

Research Article

An Extreme Learning Machine-Based Community Detection Algorithm in Complex Networks

Feifan Wang, Baihai Zhang, Senchun Chai , and Yuanqing Xia

School of Automation, Beijing Institute of Technology, No. 5 Zhongguancun South Street, Beijing 100081, China

Correspondence should be addressed to Senchun Chai; chaisc97@163.com

Received 14 February 2018; Accepted 2 July 2018; Published 6 August 2018

Academic Editor: Shyam Kamal

Copyright © 2018 Feifan Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Community structure, one of the most popular properties in complex networks, has long been a cornerstone in the advance of various scientific branches. Over the past few years, a number of tools have been used in the development of community detection algorithms. In this paper, by means of fusing unsupervised extreme learning machines and the k -means clustering techniques, we propose a novel community detection method that surpasses traditional k -means approaches in terms of precision and stability while adding very few extra computational costs. Furthermore, results of extensive experiments undertaken on computer-generated networks and real-world datasets illustrate acceptable performances of the introduced algorithm in comparison with other typical community detection algorithms.

1. Introduction

As one of the most popular research fields over the past decades, complex networks have stimulated scientific advances in various fields such as biology [1], social networks [2], epidemiology [3], computer science [4], and transportations [5]. Numerous articles have explored different types of properties in complex networks. Among these articles, the study on community structure, which means vertices in a given network are inherently segregated into groups inside which the connections are relatively denser than those outside, has been one of the most popular [6]. Finding out the divisions of nodes in networks, which is called community detection or network clustering, is a hot spot for investigators because it is a good means to uncover the underlying semantic structure, mechanisms, and dynamics of certain networks [7]. Using such extracted information, internet service providers (ISPs) could set up a dedicated mirror server for intense web visits from the same geographic region to improve their customers' internet surfing experiences [8], and/or online retailers could provide more efficient recommendations to customers in favor of creating a more friendly purchase environment [9].

To address the community detection problem, researchers have developed numerous algorithms. Social network scientists used to solve this problem by traditional methods such as graph partitioning, hierarchical clustering, partitional clustering, and spectral clustering [7, 10]. Girvan and Newman proposed the first divisive algorithm named after them, which is a milestone historically because it introduced more physicists and computer scientists to this field [11]. Divisive algorithms use the concept of betweenness as criteria to judge how often an edge participates in a graph process and break up connections one by one to determine the most significant community structure [10]. A byproduct of Girvan and Newman's algorithm, called modularity Q , a quality function originally proposed as a criterion to decide when to stop the calculation, is another landmark that supports clustering methods focusing on the modularity optimization problem [12]. Although it has been proven impossible to list all the feasible divisions to determine the best strategy in deterministic polynomial time (the problem is np-hard) [13], many alternative approximate optimization techniques, including greedy algorithms [14], random walks [15], fast unfolding algorithm [16], information-theoretic framework [17], belief propagation [18], extremal optimization [19], simulated annealing [20], and genetic algorithms [21], have

been deployed to solve the problem. Along with the optimization tools of interest, many instruments have been involved in this field. For example, spectral algorithms explore eigenvalues of Laplacian matrices of graphs using traditional clustering techniques [22]. Similar to spectral clustering, similarity matrix factorization and blocking can also be applied [23]. Label propagation, that is, attaching labels to each node based on the neighbor information, is known as a fast and effective clustering method [24].

k -means clustering has long been one of the best off-the-shelf tools that exhibit relatively high precision and low computational complexity [25]. However, the performance of k -means relies too heavily on the selection of the initial centroids; hence many updates have been proposed to overcome this drawback. k -means++ chooses distinct initial seeds far from each other in a probabilistic manner, which leads to more stable clustering results but involves increased complexity [26]. Through ranking nodes in the same manner as Google’s cofounders did [27] and picking the center nodes from the highest ranking ones, k -rank achieves small fluctuation in the community detection output although it requires additional running time [28]. Another defect of k -means, explained by Ng et al. [29], is that it is only capable of finding clusters corresponding to convex regions. To address this problem, one could map the original data into a more suitable feature space. For example, Li et al. made use of principal component analysis (PCA) to implement k -means in a lower-dimensional space for community detection tasks [30].

The prevalence of extreme learning machines (ELM), originally proposed by Huang et al., should be largely credited to the simplicity of its implementation [31]. It has been demonstrated that given random input weights and biases of the hidden layer, a single-layer feedforward network (SLFN) could approximate any continuous functions simply by tuning the output weights [32]. As a result, the abstracted task in ELM is equivalent to a regularized least squares problem which can be solved in closed form using the Moore-Penrose generalized inverse [33]. Recently, semisupervised and unsupervised ELMs have been exploited based on the manifold regularization framework [34]. Regarding the clustering task, the unsupervised ELM can be interpreted as an embedding process that maps the input data into low-dimensional space.

In this paper, we propose an extreme learning machine community detection (ELM-CD) algorithm based on the combination of k -means and unsupervised ELM to fulfill the community detection task. Unsupervised ELM, inheriting the efficiency of ELM, has been utilized as a mapping mechanism that transforms the adjacency matrix to low-dimensional space, where k -means can be employed to label the groups. In consideration of no additional computational load, we prefer the original lite weighted k -means to other reinforced editions. Extensive comparison trials on both artificial networks and realistic networks indicate that ELM-CD outperforms traditional k -means in light of different precision criteria. Meanwhile, the introduced algorithm has remarkably low complexity, approaching that of k -means, and outperforms all other competitors evaluated.

The remainder of this article is organized as follows. Section 2 provides details of our algorithm. In Section 3, evaluations and comparisons are made in artificial and real-world networks. Finally, we conclude our work in Section 4.

2. Model and Algorithm

2.1. Preliminaries. We focus on an undirected network $G(\mathbf{V}, \mathcal{E})$, where \mathbf{V} represents the set of vertices numbering in total N and \mathcal{E} represents the set of edges with total number M . Neglecting self-loops, which mean an edge starts and ends at the same point, connections among nodes are expressed as a symmetric adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$.

$$A_{ij} = \begin{cases} 1, & \text{if vertex } i \text{ and } j \text{ are connected,} \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where $i = 1, \dots, N$ and $j = 1, \dots, N$.

According to \mathbf{A} , the Laplacian matrix $\mathbf{L} \in \mathbb{R}^{N \times N}$ is defined as

$$L_{ij} = \begin{cases} -1, & \text{if } A_{ij} = 1, \\ d_i, & \text{if } i = j, \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

in which $d_i = \sum_{j=1}^N A_{ij}$ is the degree of vertex i , $i = 1, \dots, N$, and $j = 1, \dots, N$.

A community detection algorithm segregates \mathbf{V} into mutually exclusive districts by means of attaching labels on each node to indicate which group it belongs to. Because \mathbf{A} is symmetric, each row or column of \mathbf{A} can be considered as an input sample denoted by $\mathbf{a}_i \in \mathbb{R}^N$, $i = 1, \dots, N$. Instead of directly assigning each node with a community label, ELM-CD first embeds \mathbf{A} into a smaller matrix $\mathbf{E} \in \mathbb{R}^{N \times n}$ in the n dimensional feature space, where k -means clustering proceeds to output a vector $\mathbf{t} \in \mathbb{N}_+^N$ whose items are the community labels of each node.

2.2. Embedding Process. Following the universal semisupervised and unsupervised ELM framework [34], given N_h neurons in the hidden layer, we define $\mathbf{E}_i \in \mathbb{R}^n$ as the output vector from ELM with respect to input vector \mathbf{a}_i .

$$\mathbf{E}_i = \sum_{j=1}^{N_h} \beta_j g(\mathbf{w}_j^T \mathbf{a}_i + b_j), \quad (3)$$

in which $\mathbf{w}_j \in \mathbb{R}^N$ and $b_j \in \mathbb{R}$ represent input weights and bias of the j th neuron in the hidden layer, respectively; $\beta_j \in \mathbb{R}^n$ is the output weight from the j th neuron to n dimensional output elements; and $g(\cdot)$ is the sigmoid function

$$g(x) = \frac{1}{1 + e^{-x}}. \quad (4)$$

Assume that the input dataset \mathbf{A} can be classified as unlabeled set $\{\mathbf{a}_i\}$, $i = 1, \dots, u$, and that u represents the number of unlabeled nodes and labeled set $\{\mathbf{a}_i, \mathbf{y}_i\}$, $i = 1, \dots, l$, where $\mathbf{y}_i \in \mathbb{R}^n$ is the corresponding community label and l is the number of labeled nodes so that $u + l = N$. The calculation of ELM would fall into two parts. In the first part, we randomly decide the input weights and biases according to a uniform distribution. In the second, let (3) be expressed as the inner product

$$\mathbf{E} = \mathbf{H}\boldsymbol{\beta}, \quad (5)$$

where

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{w}_1^T \mathbf{a}_1 + b_1) & \cdots & g(\mathbf{w}_{N_h}^T \mathbf{a}_1 + b_{N_h}) \\ \vdots & \cdots & \vdots \\ g(\mathbf{w}_1^T \mathbf{a}_N + b_1) & \cdots & g(\mathbf{w}_{N_h}^T \mathbf{a}_N + b_{N_h}) \end{bmatrix} \in \mathbb{R}^{N \times N_h}, \quad (6)$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \vdots \\ \boldsymbol{\beta}_{N_h}^T \end{bmatrix} \in \mathbb{R}^{N_h \times n}. \quad (7)$$

We define each line of \mathbf{H} as $\mathbf{h}_i \in \mathbb{R}^{1 \times N_h}$, indicating the output with respect to \mathbf{a}_i . The target of the second stage could be interpreted as a manifold-regularized optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \frac{1}{2} \sum_{i=1}^l C_i \|\mathbf{e}_i\|^2 + \frac{1}{2} \|\boldsymbol{\beta}\|^2 + \frac{\lambda}{2} \text{Tr}(\mathbf{E}^T \mathbf{L} \mathbf{E}) \\ \text{s.t.} \quad & \mathbf{e}_i^T = \mathbf{y}_i^T - \mathbf{h}_i \boldsymbol{\beta}, \quad i = 1, \dots, l, \end{aligned} \quad (8)$$

in which $\|\cdot\|$ is the Euclidean distance, $\text{Tr}(\cdot)$ calculates the trace of a matrix, and λ is a trade-off parameter. The first term of (8) is the loss function taking into account all labeled nodes with diverse penalty coefficients C_i . The second term refers to the classical antioverfitting regularization item that constrains the output weight to be as small as possible. The third term is the manifold regularization where the unlabeled data comes into play. Concretely speaking, the manifold regularization framework believes in the assumption that if two points on the manifold sphere are close to each other, then they would also result in similar predicted outcomes [35]. Consider the adjacency matrix as a measure of distance that two nodes are close to (connected to) each other if $A_{ij} = 1$, otherwise they will be far away (disconnected) from each other. The approximation of the manifold regularization is provided as

$$R_m = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N A_{ij} \|\mathbf{E}_i - \mathbf{E}_j\|^2 = \text{Tr}(\mathbf{E}^T \mathbf{L} \mathbf{E}). \quad (9)$$

This regularization penalizes the large difference in prediction of two connected nodes.

When there exist no labeled nodes in the input dataset and substituting (5) into (8), the optimization formulae could be reformulated as

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \|\boldsymbol{\beta}\|^2 + \lambda \text{Tr}(\boldsymbol{\beta}^T \mathbf{H}^T \mathbf{L} \mathbf{H} \boldsymbol{\beta}) \\ \text{s.t.} \quad & (\mathbf{H}\boldsymbol{\beta})^T \mathbf{H}\boldsymbol{\beta} = \mathbf{I}_n. \end{aligned} \quad (10)$$

The additional constraint abides by the suggestion of Belkin and Niyogi in the case of the degeneration of the solution [36], and \mathbf{I}_n is an n dimensional identity matrix.

Let γ and \mathbf{v} denote the eigenvalue and the corresponding eigenvector, respectively. When $N \geq N_h$ and \mathbf{H} have full column rank, it has been proven in [34] that solving (10) is equivalent to selecting n generalized eigenvectors corresponding to the n smallest eigenvalues of the problem.

$$(\mathbf{I}_{N_h} + \lambda \mathbf{H}^T \mathbf{L} \mathbf{H}) \mathbf{v} = \gamma \mathbf{H}^T \mathbf{H} \mathbf{v}. \quad (11)$$

To organize the matrix $\boldsymbol{\beta}$, we abandon the first eigenvector because it corresponds to eigenvalue 0 and contributes little in the embedding process. Thus, given the first $n + 1$ smallest eigenvalues $\gamma_1, \gamma_2, \dots, \gamma_{n+1}$ sorted by ascending order and their corresponding eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{n+1}$, the output weights $\boldsymbol{\beta}$ are

$$\boldsymbol{\beta} = [\tilde{\mathbf{v}}_2, \tilde{\mathbf{v}}_3, \dots, \tilde{\mathbf{v}}_{n+1}], \quad (12)$$

in which $\tilde{\mathbf{v}}_i = \mathbf{v}_i / \|\mathbf{H} \mathbf{v}_i\|$, $i = 2, \dots, n + 1$, indicates the normalized eigenvectors.

If $N < N_h$, meaning that the dimension of \mathbf{A} is smaller than the number of neurons in the hidden layer, the proven formulation of (11) is underdetermined and has the following alternative formulation:

$$(\mathbf{I}_n + \lambda \mathbf{L} \mathbf{H} \mathbf{H}^T) \mathbf{v} = \gamma \mathbf{H} \mathbf{H}^T \mathbf{v}. \quad (13)$$

In turn, the solution is

$$\boldsymbol{\beta} = \mathbf{H}^T [\tilde{\mathbf{v}}_2, \tilde{\mathbf{v}}_3, \dots, \tilde{\mathbf{v}}_{n+1}], \quad (14)$$

where the normalized eigenvectors are given by $\tilde{\mathbf{v}}_i = \mathbf{v}_i / \|\mathbf{H} \mathbf{H}^T \mathbf{v}_i\|$, $i = 2, \dots, n + 1$.

Finally, we substitute $\boldsymbol{\beta}$ into (5) to obtain the embedding matrix \mathbf{E} that would be fed into a k -means clustering algorithm to determine the community labels \mathbf{t} .

2.3. Clustering Process. In this article, an implementation of the original k -means clustering [25], owing to its low computational complexity, has been integrated into ELM-CD. First, ELM-CD randomly selects k rows in \mathbf{E} as the initial centroids of k clusters. Second, taking the Euclidean distance as the standard, each row in \mathbf{E} , which is represented by \mathbf{e}_i , $i = 1, \dots, N$, is assigned to a cluster whose centroid is closest to a certain row. Then, for each cluster, we calculate the mean value of all its members and designate this mean vector as the new centroid. We then iterate the cluster assignment and the centroid update processes until no row in \mathbf{E} changes its community label.

We show the entire procedure of the ELM-CD algorithm in Pseudocode 1.

```

ELM-CD (A)
1 initialize constant parameters, including  $\lambda, N_h, n$ 
2 generate  $L$  from  $A$ 
3 normalize  $A$ 
4 randomly generate input weights and biases of neurons in hidden layer
5 calculate  $H$  according to Eq. (6)
6 if  $N \geq N_h$ 
7   select the first  $n + 1$  smallest eigenvalues of Eq. (11) and assemble  $\beta$ 
   using the corresponding eigenvectors as in Eq. (12)
8 else
9   select the first  $n + 1$  smallest eigenvalues of Eq. (13) and assemble  $\beta$ 
   using the corresponding eigenvectors as in Eq. (14)
10 archive the embedding matrix  $E = H\beta$ 
11 randomly select  $k$  rows from  $E$  as centroids, say  $c_1, c_2, \dots, c_k$ 
12 while  $t$  changed
13   for  $i = 1$  to  $N$ 
14     for  $j = 1$  to  $k$ 
15        $dist_j = \|e_i - c_j\|$ 
16     find the smallest element in  $dist$  and its corresponding index  $s$ 
17      $t_i = s$ 
18   for  $i = 1$  to  $k$ 
19     update  $c_i$  to be the mean value of all members in cluster  $i$ 
20 return  $t$ 

```

PSEUDOCODE 1: (pseudocode of the ELM-CD).

3. Experiments and Results

In this section, we deploy our proposed algorithm ELM-CD on computer-generated networks as well as four well-known real-world networks to compare its performance with that of three competitors, including two divisive algorithms: GN [11] and NF [37], and conventional k -means [30]. All experiments took place using MATLAB R2016a running on an AMD Athlon(tm) X4 740 3.2 GHz desktop with 4 GB RAM. The constructed ELM network has $N_h = 2000$ neurons and randomly initialized input weights between -1 to 1 with all biases being ignored (set to 0). Following the empirical configuration in [34], we set the embedding feature space dimension $n = 3$ and the trade-off parameter $\lambda = 0.1$.

3.1. Modularity and NMI. Our first criterion, the modularity Q proposed by Newman and Girvan [38], is defined as the margin between the density of intracommunity edges on the estimated network and the density on a randomly reorganized network with the same number of nodes and edges.

$$Q = \frac{1}{2M} \sum_{i=1}^N \sum_{j=1}^N \left(A_{ij} - \frac{d_i d_j}{2M} \right) \delta_{\mathcal{G}_i \mathcal{G}_j}, \quad (15)$$

where $d_i = \sum_{j=1}^N A_{ij}$ still represents the degree of vertex i as before and \mathcal{G}_i indicates the group label of node i . δ_{ij} is the Kronecker delta, leading to

$$\delta_{\mathcal{G}_i \mathcal{G}_j} = \begin{cases} 1, & \text{if vertex } i \text{ and } j \text{ belong to the same group,} \\ 0, & \text{otherwise.} \end{cases} \quad (16)$$

A larger Q corresponds to a more apparent community structure in the underlying network and vice versa.

The normalized mutual information (NMI), another de facto benchmark for community detection, has also been deployed to evaluate the accuracy of algorithms [39]. Based on information theory, the NMI of two division methods A and B to partition N nodes is defined as

$$I(A, B) = \frac{-2 \sum_{i=1}^{C_A} \sum_{j=1}^{C_B} N_{ij} \log(N_{ij} N / N_i N_j)}{\sum_{i=1}^{C_A} N_i \log(N_i / N) + \sum_{j=1}^{C_B} N_j \log(N_j / N)}, \quad (17)$$

where C_A and C_B are the numbers of communities in A and B , respectively, and N_{ij} denotes the number of nodes that both appear in group i of A and in group j of B , where $N_i = \sum_{j=1}^{C_B} N_{ij}$ and $N_j = \sum_{i=1}^{C_A} N_{ij}$. The larger $I(A, B)$ indicates the more similar A is to B . Given the exact community distribution result B and the candidate algorithm A , $I(A, B) = 1$ means that the algorithm has found all the communities identical to the real structure while $I(A, B) = 0$ indicates a complete failure of the algorithm.

3.2. Computer-Generated Networks. We fabricate networks with community structures per Girvan and Newman's method [11]. In this case, 128 vertices are separated into four communities uniformly, meaning 32 nodes in each group, and edges are randomly added between each pair of nodes with probability P_{in} when the two nodes come from the same community and P_{out} for vertices belonging to different groups. In restraint of $P_{in} > P_{out}$ and the average degree of a vertex $z = 16$, through changing P_{out} or the average number of intercommunity edges per vertex z_{out} (the two are

equivalent), we can achieve different networks with obvious (small z_{out}) or ambiguous (large z_{out}) community characteristics. These computer-generated networks with decided graph knowledge are then fed into our algorithm and its competitors to calculate their NMI, and the results are presented in Figure 1.

Each point in Figure 1 represents the average NMI of a certain algorithm on 50 randomly generated scenarios with predefined z_{out} varying from 0 to 8. The error bars accompanying those points mark the normalized deviation of NMI in 1000 experiments. Lines connecting points are included solely as a guide. The community detection resolution of all methods drops with increasing z_{out} ; however, our algorithm outperforms all competitors, especially when z_{out} is approaching 8, which means the community structure is becoming extremely ambiguous. Even when $z_{\text{out}} = 8$, ELM-CD could still correctly classify half the number of nodes. Another outstanding feature of our algorithm is its comparative robustness. Different from what happens to ELM-CD, both GN and k -means run competitively in terms of accuracy until $z_{\text{out}} = 6$, but performance noticeably deteriorates after that.

3.3. Real-World Networks. In this part, four real-world networks with decided numbers of communities have been selected as test beds for measurement of our algorithm’s accuracy as well as its running times. Table 1 shows the four datasets in detail. For each scenario, we took 1000 trials for each of the three probabilistic algorithms NF, ELM-CD, and k -means. For the decided algorithm GN, only one trial is sufficient.

First, we checked each algorithm’s understanding of community structures based on the definition of modularity. For GN, ELM-CD, and k -means, we need to predefine the number of communities as one of the input parameters. Practically, we varied this parameter from 1 to 10 for the first three networks (Karate, Dolphins, and Polbooks) and from 1 to 20 for the last (football). Among all partition results, the clustering scheme with the highest Q was chosen along with its corresponding NMI scores. Table 2 lists these experimental results and shows that ELM-CD exhibits obvious but not dominant superiority compared with the other algorithms. Specifically, compared with k -means, ELM-CD demonstrates a stronger capability to determine more modular structures, simultaneously with higher NMI in most instances. We cannot ignore the higher modularity scores of GN and NF obtained in the dolphin network, but we should also bear in mind that in this test, GN and NF determined five and four clusters, respectively, compared to ELM-CD’s three clusters.

Second, we fixed the predefined number of communities as the inherent number for each network and calculated the NMI of each algorithm to evaluate how close the difference between the discovered community structure and the given underlying structure could be. Meanwhile, the consumed time was recorded to compare the complexity of the algorithms. We display the maximum NMI accompanied by the corresponding modularity and the calculation time in Table 3. From this table, in spite of the minor improvement

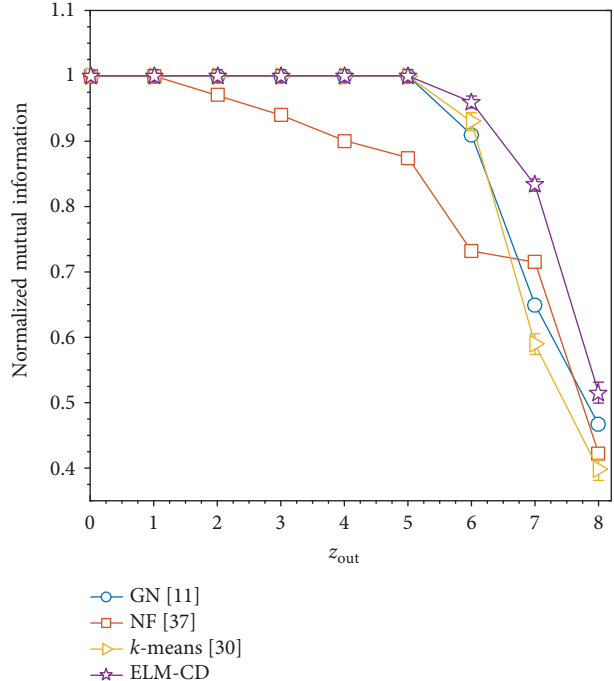


FIGURE 1: NMI of different algorithms on computer-generated networks.

TABLE 1: Information of datasets.

Dataset	Number of nodes	Number of edges	Number of communities
Karate	34	78	2
Dolphins	62	159	2
Polbooks	105	441	3
Football	115	613	12

TABLE 2: Maximum Q values and corresponding NMI of different algorithms.

Dataset	Index	GN	NF	k -means	ELM-CD
Karate	Modularity	0.4013	0.3807	0.3715	0.4198
	NMI	0.7072	0.6925	0.1.0	0.6873
Dolphins	Modularity	0.5196	0.4955	0.4216	0.4617
	NMI	0.5016	0.5727	0.4964	0.6548
Polbooks	Modularity	0.5120	0.5020	0.4509	0.5221
	NMI	0.4018	0.4396	0.5525	0.5538
Football	Modularity	0.5950	0.5682	0.5779	0.5797
	NMI	0.4237	0.4436	0.8490	0.8207

in precision, GN is much slower in comparison with the other three algorithms. Regarding NF, although it runs fast enough, the low modularity and NMI results represent a shortfall in the overall performance. ELM-CD no doubt consumes more time than k -means, whose computational complexity is $O(N \cdot k)$ where k is the predefined number of

TABLE 3: Compared results for maximum NMI, corresponding Q values, and time elapsed (s) of different algorithms.

Dataset	Index	GN	NF	k -means	ELM-CD
Karate	Modularity	0.3600	0.3870	0.3715	0.3715
	NMI	0.8365	0.6925	1.0	1.0
	Running time	5.5980	$7.6e-4$	$8.6e-4$	0.0098
Dolphins	Modularity	0.3787	0.4923	0.3555	0.3787
	NMI	0.8888	0.6208	0.8084	0.8888
	Running time	10.2244	0.0020	0.0012	0.0171
Polbooks	Modularity	0.4831	0.5020	0.4248	0.4530
	NMI	0.5754	0.5308	0.4144	0.6650
	Running time	199.2434	0.0064	0.0033	0.0293
Football	Modularity	0.5973	0.5773	0.5523	0.5585
	NMI	0.9214	0.7624	0.8954	0.8944
	Running time	808.1076	0.0120	0.0021	0.0308

communities, because ELM-CD requires the ELM calculation expenditure, which costs $O(N_h)$ time. Theoretically, once k is determined and when $N \gg N_h$, the complexity of ELM-CD is asymptotically $O(N)$, of the same as k -means. On the other hand, owing to the application of ELM, the stability of ELM-CD is much better than that of k -means. We demonstrate this by listing the average modularity and NMI results of the three fast algorithms in Table 4.

In Table 4, k -means illustrates very poor stability (even worse than that of NF) in light of the large differences between the average results and the maximum results shown in Table 3. In contrast, ELM-CD overcomes this inherited shortcoming due to the inclusion of the manifold regularization term in the equivalent optimization process as depicted in (8). Another trend exposed is that the robustness of k -means and ELM-CD is stronger for large networks with higher numbers of communities. The graphical clustering results of ELM-CD in Table 2 and Table 3 have been illustrated in the left and right parts of Figures 2–5, respectively.

3.3.1. Zachary’s Karate Club Network. From 1970 to 1972, Zachary observed a karate club as an example of a social network, which captures 34 members in the club and documents 78 pairwise links between members who interacted outside the club [40]. The decided community structure formed after a series of factional confrontations between the administrator “John” and instructor “Mr. Hi” (pseudonyms) led to the split of the club into two. Half of the members formed a new club around Mr. Hi, and members from the other part found a new instructor or gave up karate.

Figure 2 shows the partition results of our algorithm corresponding to the scenarios in Table 2 and Table 3, respectively. We use triangles to mark the new group belonging to the administrator, and circles represent the instructor’s. From Figure 2(a), ELM-CD converges to four groups to reach the largest modularity, although it redundantly separates each new club into two smaller ones. When the number of communities is decided, ELM-CD was able to detect the difference after the confrontation occurred in the club, which is demonstrated in Figure 2(b).

TABLE 4: Average NMI and Q values of the three fastest algorithms.

Dataset	Index	NF	k -means	ELM-CD
Karate	Modularity	0.3807	0.1071	0.2374
	NMI	0.6925	0.3960	0.4340
Dolphins	Modularity	0.4941	0.2818	0.2788
	NMI	0.5943	0.3336	0.5728
Polbooks	Modularity	0.5019	0.3360	0.4861
	NMI	0.5282	0.3739	0.5461
Football	Modularity	0.5680	0.5126	0.5086
	NMI	0.7424	0.8396	0.8262

3.3.2. Dolphin Social Network. The second network we make use of here is an undirected social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand [41]. These dolphins, naturally separated by their sex into two categories, were studied by Lusseau et al. for several years.

In Figure 3, we present snapshots of ELM-CD clustering results for the largest Q and the maximum NMI, respectively. Again, triangle and circle markers have been used to represent the two identical genders known ahead of time. For the best effort segregation in light of the modularity criteria, ELM-CD splits the female community into two subgroups, as shown in Figure 3(a). Node 40 appears to be the key factor that has the greatest effect on NMI results. In both images of Figure 3, node 40 is the only one that has been incorrectly classified. As a matter of fact, in Figure 3(b), the maximum NMI would be 1 rather than 0.8888 if node 40 could be assigned to the correct group.

3.3.3. Political Book Network. The third benchmark is a network of books on US politics published around the time of the 2004 presidential election and sold by <https://www.amazon.com/> [42]. Edges between books represent frequent copurchasing of books by the same buyers. In total, 105 books, denoted as nodes, have been classified into three categories as liberal, neutral, and conservative according to their political orientations.

In Figure 4, these political tendencies have been identified by triangle (conservative), circle (liberal), and pentagon (neutral) markers, respectively. The identifications made by our algorithm, on the other hand, are shown in three different colors. Red and blue colors represent the conservative and liberal groups, respectively, while green indicates the neutral part, which occupies a small number of nodes. It is clear from Figure 4 that people tend to have relatively stable opinions on politics as indicated by the obvious segregation of these books. However, it is not very uncommon that some neutral books appear more liberal or conservative oriented in the eyes of some critical readers, while the political orientations of some liberal or conservative books are obscure and may be considered as neutral. This explains why our algorithm can make mistakes on the boundaries between different classes. Moreover, from these two images, some books could have been classified more reasonably. For example, node 78 has four connections to conservative books

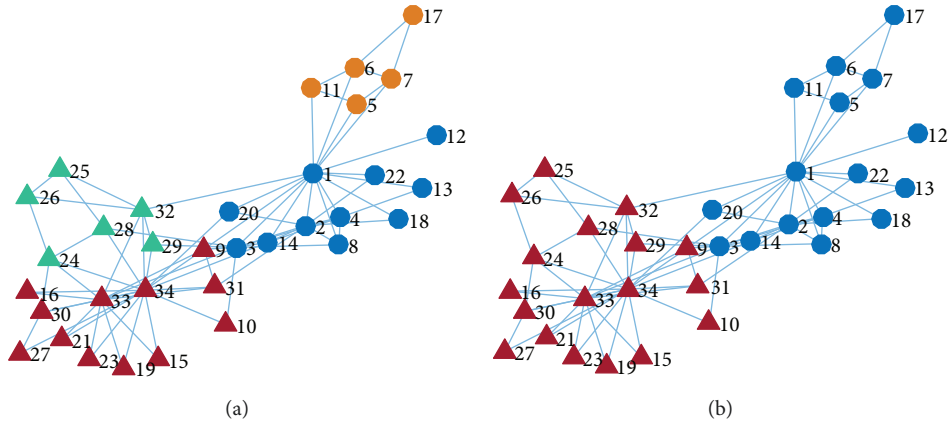


FIGURE 2: Communities extracted from Zachary's karate club network using ELM-CD. Different shapes indicate the inherent classification, while different colors show the detection results of ELM-CD. (a) Clustering result with the largest Q . (b) Given $k = 2$ the scenario with the best NMI ($=1$).

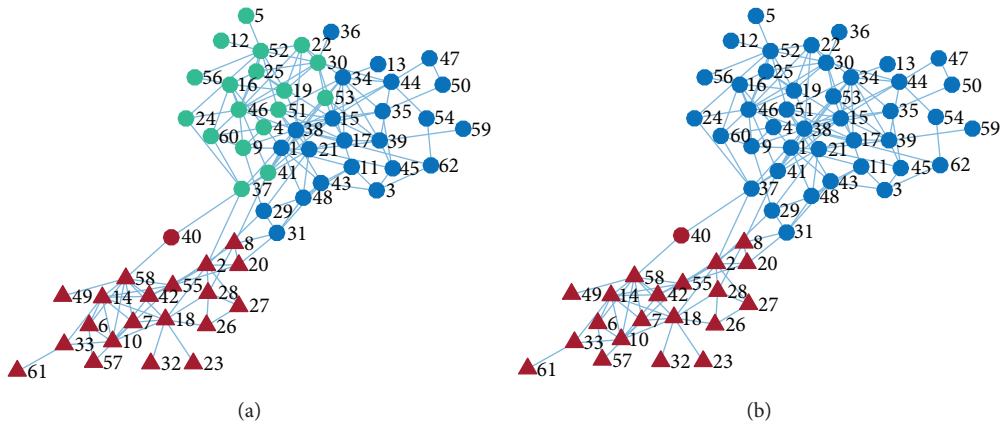


FIGURE 3: Communities extracted from the dolphin social network using ELM-CD. Different shapes indicate the inherent classification, while different colors show the detection results of ELM-CD. (a) Clustering result with the largest Q . (b) Given $k = 2$ the scenario with the maximum NMI.

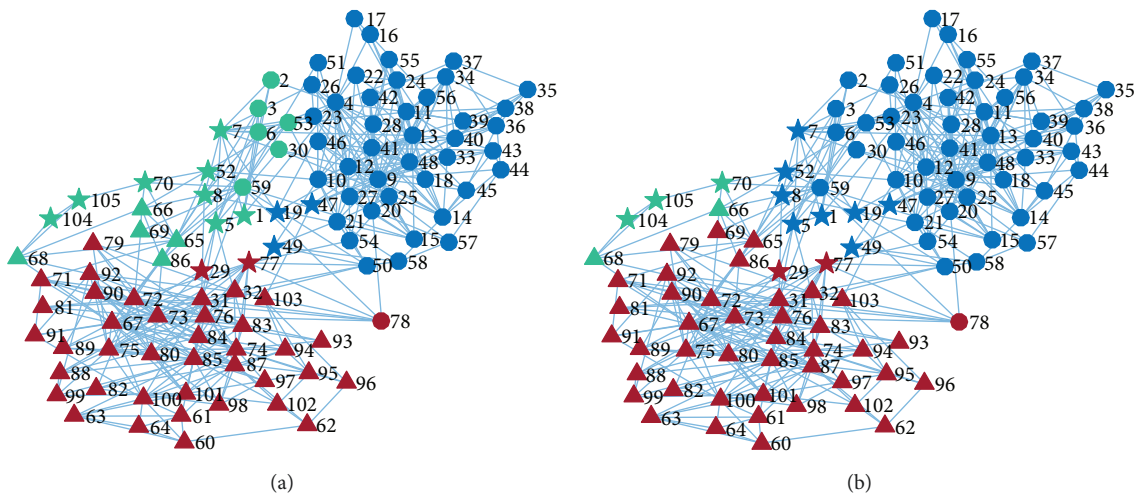


FIGURE 4: Communities extracted from the political book network using ELM-CD. Different shapes indicate the inherent classification, while different colors show the detection results of ELM-CD. (a) Clustering result with the largest Q . (b) Given $k = 3$ the scenario with the maximum NMI.

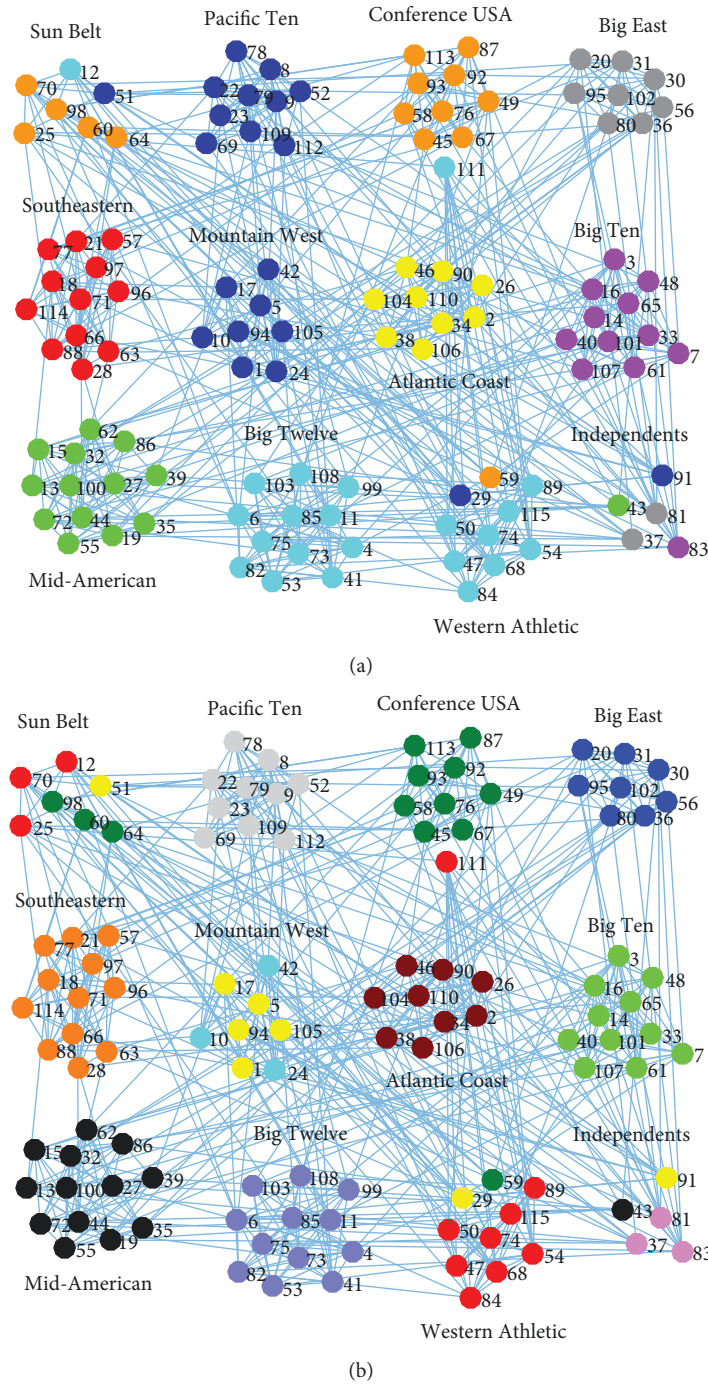


FIGURE 5: Communities extracted from the US college football network using ELM-CD. Twelve conferences have been placed in groups with conference names as labels. The detection results of ELM-CD are highlighted in different colors, each of which indicates one detected community. (a) Clustering result with the largest Q . (b) Given $k = 12$ the scenario with the maximum NMI.

while only sharing two connections with liberal ones; nevertheless, it is defined as a liberal book. Another example is the neutral node 77 that has no link to any other neutral books. Both the best efforts in Figures 4(a) and 4(b) have successfully corrected these two unreasonable classifications.

3.3.4. US College Football Network. The last example is the US college football network. Girvan and Newman introduced

this network of American football games between Division I colleges during the 2000 regular season, where 115 teams (nodes) from 12 conferences played 613 games (edges) in total [11]. On average, there are seven intraconference games and four interconference games, meaning that each team has a greater chance of confronting a competitor in the same conference. This feature gives us a good opportunity to deploy our algorithm to evaluate its accuracy.

In Figure 5, each conference has been located in a cluster relatively far from the other conferences and the community detection results have been recolonized using different colors. Aiming at the highest score in modularity, ELM separates these teams into eight classes. From Figure 5(a), except for the Southeastern and the Atlantic Coast conferences, all others have been misunderstood to some extent. Some of them, say the Pacific Ten and the Mountain West in dark blue, have been merged into a larger group. Some have been disassembled and displaced into other groups, such as the Independents and the Sun Belt. Such categorization could be rational because conferences in a larger group do compete with each other more frequently than smaller conferences. In contrast, teams in the Independent conference, except for nodes 81 and 83, share no game with each other; hence, it is no surprise to see them in other categories. When the number of communities is restrained to 12, as shown in Figure 5(b), ELM-CD performs well, with $NMI = 0.8944$ according to Table 3. This time, most conferences have retained the majority of nodes to organize independent communities, except for the Sun Belt, which has been fragmented, and the Mountain West, which contains two subclasses. However, we also have to note some unusual clustering behavior in this procedure, such as node 37 being placed in the same group with nodes 81 and 83, having nothing to do with 37 at all. Both of the snapshots in Figure 5 on some level expose the irrationality of the original conference definitions. For instance, node 111 belongs to Conference USA, but this team has always played its games with opponents from other conferences, which is contrary to the definition of a community.

4. Conclusions

In this article, we have proposed a community detection algorithm by combining the extreme learning machine and k -means techniques. Compared with the traditional k -means clustering method, the most evident advantages of the introduced algorithm, named ELM-CD, are the increase in precision and stability while adding very little calculation. ELM-CD has also been tested on Girvan and Newman's artificial networks and on four real-world networks and has been compared to GN, NF, and conventional k -means based on modularity and NMI benchmarks. From the implementation of ELM as an embedding procedure ahead of k -means clustering, ELM-CD demonstrates competitive performances both in accuracy and complexity.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors sincerely thank Dr. Xiao Zhang from the University of Michigan for his consultation and assistance. This work was funded in part by the National Natural Science Foundation of China under Grant no. 61573061.

References

- [1] J. Chen and B. Yuan, "Detecting functional modules in the yeast protein-protein interaction network," *Bioinformatics*, vol. 22, no. 18, pp. 2283–2290, 2006.
- [2] A. Garas, A. Lapatinas, and K. Poullos, "The relation between migration and FDI in the OECD from a complex network perspective," *Advances in Complex Systems*, vol. 19, article 1650009, 2016.
- [3] C. Stegehuis, R. van der Hofstad, and J. S. H. van Leeuwen, "Epidemic spreading on complex networks with community structures," *Scientific Reports*, vol. 6, no. 1, article 29748, 2016.
- [4] H. Zhao, L. Li, H. Peng, J. Xiao, Y. Yang, and M. Zheng, "Fixed-time synchronization of multi-links complex network," *Modern Physics Letters B*, vol. 31, no. 2, article 1750008, 2017.
- [5] W.-B. Du, X.-L. Zhou, O. Lordan, Z. Wang, C. Zhao, and Y.-B. Zhu, "Analysis of the Chinese Airline Network as multi-layer networks," *Transportation Research Part E: Logistics and Transportation Review*, vol. 89, pp. 108–116, 2016.
- [6] M. E. J. Newman, *Networks: An Introduction*, Oxford University Press, Oxford, UK, 2010.
- [7] S. Fortunato and D. Hric, "Community detection in networks: a user guide," *Physics Reports*, vol. 659, pp. 1–44, 2016.
- [8] B. Krishnamurthy and J. Wang, "On network-aware clustering of web clients," *ACM SIGCOMM Computer Communication Review*, vol. 30, no. 4, pp. 97–110, 2000.
- [9] P. Krishna Reddy, M. Kitsuregawa, P. Sreekanth, and S. Srinivasa Rao, "A graph based approach to extract a neighborhood customer community for collaborative filtering," in *International Workshop on Databases in Networked Information Systems (DNIS 2002)*, pp. 188–200, Aizu, Japan, 2002.
- [10] M. J. Rattigan, M. Maier, and D. Jensen, "Graph clustering with network structure indices," in *Proceedings of the 24th international conference on Machine learning - ICML '07*, pp. 783–790, Corvallis, OR, USA, 2007.
- [11] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.
- [12] M. E. J. Newman, "Equivalence between modularity optimization and maximum likelihood methods for community detection," *Physical Review E*, vol. 94, no. 5, article 052315, 2016.
- [13] U. Brandes, D. Dellinger, M. Gaertler et al., "On finding graph clusterings with maximum modularity," in *International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2007)*, pp. 121–132, Dornburg, Germany, 2007.
- [14] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, no. 6, article 066111, 2004.
- [15] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 4, pp. 1118–1123, 2008.

- [16] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, article P10008, 2008.
- [17] M. Rosvall and C. T. Bergstrom, "An information-theoretic framework for resolving community structure in complex networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 18, pp. 7327–7331, 2007.
- [18] P. Zhang and C. Moore, "Scalable detection of statistically significant communities and hierarchies, using message passing for modularity," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 111, no. 51, pp. 18144–18149, 2014.
- [19] J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization," *Physical Review E*, vol. 72, no. 2, article 027104, 2005.
- [20] A. Medus, G. Acuña, and C. O. Dorso, "Detection of community structures in networks via global optimization," *Physica A: Statistical Mechanics and its Applications*, vol. 358, no. 2–4, pp. 593–604, 2005.
- [21] S. Li, Y. Chen, H. du, and M. W. Feldman, "A genetic algorithm with local search strategy for improved detection of community structure," *Complexity*, vol. 15, no. 4, pp. 53–60, 2010.
- [22] H. Zare, P. Shooshtari, A. Gupta, and R. R. Brinkman, "Data reduction for spectral clustering to analyze high throughput flow cytometry data," *BMC Bioinformatics*, vol. 11, no. 1, p. 403, 2010.
- [23] J. Chen and Y. Saad, "Dense subgraph extraction with application to community detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 7, pp. 1216–1230, 2012.
- [24] W. Li, C. Huang, M. Wang, and X. Chen, "Stepping community detection algorithm based on label propagation and similarity," *Physica A: Statistical Mechanics and its Applications*, vol. 472, pp. 145–155, 2017.
- [25] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, Berkeley, CA, USA, 1967.
- [26] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proceedings of the 18th ACM-SIAM Symposium on Discrete Algorithms (SODA'07)*, pp. 1027–1035, New Orleans, LA, USA, 2007.
- [27] L. Page, "The PageRank citation ranking: bringing order to the web," Stanford Digital Libraries Working Paper 9, 1998.
- [28] Y. Jiang, C. Jia, and J. Yu, "An efficient community detection method based on rank centrality," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 9, pp. 2182–2194, 2013.
- [29] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: analysis and an algorithm," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic (NIPS'01)*, pp. 849–856, Vancouver, BC, Canada, 2001.
- [30] L. Li, K. Fan, Z. Zhang, and Z. Xia, "Community detection algorithm based on local expansion k-means," *Neural Network World*, vol. 26, no. 6, pp. 589–605, 2016.
- [31] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, pp. 985–990, Budapest, Hungary, 2004.
- [32] G. B. Huang, L. Chen, and C. K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 879–892, 2006.
- [33] D. Serre, *Matrices: Theory and Applications*, Springer, 2010.
- [34] G. Huang, S. Song, J. N. D. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2405–2417, 2014.
- [35] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: a geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.
- [36] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
- [37] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Physical Review E*, vol. 69, no. 6, article 066133, 2004.
- [38] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, article 026113, 2004.
- [39] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, article P09008, no. 9, 2005.
- [40] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, 1977.
- [41] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Sloaten, and S. M. Dawson, "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations," *Behavioral Ecology and Sociobiology*, vol. 54, no. 4, pp. 396–405, 2003.
- [42] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 23, pp. 8577–8582, 2006.



Hindawi

Submit your manuscripts at
www.hindawi.com

