. Pointurier, and M. Rabbat, "Compressed network monitoring for ip and all-optical networks," in *ACM SIGCOMM IMC*, Oct. 2007, pp. 241–252.

[12] M. Firooz and S. Roy, "Network tomography via compressed sensing," in *IEEE Global Telecommunications Conference (GLOBECOM)*, Dec. 2010, pp. 1–5.

[13] J. Haupt, W. Bajwa, M. Rabbat, and R. Nowak, "Compressed sensing for networked data," *IEEE Signal Processing Magazine*, vol. 52, no. 2, pp. 92–101, Mar. 2008.

[14] H. Mahyar, H. R. Rabiee, and Z. S. Hashemifar, "UCS-NT: An Unbiased Compressive Sensing Framework for Network Tomography," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2013, Vancouver, Canada*, May 2013, pp. 4534–4538.

[15] H. Mahyar, H. R. Rabiee, Z. S. Hashemifar, and P. Siyari, "UCS-WN: An Unbiased Compressive Sensing Framework for Weighted Networks," in *Conference on Information Sciences and Systems, CISS 2013, Baltimore, USA*, Mar. 2013.

[16] M. Wang, W. Xu, E. Mallada, and A. Tang, "Sparse recovery with graph constraints: Fundamental limits and measurement construction," in *IEEE INFOCOM*, Mar. 2012, pp. 1871–1879.

[17] W. Xu, E. Mallada, and A. Tang, "Compressive sensing over graphs," in *IEEE INFOCOM*, Apr. 2011, pp. 2087–2095.

[18] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defning and identifying communities in networks," in *National Academy of Sciences of the United States of America*, Mar. 2004, pp. 2658–2663.

[19] L. C. Freeman, "The development of social network analysis: A study in the sociology of science," *BookSurge Publishing*, 2004.

[20] J. Chen and B. Yuan, "Detecting functional modules in the yeast proteinprotein interaction network," *Bioinformatics*, vol. 22, no. 18, pp. 2283–2290, Jul. 2006.

[21] F. Dourisboure, Geraci, and M. Pellegrini, "Extraction and classification of dense communities in the web," in *International Conference on World Wide Web (WWW)*, 2007, pp. 461–470.

[22] B. Krishnamurthy and J. Wang, "On network-aware clustering of web clients," in *SIGCOMM Comput. Commun. Rev.*, Oct. 2000, pp. 97–110.

[23] K. Reddy, M. Kitsuregawa, P. Sreekanth, and S. Rao, "A graph based approach to extract a neighborhood customer community for collaborative filtering," in *International Workshop on Databases in Networked Information Systems*, 2002, pp. 188–200.

[24] A. Y. Wu, M. Garland, and J. Han, "Mining scale-free networks using geodesic clustering," in *ACM International Conference on Knowledge Discovery and Data mining (SIGKDD)*, 2004.

[25] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev.*, vol. 69, no. 2, Feb. 2004.

[26] E. J. Candes, M. Rudelson, T. Tao, and R. Vershynin, "Error correction via linear programming," in *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, Oct. 2005, pp. 668–681.

[27] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society B*, vol. 58, pp. 267–288, 1994.

[28] P. Babarczi, J. Tapolcai, and P. H. Ho, "Adjacent link failure localization with monitoring trails in all-optical mesh networks," *IEEE/ACM Trans. Netw.*, vol. 19, no. 3, pp. 907–920, Jun. 2011.

[29] M. Cheraghchi, A. Karbasi, S. Mohajer, and V. Saligrama, "Graph constrained group testing," *IEEE Trans. Inf. Theory*, vol. 58, no. 1, pp. 248–262, Jan. 2012.

[30] N. Harvey, M. Patrascu, Y. Wen, S. Yekhanin, and V. Chan, "Non-adaptive fault diagnosis for all-optical networks via combinatorial group testing on graphs," in *IEEE INFOCOM*, May 2007, pp. 697–705.

[31] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Anthropological Research*, vol. 33, no. 4, 1977.

[32] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002.

[33] P. Gleiser and L. Danon, "Community structure in jazz," *Advances in Complex Systems*, vol. 6, no. 4, pp. 565–573, Jul. 2003.

[34] D. Lusseau, "The emergent properties of a dolphin social network," *Proceedings of the Royal Society of London. Series B: Biological Sciences*, vol. 270, pp. 186–188, Nov. 2003.

[35] D. E. Knuth, "The stanford graphbase: A platform for combinatorial computing," *Addison-Wesley, Reading, MA*, 1993.

[36] Mark Newman, A collection of network data sets, August 2013, http://www-personal.umich.edu/ mejn/netdata/.

[37] SocioPatterns datasets, August 2013, http://www.sociopatterns.org/datasets/.

[38] V. Colizza, R. Pastor-Satorras, and A. Vespignani, "Reaction-diffusion processes and metapopulation models in heterogeneous networks," *Nature Physics*, vol. 3, pp. 276–282, 2007.

[39] Sparse modeling software (spam). [Online]. Available: http://spams-devel.gforge.inria.fr/index.html