Computers, Materials & Continua
DOI:10.32604/cmc.2021.017870
Article

Tech Science Press

Advanced Community Identification Model for Social Networks

Farhan Amin[1], Jin-Ghoo Choi[2] and Gyu Sang Choi[2,*]

[1]Department of Computer Engineering, Gachon University, Gyeonggi-do, 13120, Korea
[2]Department of Information and Communication Engineering, Yeungnam University, Gyeongsan, 38541, Korea
[*]Corresponding Author: Gyu Sang Choi. Email: castchoi@ynu.ac.kr
Received: 15 February 2021; Accepted: 10 April 2021

Abstract: Community detection in social networks is a hard problem because of the size, and the need of a deep understanding of network structure and functions. While several methods with significant effort in this direction have been devised, an outstanding open problem is the unknown number of communities, it is generally believed that the role of influential nodes that are surrounded by neighbors is very important. In addition, the similarity among nodes inside the same cluster is greater than among nodes from other clusters. Lately, the global and local methods of community detection have been getting more attention. Therefore, in this study, we propose an advanced community-detection model for social networks in order to identify network communities based on global and local information. Our proposed model initially detects the most influential nodes by using an Eigen score then performs local expansion powered by label propagation. This process is conducted with the same color till nodes reach maximum similarity. Finally, the communities are formed, and a clear community graph is displayed to the user. Our proposed model is completely parameter-free, and therefore, no prior information is required, such as the number of communities, etc. We perform simulations and experiments using well-known synthetic and real network benchmarks, and compare them with well-known state-of-the-art models. The results prove that our model is efficient in all aspects, because it quickly identifies communities in the network. Moreover, it can easily be used for friendship recommendations or in business recommendation systems.

Keywords: Community detection; social network analysis; complex networks

1 Introduction

Complex networks have grown steadily to become a major area of scientific and technological research. The most interesting aspect of studying complex networks is that they arise in any field, such as engineering, physiology, biology, and business [1]. In the scientific fields, one of their common features is that they can be represented as a graph, with nodes as individual entities and links for interactions [2]. Generally, they share common structural properties that distinguish them from purely random graphs [3]. There are several methods used for the investigation of complex networks, among which, community identification is one of the important and useful techniques

used for useful information discovery [4,5]. A community or group consists of a dense set of connected nodes, whereas the links among these nodes are also a set, where the outside set is sparse [6]. Communities are also known as modules or clusters, for example, a group of people who exchange information share common properties or have similar functions within a network. Community detection is very important and helpful for several reasons. The identification process allows finding nodes in a network based on their structural positions [6]. It beneficial to find the hierarchical structure that may exist in many real-world interconnections [7].

Generally, community detection is a fundamental problem that exists in large-scale networks. In these networks the number of communities is unknown, or the community structure is not clear. A community comprises a group of individuals with diverse backgrounds but mutual interests in a real social grouping [8]. There are many real applications in which community detection is used. In sales–purchase networks, the community is the group of people (customers) having similar purchasing interests. In brain networks, the nodes together can perform local computations and also give insights into the structural units of the brain [9]. In citation networks, communities denote a group of related research papers published in one direction. This group is used to identify authors who share common interests. A variety of community identification algorithms for complex networks has been developed so far, based on diverse ideas, and can be classified into five main groups. The first class is the graph-partitioning algorithms. Generally, the algorithms discussed in [10] are more suitable, especially in cases where the number of communities is known, such as the spectral bisection [11] and the Kernighan et al. [12]. The second class is hierarchical clustering. This class is generally based on two philosophies: division and agglomeration. Division is based on the splitting of communities by removing links. In this thought process, a link is removed by finding nodes that have a little similarity. On the other hand, the agglomeration method is based on the merging of communities only when their similarity scores are high. In this class, famous algorithms are Fast Newman (FN) [13] and Girvan–Newman (GN) [14], based on division and agglomeration, respectively. In spectral clustering [15], community detection is performed by using an eigenvector and was discussed by Donath et al. [16]. Normalized spectral clustering methods were proposed by Jianbo et al. [17]. Recently, an interesting spectral clustering-based algorithm was proposed by Mahmood et al. [18], termed Sparse Subspace Communities with Fusion (SSCF). In this study, the authors discussed the problem associated with linear coding with a norm constraint. The fourth class is optimized modularity. Modularity is one of the most famous and best optimization methods used to find the quality of communities and is also considered an NP-hard problem. A verity algorithm is available in which the process of maximization and modularity is performed at a specific time. Examples of these genetic algorithms are simulated annealing and the greedy technique. The most important and promising community detection algorithm in this class is the Louvain method [19]. The fifth class is mainly based on the label propagation algorithm (LPA) [20]. In this algorithm, all nodes are initially assigned a unique label, and these labels are propagated across the network. This process is repeated until convergence is achieved. The leading algorithms are LPA, LPAm [21], and CK-LPA [22]. Furthermore, it is noted that certain algorithms, such as multi-objective evolutionary, local expansion, and evolutionary algorithms, adopted similar ideas from the above-mentioned categories.

We surveyed various studies and noted that community identification methods mostly fall within the scope of clustering. The fundamental concept of clustering is to use a local-node structure and to identify nodes by using similarity metrics [23]. But in node similarity measures, the distances among the nodes are not considered. This parameter is the global structure of a network. Conversely, algorithms like Infomap [24], Eigenvector [25], and LPA have global structure

information. Hence, due to the global perspective, we get all the information in a network. It is also noteworthy that the global structure information considered in these algorithms often decreases their effectiveness. Generally, earlier methods were mainly based on global information, which includes all network information and guarantees a good structure, but both the complexity and the cost are high [26]. In general, online social networks (OSNs) [27] are very large, and therefore, community identification in these networks is a challenging task [28]. The algorithms having local structure information might fall into a local optimum, even though they have low time complexity. It is a very interesting problem to balance both local and global information using time, as well as accuracy and complexity while designing these algorithms [29]. We noticed that in the above-stated community detection algorithms, the community structure is not clear. Additionally, few algorithms are not parameter-free. Therefore, it is necessary to build a new community detection algorithm for the research community to handle the above-stated issues. The communities in large-scale networks are very useful, and many applications use community identification algorithms to expose the internal hidden structures of the network. For example, it would be interesting to find users who have similar interests and behaviors in social networks, as well as similar customers on e-commerce platforms based on their shopping habits, etc. There are so many examples, such as making a group of social media subscribers, finding good advertising, and facilitating recommendations to readers. Similarly, in data networks, such algorithms are helpful identifying malicious user communities [30] and are also helpful in product recommendations for online shopping systems.

1.1 Problem Formulation and Motivation

Community detection is a fundamental problem in many disciplines [31]. A lot of research has been done in this field and it still needs more attention. Our primary motivation for this study is to present a strong community structure—enhancement model for real networks where the community structure is not clear. Generally, these algorithms are based on local or global methods and provide a good community. In these methods, the costs and complexity are very high. On the other hand, algorithms employing local methods fall into a local optimum and hence have less complexity. Therefore, it is a challenging task to select one of them and provide a good community structure. The basic factors that drive the generation of social networks, such as content, network topology, and community structure, have not been well investigated in the literature. Similarly, in most algorithms, prior information has been required. Therefore, to address these issues, we propose a combined community enhancement model for social networks by considering local and global information. In this study, we extend the concept of community detection by using social influence for large-scale networks. The role of the most influential or central nodes is important in various aspects. In particular, the most central nodes in a community have a large number of neighbors. The community centers might be far from each other, and the similarity among these nodes are greater. Therefore, global and local methods shed light on this area. Briefly, our extended model is based on the most influential node identifications, local expansion powered by label propagation (the local method), and the communities using network modularity (the global method).

1.2 Research Contributions

- This study offers an extended model to address the community identification problem for real networks where the community structure is not clear. Our model identifies and enhances the community structure of networks by using both local and global network properties.

- There is no prior information, such as the number of communities, etc., required in this model. Our model is completely parameter-free and does not need optimization of any object functions.
- It is suitable for large-scale networks because of the local information and has its acceptable time complexity. It is fast because it is not necessary to calculate the communities each time. Therefore, the runtime of our proposed algorithm is reduced.
- The efficiency and the effectiveness of our proposed model are measured through real-world and synthetic benchmarks. These results prove that our suggested community structure—enhancement model detects more accurate communities as compared to the state-of-the-art methods. Moreover, the efficiency is measured through normalized mutual information (NMI) and modularity. Finally, it is verified using synthetic and real networks with different communities and node sizes.

The rest of our study is organized as follows. In Section 2, we discuss various studies related to community detection. In Section 3, we discuss our proposed model. In Section 4, we discuss the achieved results. Finally, Section 5 offers conclusions from this study and suggests future work.

2 Related Work

The presence of communities in complex networks has gained a lot of interest from researchers in different fields. Rhouma et al. [32] proposed an algorithm to identify overlapping nodes in a network. This algorithm is based on the local optimization of a fitness function and fuzzy logic, which is used to identify the degree of belonging for various nodes. The membership of these nodes depends upon the community path length between the node and the members. They proposed an objective function to quantify the quality of a community in the network. The only problem with this model is that they did not use it for noisy networks. Generally, in noisy networks, the nodes change over a certain time interval. In this regard, it is not a suitable approach. Similarly, Javadi et al. [33] introduced a local community detection method that is based on the leadership concept. In this study, the leaders were identified over a certain time in the first step. In the next step, the community is identified. The initial set of leaders is called as the first snapshot. The problem with this approach is that the lifetime of these members was not discussed. A recent study was conducted by Ma et al. [34]. They proposed a new version of the LPA algorithm named modularity-based incremental LPA (MILPA). In LPA algorithm, each node has been randomly assigned through different labels [34]. Therefore, the label updating process is very difficult. Therefore, the accuracy and stability of this algorithm are quite low, and hence, it cannot be used for large-scale networks. Therefore, to handle this difficulty, the authors proposed an optimization function that is helpful in the labeling of nodes and in getting an optimal partition. Unlike the LPA, at first, each node is determined based on its degree and the membership of the nodes. Consequently, a node is assigned the same label. The objective function is used to guide the updating and labeling of nodes. They did not verify it for time complexity, and it was not designed for directed and weighted networks. Community detection is also helpful for understanding how large and complex networks are organized [35]. A lot of algorithms on community identification have been proposed so far. Generally, these algorithms fall into two main categories; the first is local, and the second is global. The local community identification algorithms deal with finding a community by using local information on the network [36]. Also, these algorithms are extendable. For example, if the local community identification algorithm is rapidly executed, there will be more chances to identify more local communities. Therefore, the community structure is clearer and more enriched. Conventionally, The community detection

algorithms require global information about the network [36], and it is very easy to examine and collect the information from a large-scale network having millions of nodes. Therefore, we have decided to study both local and global community detection algorithms. The combination of both methods provides a unique solution and will improve the efficiency of community detection in the network.

3 Proposed Model

In this study, we consider an undirected and unweighted graph denoted by G, and $G = (V, E)$ where V and E are vertices and edges, respectively. Each node in G denotes an element in the network, and each edge shows a link between pairs of nodes. In the network, $n = |V|$ denotes nodes, and $m = |E|$ denotes edges. The adjacency matrix is $A = (aij)_{n*n}$; if node i and j are connected by links, then $aij = 1$; otherwise, $aij = 0$.

3.1 Advanced Community Identification Model

Fig. 1 illustrates our proposed architectural model. In this figure, scientist applies sample data to our proposed model. Our model is based on the most powerful data sciences tool: Network X using Python. After receiving the query message, it identifies the community in a given data set in three steps. In the first step, the most influential nodes are discovered, and node ranking is performed. In the next step, similar nodes will be identified by using label propagation, and finally, the community formation is performed. We describe it in later subsections.

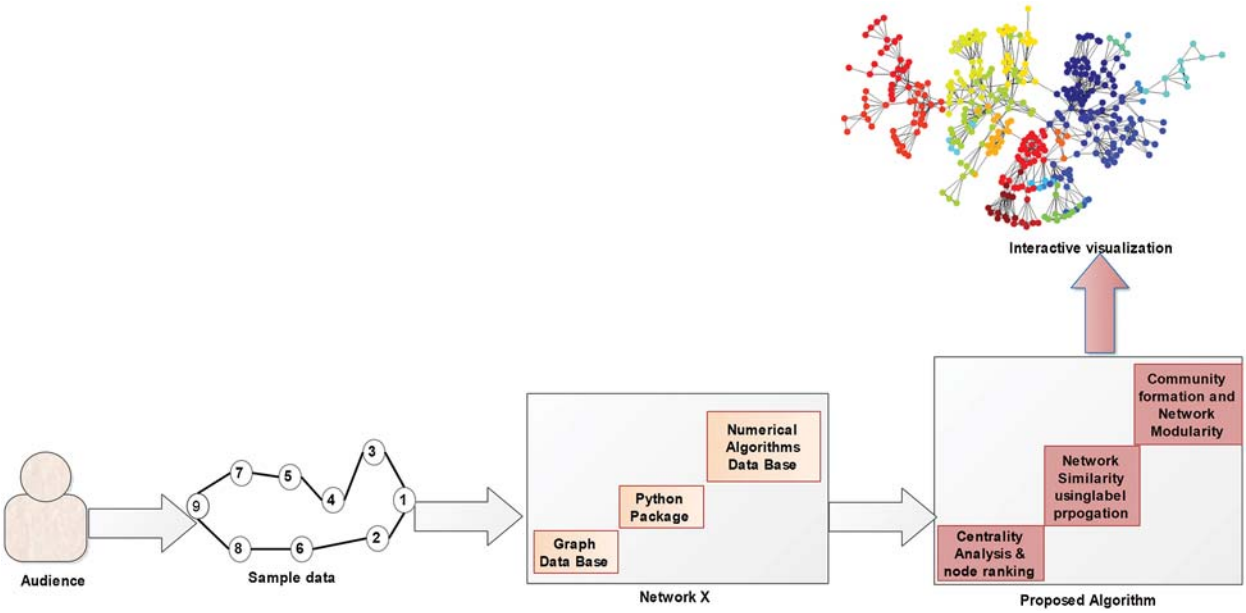

Figure 1: The proposed architectural model

3.1.1 Most Influential Node Identification and Ranking

The most influential nodes and the node ranking is performed in the first step. The node identification is based on the assumption that a central node in a community is connected to many nodes in the network. Tab. 1 explores details of the synthetic communication network data

set that we used in this experiment. In this table, we can see that we have a small number of nodes, i.e., 1–50, where each node is represented by a unique label, i.e., P_1, P_2,..., P_{50}, etc. The elements of this data set are inviter and invitee. There is a link between the inviter and invitee and is represented by Msgcount. Fig. 2 presents the visualization of the communication network data set. In this figure, each participant is denoted by a unique label (due to the small size, the labels are hidden), and the edge is a connection between these nodes. The top-10 nodes are identified and also node ranking is performed. In this extended study, we have modified our earlier work discussed in [37]. We used eigenvector centrality to find the most powerful nodes across the network. This centrality measure is used to discover how properly the nodes inside a network are connected. It also helps to find the types of relationships among them. Generally, the eigenvector is the sum of the centrality of all connections. Moreover, it is also considered a quality factor. Therefore, each node v in graph G calculates an eigenvector score based on Eq. (1), where x_i represents the eigenvector and is equal to the weighted sum of all nodes across the network:

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{N} A_{i,j} x_j \tag{1}$$

Table 1: Communication data network

Number	Inviter	Invitee	Msgcount
1	P_1	P_{63}	180
2	P_2	P_{64}	135
3	P_3	P_{65}	88
4	–	–	–
50	P_{34}	P_{102}	33

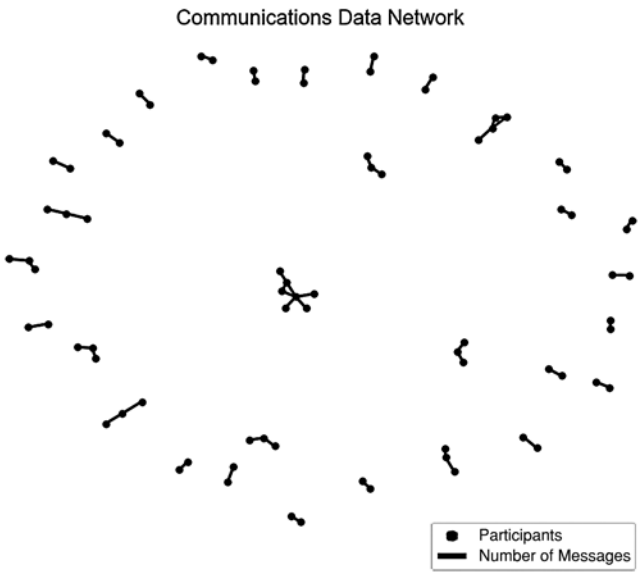

Figure 2: Visualization of communication data network

$A_{i,j}$ is the adjacency matrix, and the elements are denoted by i, j; λ is the largest eigenvalue of $A_{i,j}$. It is:

$$X = \lambda^{-1} AX \tag{2}$$

The largest Eigen score demonstrates the eigenvalue of node i. Also, if a node has more connections, then it will have more chances to be central. Therefore, we designed a function based on the eigenvalue. Our designed function can measure the number of links in the network. Besides, our function uses the power of the eigenvector, which helps to find the largest eigenvector value in a graph. Also, the eigenvector is proportional to the individual neighbor centralities. And so on, individual nodes will be contacted more often, along with individuals. Fig. 3 shows the process of finding the most influential nodes. In this figure, each node is connected to the network. We see that each node is labeled by using a white color, for example, P_{66},..., P_2, are connected. Tab. 2 exhibits the list of top-10 nodes. In this table, each node is labeled along with the computed score. Also, the computed centrality score is written against each node. The ranking is performed from the highest to the lowest number of nodes. Algorithm 1 demonstrates the procedure of identifying the most influential nodes, i.e., C_0.

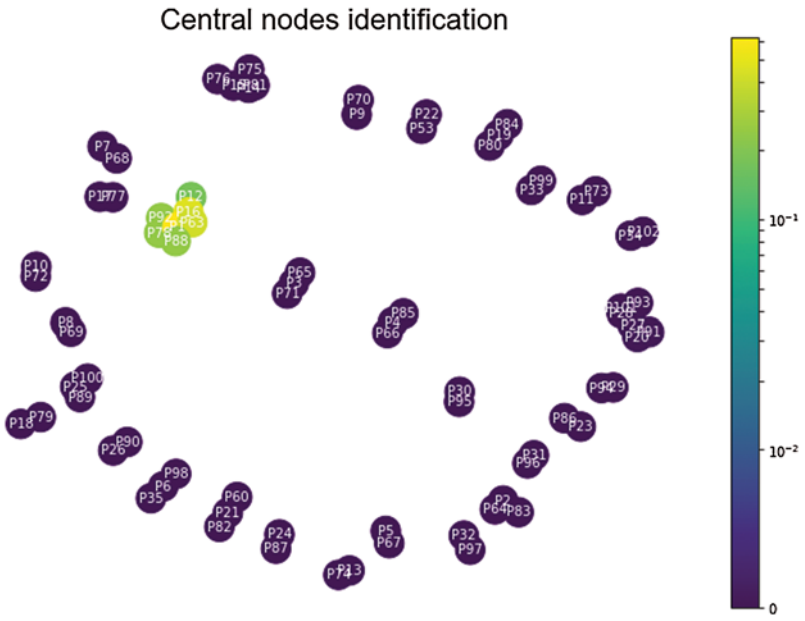

Figure 3: Most influential node identification

3.1.2 Local Subset Expansion via Label Propagation

We have designed a similarity function to identify the neighbors having high centrality. The similarity is calculated based on Eq. (3):

$$Similarity(\mathrm{i}, j) = A_{ij} \frac{|\tau_i \cap \tau_j|}{|\tau_i \cup \tau_j|} \tag{3}$$

Table 2: Top-10 central nodes with computed score

Number	Node label	Computed score
1	P_{63}	0.423
2	P_{88}	0.240
3	P_{91}	0.00015
4	P_{92}	0.240
5	P_{27}	0.00017
6	P_{78}	0.240
7	P_{12}	0.18227
8	P_1	0.623
9	P_{20}	0.00515
10	P_{16}	0.474

Algorithm 1: Central node identification and ranking

Input: Graph $G = (V, E)$
Output: Select the most influential nodes C_0; $C_0 = \{V_0, V_1, V_3, \ldots, V_k\}$
Start ()
Step 1: {Compute the node strength via Eq. (1)}
Step 2: {Sort the nodes based on node strength and select node C_0 to be the initial central node, where $V_i \in C_0$}
Step 3: {Update C_0 by $C_0 U V_j$, if $V_j \in V - C_0$ and satisfies Eq. (1)}
Step 4: {Repeat until no node in $V - C_0$ remains}
Step 5: {Return the final outcome as C_0 where $C_0 = \{V_0, V_1, V_3, \ldots, V_k\}$
End ()

where A_{ij} is the adjacency matrix, and $|\tau_i \cap \tau_j|$ is the number of neighbors that i and j have in common; $|\tau_i \cup \tau_j|$ is the total number of i and j neighbors, in which τ_i show the node i neighborhood up to the depth of the network.

We have used Jaccard Similarity Index (JSI). The JSI finds pairings of nodes in a given network. We have not used the overlap coefficient for this study. The deletion of a community in the network is completely based on one assumption: that one node belongs to only one community. The most suitable node pairing is achieved based on the ratio of the intersection of neighbors to the union of all of them, as shown in Eq. (3). This hub will get a pairing of nodes with the highest similarity. In this step, the initial community is discovered. Let the initial community be $\{C_1', C_2', C_3' C_4', C_5', C_6', C_k'\}$, where $C_j' = \{v_{ij}\}$ and $v_{ij} \in C_0$ for $i = 1, \ldots, k$. That is the most influential node corresponding to a community. In this step, the node label $v \in V - C_0$, and the color are the same as node $U \in C_0$ if v is the neighbor of U and satisfies:

$$Similarity\,(i, j) = \max_{i \in C_0} \max_{j \in N(vi)} sim(i, j) \tag{4}$$

The next step is to color a label for i to j, and if $i \in C_k'$, then we update C_j' by $C_j' Uv$. This process is iteratively propagated until all nodes in $v - C_0$ are colored. In this way, we identify several communities with the most influential leaders having the same color. Clearly, central nodes

with the greatest similarity among the six maximal similarities are shown, i.e. P_{60}, P_{21}, and P_{82}, etc. This step is explained in Algorithm 2.

3.1.3 Network Modularity and Community Combination

Modularity is an evaluation parameter used to find the quality of community structures in networks. Generally, this objective function provides aid during the process of calculating communities. The modularity is represented by parameter Q. Higher values for Q mean a better community structure. For that reason, the object is used to find the community assignment for each node in a network in such a way that Q is maximized. This is the final stage, in which the optimization function is used to merge two communities, $C_i^{'}$, $C_j^{'}$, in a pre-community into one $C_i^{'}$, $C_j^{'}$. In this way, the modularity of Q is achieved and satisfied:

$$Q\left(C_i^{'}UC_j^{'}\right) \geq max\left\{Q\left(C_i^{'}UC_j^{'}\right), 0\right\} \tag{5}$$

for any $C_i^{'}, C_j^{'} \in PreC$ where $Q = \frac{1}{4m}\sum_{v_i,v_j,}\left(A_{ij} - \frac{(d(v_i)(v_j)}{2m}\right)\delta\left(C_i^{'}, C_j^{'}\right)$ is the modified modularity derived from [37], in which A_{ij} is the element of the adjacent matrix, and $d(v_i)$ is the degree of v_i. In addition, $C_i^{'}$ is the community in which node v_i belongs, and $\delta\left(C_i^{'}, C_j^{'}\right)$ is an indicator function. If $\delta\left(C_i^{'}, C_j^{'}\right) = 1$, then $C_i^{'} = C_j^{'}$; otherwise, it is 0, and $\frac{1}{4m}\sum_{vi} \in d\,(v_i)$. In this way, any two communities can be merged. This process is iterated until modularity is achieved. When this process ends, an optimal community will be formed. Algorithm 3 shows the steps involved in this procedure.

Algorithm 2: Subset expansion via label propagation

Input: A network $G = (V, E)$, a central node C_0
Output: A pre-community, $C' = \{C_1^{'}, C_2^{'}, \ldots\ldots C_k^{'}\}$
Start ()
{Add node C_0 and all the first-step members of C_0 to the initial subset, C'}
{Let $\{C_1^{'}, \ldots, C_k^{'}\}$ be the initial communities, $C_j^{'} = \{V_{ij}\}$ and $V_{ij} \in C_0$}
{For $U \in C_1^{'}$ and $V \in v - C_0^{'}$,if similarity $\{u - v\}$ satisfies Eq. (2)}
{Then, update $C_j^{'}$ by $C_j^{'}U\{v\}$ and C_0 by $C_0U\{v\}$}
{Go to the process until $V - C_0 = \varnothing$. Return the pre-community outcome, $C' = \{C_1^{'}, C_2^{'}, \ldots, C_k^{'}\}$}
End ()

Algorithm 3: Proposed algorithm

Input: A network $G = (V, E)$
Output: Final communities set $C = \{C_1, C_2, \ldots, C_n\}$
Start ()
Step 1: {While there exist nodes for expansion, do}

(Continued)

Step 2: {Find the most influential nodes based on centralization C_0 using Algorithm 1}
Step 3: {Expand the community of a node, C_0, using Algorithm 2}
Step 4: {End while}
Step 5: {If there exists any node that does not belong to any community, add it to the outlier node set}
Step 6: {Generate a network, $G' = (V', E')$, in which the nodes are detected as communities}
Step 7: {While there exist any communities for the merge, do}
Step 8: {Combine C'_{i0} and C'_{j0} into one if they satisfy optimization function Eq. (4)}
Step 9: {Update C'_{i0} and C'_{j0} by $C'_{i0} U C'_{j0}$ in pre-community C'}
Step 10: {Repeat until modularity is no longer achieved}
Step 11: {End while}
Step 12: {Return all communities set C $= \{C_1, C_2, \ldots, C_t\}$}
End ()

3.2 Algorithm Explanation

Our proposed model is described in Algorithms 1–3. As it is based on the local approach, therefore is helpful to identify the small clusters in the network. Generally, the most influential nodes are the community core nodes that's why they are very helpful in community detection. To identify these nodes, we proposed a new linear centrality method. It has been given above in Algorithm 3. The first step is the computation of node strength using Eq. (1). It results in allocating a unique Eigen score to all nodes in the network. The nodes are sorted in decreasing order. The high-score node is nominated as an influential, or leader, node. Therefore, C_0 is known as the first leader node. This process is continuously repeated until community leaders are selected. The next step is the expansion of a community. The process of expansion is performed by using a similarity index. The similarity of the two neighbors is identified based on the JSI. Finally, the first community, C', is discovered. The next step is to perform node labeling. In this step, the leader and the members are discovered. This process is repeated continuously so that pre-commutation of the same color is accomplished, i.e., $C' = \{C'_1, C'_2, \ldots, C'_k\}$. There are two additional cases. In case 1 more than one neighbor was discovered have maximum similarity with the leader node, i.e., C and C'. Then, take the leader node into both communities and update both based on the choice of algorithm. If more than one node, say V_1 and V_2, having maximum similarity should be placed in the same community. The remaining nodes are labeled in the same way until all nodes are labeled. In case 2: The merging of two communities is performed through an optimization function. In this step, a call to the optimization function is made and it results in the combination of any two communities. This process is repeated until the modularity is achieved. At that time, the process of community merging is over, and hence, an optimal community is formed in the network.

4 Performance Evaluation and Experimental Results

In this section, we discuss the efficiency of our proposed algorithm by comparing it with current state-of-the-art algorithms. For simulation, we used the most powerful graph network analysis tool, Network X, for the implementation of our proposed model. This tool uses the powerful programming language Python for the creation and manipulation of data and to study the internal structure and dynamics of complex networks. Several network tools mostly use custom-compiled code and Python, whereas Network X focuses on computational network modeling instead of pure software modeling. Also, it is super-fast, due to adaptability features, such as Force Atlas,

and hence, it is perfect for getting immediate results. We obtained results from two aspects: first is real-world networks, and second is artificial networks.

4.1 Real-world Networks

To evaluate the performance of our proposed model, we tested it using small-scale and large-scale real datasets. In Tab. 3, Florentine families, the Zachary Karate Club, the Dolphin social network, American college football, and the Polbooks data network. The details of these data sets include nodes, edges, and a description. These real-world data sets are discussed in the next section. Similarly, for a large real-world network, we used Bright kite, Facebook, Amazon, and DBLP. Tab. 4 illustrates the description details of large real-world network datasets.

Table 3: Small real-world network data sets

Network name	Nodes	Edges	Description
Florentine families	16	20	Network of florentine families
Karate club	34	78	Zachary karate club network
American college football	20	616	American football network
Polbooks	105	441	Polbooks network
Dolphins	61	159	Dolphins network

Table 4: Large real-world network data sets

Networks	Nodes	Edges	Description
Bright kite	58,228	214,078	Location-based network.
Facebook	4039	88234	Facebook data set.
Amazon network	334,863	925,872	Data set comprised of many products listed on amazon.
DBLP	317080	1049866	DBLP data set.

4.1.1 Experimental Results Using Real-World Networks

Tab. 3 presents small real-world datasets. Florentine Families is the first dataset. It is an unweighted and undirected network that was collected from the historical documents of John Padgett. In this network, a collection of marriages and business ties among the most powerful families of Florence is described. Two ties are discussed: the first is business, and the second is wealth. Also, these social ties are all about marriages based on certain businesses among the 16 most powerful families. The second data set is named Zachary Karate club. It is one of the famous social network benchmarks comprising 34 members and 78 links. In this data set, some members belong form a small group around the coach; some members choose a new coach, and finally, the last group of members bails out of karate. The third data set is the American college football club network. It comprises football games played between the different college divisions during the regular season in the fall of 2000. The fourth data set is Polbooks [38], which comprises 105 nodes and 441 edges. This network is a collection of books created by Kerbs, and is a collection of the co-purchasing relationship of booksellers over the Amazon website. The edges between these books demonstrate the frequent co-purchasing of books by the same buyers. The fifth network is

named Dolphin social network [39]. It is formed by various dolphins playing together. It has 62 nodes and around 159 edges.

The large real-world data sets are listed in Tab. 4. Bright kite is a location-based network and is comprised of 5K nodes and 2K edges. The Facebook network is comprised of 4K nodes and 8K edges. The Amazon network is a product co-authorship network, where the nodes represent products and the edges represent the co-purchased products. In this network, each product belongs to one or more hierarchically organized categories provided by Amazon. We ran our proposed algorithm using these real network data sets. The identified communities are shown in Figs. 4 and 5. In Fig. 4, the identified communities using the karate club are presented. In this experiment, we have used different colors to represent different communities (yellow, sky, blue, and maroon). The four communities along with different node labels and different colors are shown. The first identified set is {26, 29, 22, 15, 18, 14, 20, 30}, the second is {4, 5, 6, 10, 16}, the third is {1, 2, 3, 19}, and the fourth is {23, 24, 25}. Fig. 5 presents the communities in the Florentine network. In these figures, four different communities are shown in different colors, such as; yellow, maroon, sky, and blue. The yellow color presents {Albizzi Guadagni, Lamberts, and Ginori}, the maroon color {Pazzi and Salviati}, the sky color {Barbadori, Ridolfi, etc.}, and finally the blue is {Strozzi and Peruzzi, etc.}.

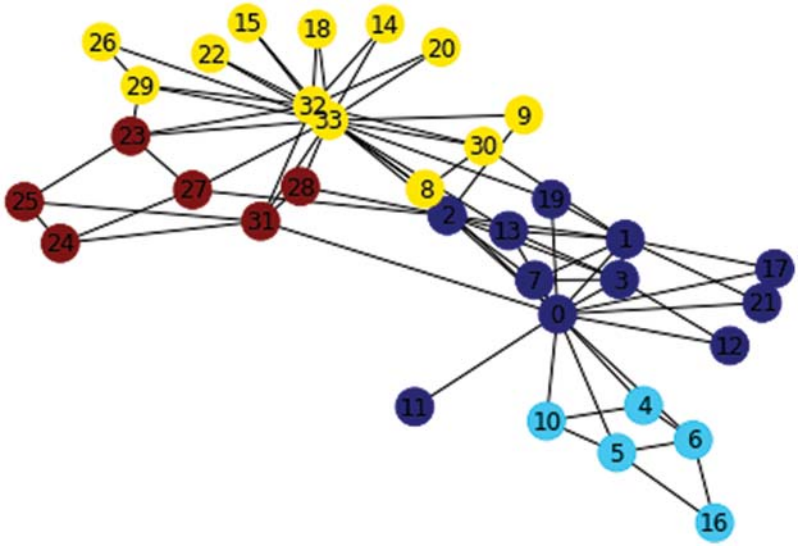

Figure 4: Identified communities in the Karate club network

4.2 Synthetic Networks

We have used Girvan–Newman, relaxed caveman (RC) [40], and Lancichinetti–Fortunato–Radicci (LFR) [41] artificial network to analyze and evaluate the performance of our proposed model. These standard synthetic benchmarks are used to verify the accuracy in communities.

4.2.1 Comparison with State-of-the-Art Algorithms

To measure the performance of our proposed algorithm, we compared our proposed model with the leading state-of-the-art algorithms. In this experiment, we have used the normalized mutual information (NMI) index and modularity. Before going into further detail, we first

introduce the NMI index [42] and the modularity [42]. The mutual index (MI) is used to understand the uncertainty of a variable [42]. The NMI is used to compare the original and the detected partitions in the network. The NMI measure is calculated based on Eq. 6:

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

where (X, Y) presents the two diverse partitions, and $H\,(X)$ and $H(Y)$ are both partition entropies, in which X and Y are the variables and the value of $I(X, Y)$ is between 0 and 1. This is known as the mutual information between X and Y. If variables X and Y are both independent, then X does not indicate any information about Y; in this case, the value of $NMI = 0$. Inversely, if variables X and Y are determined by each other, then all information covered by X is shared with Y, and the value of $NMI = 1$. In Fig. 6, Test a and Test b indicate the evaluation of clustering values for both measures. Blue and yellow are used to show the occurrence of both NMI and ARI scores. We can see that when the number of nodes are increasing, it will produce different AMI and ARI scores. We continuously repeated it several times using small to large numbers of nodes. We observed that increasing the number of nodes did not affect the performance of our proposed model if the time interval is increasing, i.e., for a small number of nodes it would be 0.072 s, and for a large number, it would be 26.321 s.

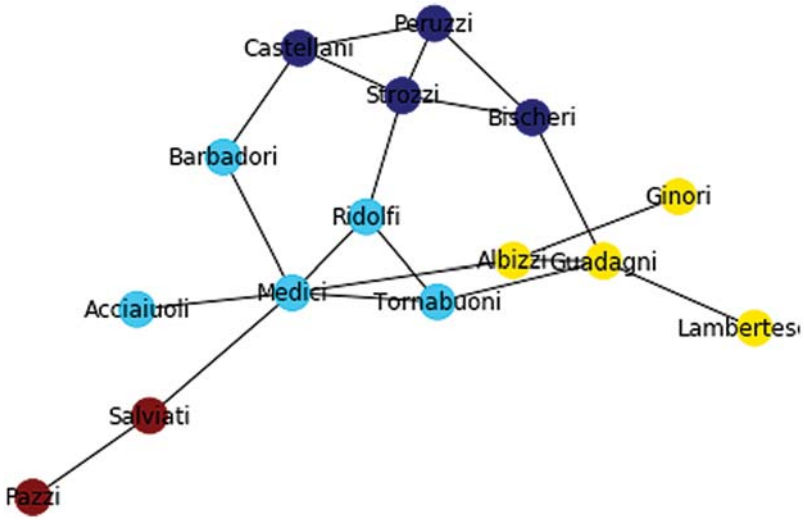

Figure 5: Identified communities in the florentine network

4.3 Algorithm Comparison Using Real-World Networks

Tab. 5 illustrates the description details of real data sets. In this table, we can see the results drawn from Figs. 4 and 5. In the table, each data set with complete parameters and the detected communities along with the modularity score are given. To test the efficiency of our proposed algorithm, we compared it with the top-three state-of-the-art algorithms, and the results are given in Tab. 6. In the table, the network along with the modularity score and the computed communities are given. We tested our proposed algorithm with both small numbers and large numbers

of networks. We observed that for a small network, the computed modularity was different from a large network, and the detected communities were larger. The computed NMI values and the modularity score of our proposed model and the current models are shown in Tab. 7. We see that there are only three communities identified in the football network, whereas when the same procedure is repeated for our proposed model, four communities were detected. Similarly, we repeated it for both eigenvector and LPA algorithms and observed that the incoming NMI values were larger than ours. Therefore, based on these results, we can say that our proposed model outperforms the other state-of-the-art algorithms. Briefly, our proposed model performs well based on the computed NMI and modularity score. Figs. 7 and 8 summarize the NMI and modularity score Q for various algorithms using real networks. In terms of NMI score, our proposed model performed well and produced good results, compared to the state-of-the-art methods, especially with the Dolphin, Florentine networks. In Fig. 7, we see that the proposed model outperformed, compared to the football and karate clubs. Because our proposed model is deterministic hence it is scalable.

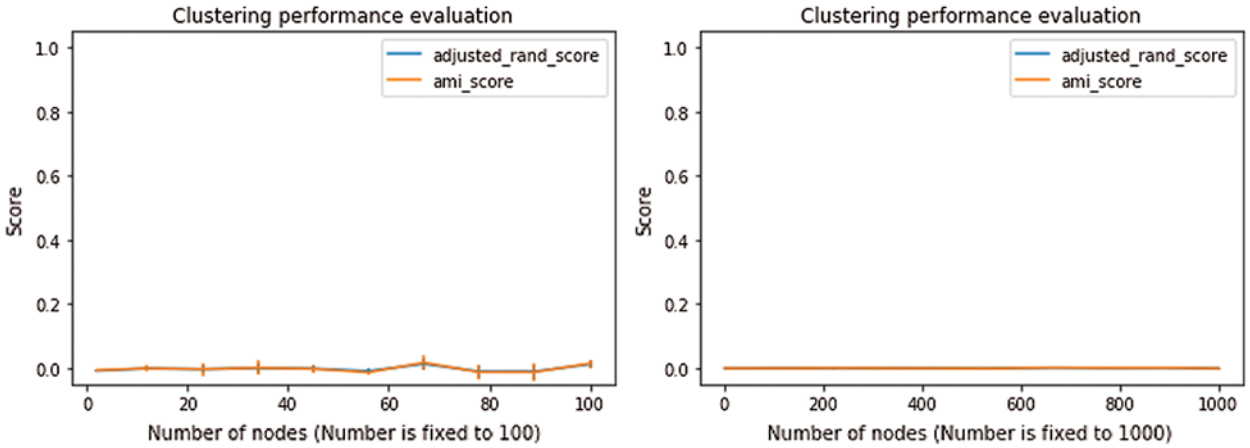

Figure 6: Test (a) for a small number of nodes Test (b) for a large number of nodes

Table 5: Real data sets and detected communities

Data set	Nodes	Edges	Detected communities	Modularity
Florentine families	16	20	7	0.3975
Zachary karate club	34	78	8	0.415
American college football	20	616	8	0.3
Polbooks	105	441	10	0.21
Dolphin social network	62	159	5	0.551

Table 6: Performance comparison using different network data sets

Network	Communities (C)	Modularity (Q)
Our network		
Dolphins	6	0.38
Karate club	4	0.37
American college football	12	0.55
Amazon	60686	0.210
DBLP	9421	0.501
Bright kite	1957	0.511
Info map		
Dolphins	2	0.38
Karate club	3	0.37
American college football	8	0.55
Amazon	29470	0.232
DBLP	0.714	24414
Bright kite	4810	0.578
LPA		
Dolphins	3	0.38
Karate club	4	0.37
American college football	6	0.55
Amazon	22523	0.783
DBLP	20820	0.674
Bright kite	1756	0.618

Table 7: Performance comparison using different real network data sets

Network	Communities (C)	NMI	Modularity (Q)
Our network			
Dolphins	2	0.53	0.41
Karate club	4	0.63	0.42
American college football	10	0.75	0.60
Info map			
Dolphins	2	0.53	0.39
Karate club	2	0.69	0.51
American college football	10	0.95	0.58
LPA			
Dolphins	2	0.52	0.51
Karate club	2	0.59	0.41
American college football	12	0.92	0.58

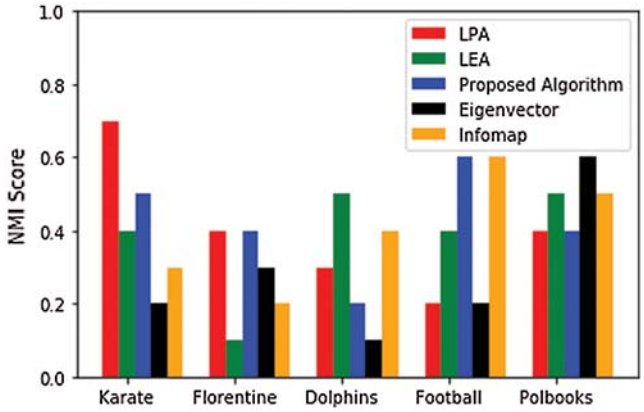

Figure 7: NMI score compassion using real networks

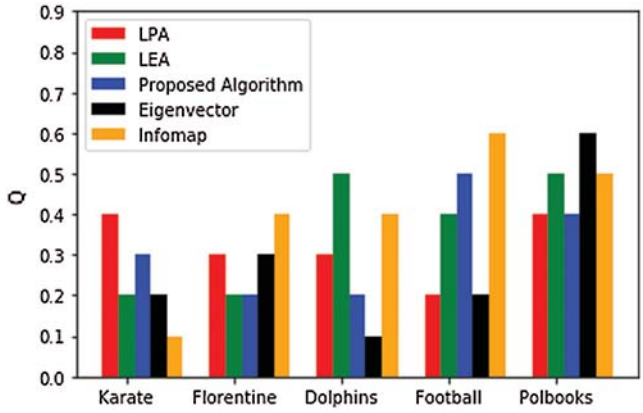

Figure 8: Modularity comparison using real networks

4.3.1 Algorithm Comparison Using Synthetic Networks

We used a standard benchmark synthetic network named LFR. These networks are very popular in the research community owing to various aspects, such as power-law distribution [43], community size, and node degrees. In general, the parameters of LFR networks are average degree (represented by k), size of the network (represented by n), alpha, beta (node degree), the upper bound of the degree (represented by k_{max}), the power-law exponents for the size of communities, and minimum and maximum sizes of communities, *minc* and *maxc*, respectively. Mixing parameter μ represents the probability of nodes connected with nodes of an external community [44]. It is feasible to produce a different type of network by assigning different values to these parameters, $\mu = \frac{\sum c\,|E\,(c,\,V-C)|}{\sum v \in V^{d(v)}}$, where $E\,(c,\,V-C)$ is the edges between C and other nodes except C, and |.| indicates the size function; μ denotes the ratio of edges for 'intra' communities. If the value of μ is higher, it means more ambiguous communities are generated. To measure the performance of

our proposed model, we compared with top leading community detection algorithms such as LPA, eigenvector, and info map, etc. Fig. 9 illustrates the LFR network comparison 1. The panels in Fig. 9 demonstrate a decrease in NMI values in the y-axis from increasing the size of the network, n, and the mixing parameter, μ, in the x-axis. In this experiment, we used 50 nodes, hence, $n = 50$. The average degree is 5, and $kmax\, c = 0.1$, $n = 5$, $max\, c = 0.1$, and $m = min\, c = 5$. The reason is the value of μ. If the value of μ is large, it means that we are getting a more ambiguous community structure. So, it is very difficult to identify the accurate number of communities by increasing the value of μ. It is the very same in these networks. It is not very difficult to understand the behavior of all these algorithms. When we use $\mu \leq 0.3$, these algorithms behave differently and gradually reach a value of 1. If we look clearly at the LPA algorithm, it starts slightly from 0.7 and gradually adjusts with the value $\mu = 0.2$. The behavior of the eigenvector algorithm is quite similar, and the value $\mu = 0.6$ reaches that point. If we carefully look at our proposed model, the attained values of μ are more stable than the prior state-of-the-art algorithms. Fig. 10 illustrates LFR network comparison 2 using $n = 500$ and $\beta = 1$. We examine how the values of n and β affect the performance of these algorithms. We examined the diversity of the results achieved in both cases. The values obtained with our proposed model display many similarities. In particular, when mixing parameter $\mu \geq 0.5$, the LPA algorithm slightly decreases, and this behavior can be seen in this figure. This achieved result indicates that the size of communities does not distinguish the performance of these algorithms. If we increase the value of μ, it results in a more ambiguous structure. When the value of μ increases, it is difficult to get an accurate number of communities. Our third observation is that few algorithms lose their efficiency when the value of μ increases. Moreover, the value of the μ parameter affects the performance of the above-discussed algorithms, and hence, the networks become more ambiguous (as mentioned earlier). Briefly, it is necessary to fix the value of μ to 0.6 or 0.7. These two figures indicate that, at first, they lose their efficiency if the value of μ is increasing. For $n = 1500$, 2500 and 3500, the performance of our proposed algorithm was better than the earlier algorithms.

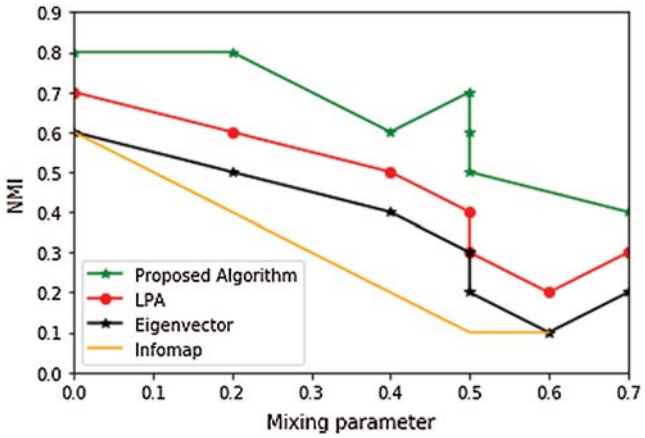

Figure 9: LFR network comparison 1

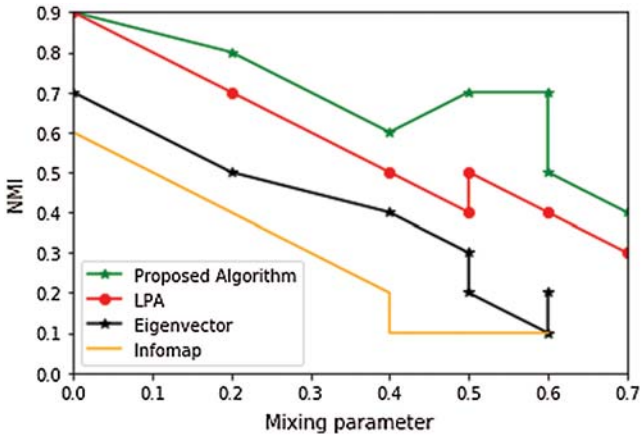

Figure 10: LFR network comparison 2

5 Conclusion

In this study, we proposed a community detection model for social networks. It is a three-step model in which the communities are identified without any prior information. The first step is to identify the most powerful nodes; the second step is to find the neighbors and perform labeling, and the third step is to combine them. We compared our proposed model against the top-5 state-of-the-art models using both real and synthetic benchmarks. In a few of these models, the result was not more efficient, but the achieved result was very close to LPA and the Infomap algorithm. Also, we proved that our proposed model provides more robust and accurate results compared to other models. Therefore, our model offers an alternate method to identify communities in social networks. Generally, community detection is very useful to identify criminal user groups in networks. Our proposed model can be useful in precision marketing and business recommendations. In the future, we have a plan to improve our model for very large networks. In fact, in these networks, the complexity is very high. So, we will consider it for future networks.

Acknowledgement: This research was supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under the Industrial Technology Innovation Program, No. 10063130, and by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1A2C1006159), and by the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center (ITRC) support program (IITP-2019-2016-0-00313) supervised by the Institute for Information & communications Technology Promotion (IITP).

Funding Statement: The authors extend their appreciation to Yeungnam University for funding this work through the researchers' Supporting Project number (NRF-2019R1A2C1006159), Yeungnam University, Gyeongsan, Korea.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Mata, "Complex networks: A mini-review," *Brazilian Journal of Physics*, vol. 50, pp. 658–672, 2020.

[2] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," *Knowledge and Information Systems*, vol. 1, pp. 181–213, 2013.

[3] Y. Su, C. Liu, Y. Niu, F. Cheng and X. Zhang, "A community structure enhancement-based community detection algorithm for complex networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 1, pp. 1–14, 2019.

[4] T. Wang, J. Tan, W. Ding, Y. Zhang and F. Yang, "Intercommunity detection scheme for social Internet of things: Compressive sensing over graphs approach," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4550–4557, 2018.

[5] Z. Lu, J. Wahlström and A. Nehorai, "Community detection in complex networks via clique conductance," *Scientific Reports*, vol. 8, pp. 5982, 2018.

[6] M. Granovetter, "The strength of weak ties," *Journal Storage*, vol. 78, pp. 1360–1380, 1973.

[7] K. Reddy, P. Kitsuregawa, M. Sreekanth and P. S. Rao, "A graph based approach to extract a neighborhood customer community for collaborative filtering," in *Proc. of the Databases in Networked Information Systems*, Berlin, Germany, pp. 188–200, 2002.

[8] A. Hajibagheri, H. Alvari, A. Hamzeh and S. Hashemi, "Community detection in social networks using information diffusion," in *Proc. of the Int. Conf. on Advances in Social Networks Analysis and Mining*, Istanbul, Turkey, pp. 702–703, 2012.

[9] S. Mostafi, F. Khan, A. Chakrabarty, D. Y. Suh and M. J. Piran, "An algorithm for mapping a traffic domain into a complex network: A social internet of things approach," *IEEE Access*, vol. 7, pp. 40925–40940, 2019.

[10] A. Pothen, "Graph partitioning algorithms with applications to scientific computing," in *Handbook of Parallel Numerical Algorithms*. Norfolk, Virginia, USA: Springer, pp. 323–368, 1997.

[11] E. R. Barnes, "An algorithm for partitioning the nodes of a graph," in *Proc. of the IEEE Conf. on Decision and Control including the Symp. on Adaptive Processes*, San Diego, CA, USA, pp. 303–304, 1981.

[12] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *The Bell System Technical Journal*, vol. 49, no. 2, pp. 291–307, 1970.

[13] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, pp. 1–16, 2004.

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

[15] X. You, Y. Ma and Z. Liu, "A three-stage algorithm on community detection in social networks," *Knowledge-Based Systems*, vol. 187, pp. 1–15, 2020.

[16] W. E. Donath and A. J. Hoffman, "Lower bounds for the partitioning of graphs," *IBM Journal of Research and Development*, vol. 17, no. 5, pp. 420–425, 1973.

[17] S. Jianbo and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[18] A. Mahmood and M. Small, "Subspace based network community detection using sparse linear coding," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 3, pp. 801–812, 2016.

[19] 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, pp. 1–12, 2008.

[20] U. N. Raghavan, R. Albert and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review*, vol. 76, pp. 36106, 2007.

[21] M. J. Barber and J. W. Clark, "Detecting network communities by propagating labels under constraints," *Physical Review*, vol. 80, pp. 26129, 2009.

[22] Z. Lin, X. Zheng, N. Xin and D. Chen, "Efficient community detection algorithm based on label propagation with community kernel," *Physica A: Statistical Mechanics and its Applications*, vol. 416, no. 3, pp. 386–399, 2014.

[23] M. G. Puxeddu, M. Petti, F. Pichiorri, F. Cincotti and D. Mattia, "Community detection: comparison among clustering algorithms and application to egg-based brain networks," in *Proc. of the Int. Conf. of the IEEE Engineering in Medicine and Biology Society*, Jeju, korea, pp. 3965–3968, 2017.

[24] M. Rosvall and C. T. Bergstrom, "Maps of information flow reveal community structure in complex networks," *National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[25] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical review*, vol. 74, pp. 36104–36104, 2006.

[26] S. Chen, Z.-Z. Wang, L. Tang, Y.-N. Tang and Y.-Y. Gao, "Global vs local modularity for network community detection," *PLoS ONE*, vol. 13, pp. 1–21, 2018.

[27] F. Amin and G. S. Choi, "Social pal: A combined platform for internet of things and social networks," in *Proc. of the Int. Conf. on Computer and Communication Systems*, Shanghai, China, pp. 786–790, 2020.

[28] F. Amin and G. S. Choi, "Hotspots analysis using cyber-physical-social system for a smart city," *IEEE Access*, vol. 8, pp. 122197–122209, 2020.

[29] A. Alsini, A. Datta and D. Q. Huynh, "On utilizing communities detected from social networks in hashtag recommendation," *IEEE Transactions on Computational Social Systems*, vol. 2, pp. 1–12, 2020.

[30] M. A. Javed, M. S. Younis, S. Latif, J. Qadir and A. Baig, "Community detection in networks: A multidisciplinary review," *Journal of Network and Computer Applications*, vol. 108, no. 6, pp. 87–111, 2018.

[31] H. Jiang, L. Sun, J. Ran, J. Bai and X. Yang, "Community detection based on individual topics and network topology in social networks," *IEEE Access*, vol. 3, pp. 1, 2020.

[32] D. Rhouma and L. B. Romdhane, "An efficient algorithm for community mining with overlap in social networks," *Expert Systems with Applications*, vol. 41, no. 9, pp. 4309–4321, 2014.

[33] S. H. S. Javadi, P. Gharani and S. Khadivi, "Detecting community structure in dynamic social networks using the concept of leadership," in *Handbook of Sustainable Interdependent Networks: From Theory to Application*, New York City, USA: Springer International Publishing, pp. 97–118, 2018.

[34] Y. Ma, Y. Zhao, J. Wang, M. Liu and W. Shen, "Modularity-based incremental label propagation algorithm for community detection," *Applied Sciences*, vol. 10, pp. 4060, 2020.

[35] H. Cherifi, G. Palla, B. K. Szymanski and X. Lu, "On community structure in complex networks: Challenges and opportunities," *Applied Network Science*, vol. 4, no. 1, pp. 117, 2019.

[36] W. Luo, N. Lu, L. Ni and W. Zhu, "Local community detection by the nearest nodes with greater centrality," *Information Sciences*, vol. 517, no. 1, pp. 377–392, 2020.

[37] F. Amin, J.-G. Choi and G. S. Choi, "Community detection based on social influence in large scale networks," in *Proc. of the Web, Artificial Intelligence and Network Applications*, Caserta, Italy, pp. 122–137, 2020.

[38] V. Krebs, Books About US Politics Network Dataset (unpublished). 2003. [Online]. Available: http://www.orgnet.com/.

[39] 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, 2003.

[40] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47, 2002.

[41] A. Lancichinetti, S. Fortunato and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Physical Review*, vol. 78, pp. 46110, 2008.

[42] L. Danon, A. Díaz-Guilera, J. Duch and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 9, pp. 9008, 2005.

[43] V. Guerriero, "Power law distribution: Method of multi-scale inferential statistics," *Journal of Modern Mathematics Frontier*, vol. 1, pp. 21–28, 2012.

[44] K. Berahmand, A. Bouyer and M. Vasighi, "Community detection in complex networks by detecting and expanding core nodes through extended local similarity of nodes," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 4, pp. 1021–1033, 2018.