Int. J. Advance Soft Compu. Appl, Vol. 8, No. 3, December 2016
ISSN 2074-8523

K-indicators Method for Community Detection in Social Networks

Mohammad H. Nadimi-Shahraki[1], Mehrafarin Adami-Dehkordi[1]

[1]Faculty of Computer Engineering,
Najafabad branch, Islamic Azad University,
Najafabad, Isfahan, Iran.
e-mail:nadimi@ieee.org

Abstract

In the last decade, a huge amount of data has been generated by online social networks which include informative knowledge about their users. Obviously, analyzing such big data by ordinary techniques is impossible. Thus, many methods are introduced to analyze this data mostly by detecting communities of these social networks. Many applications and businesses are interested in finding communities. In this article, a new method named k-indicators is proposed to detect the self-descriptive communities by combining users' links and users' attributes. Consequently, by using k-indicators method each node of social network is joined to the most similar indicator to form the communities. Experiment results verify that this proposed method can detect the self-descriptive communities and its accuracy can be equivalent to other well-known algorithms such as Newman-Girvan and k-modes.

Keywords: *Community detection, Influential nodes, Self-descriptive communities, Social networks.*

1 Introduction

A social network is usually represented as a graph, where the nodes are the users or individuals and their relations are represented by edges or links. In addition to the links, nodes are often described by certain attributes which are referred to as the contents of nodes. For example, when it comes to the web pages, online blogs or scientific papers, the contents are usually presented by histograms of keywords. In the network of co-authorship, the contents of nodes can represent the affiliation information of researchers [1]. As online social networks become more popular, their analysis increases among the researchers. One major topic that caused these kinds of networks to be analyzable is the community detection methods. Users can be grouped based on their relations or their common attributes. By dividing the network to communities, more detailed analyzing becomes possible. Previous studies on community detection focus on link analysis. Since those methods are not able to extract semantic of the communities, the hybrid methods become more adoptable. These methods enhance community detection process through two algorithms or two data sources.

Recently, many hybrid approaches have been proposed to discover communities [1-10]. Almost all of these approaches use Bayesian models, and they combine text as the content and links in detecting communities. However, using text as the content has some restrictions to it such as handling context. Handling context as a data source for community detection is a voluminous task and needs algorithms with high computational complexities. Therefore, most researches extract the most frequent words or keywords from the context, and then use these words as the content. These approaches are not appropriate for online social networks, because comments and descriptions on the online social networks usually consist of informal words and phrases which are particularly used in virtual environments. In addition, online social networks have users from all over the world; therefore, users' contexts are written in different languages. Existence of mentioned restrictions has motivated the authors of the article to use users' attributes as one of the data sources of hybrid community detection method. These attributes can be extracted from users' profiles, available in almost all online social networks. These attributes are filled by users, are reliable and at least stay constant for a period of time. These attributes are not extracted from comments that are not clear if the writers are serious about their idea or their comments just relate to a transient emotion of the period.

In this article, a hybrid method for community detection is proposed where two kinds of data sources, links and attributes, are applied named *k*-indicators. By using *k*-indicators each node of social network is joined to the most similar indicator or centroid of all communities' indicators. Therefore, a community is a connected subgraph of nodes, consisting of the users who are the most similar to one another based on both of these sources. Link source contributes to the formation of communities through users' connections, and attributes source at the same time contributes nodes of the community to have similar characteristics too.

The rest of this article is organized in the following way. The related works are reviewed in section 2. Next, the proposed method is described in section 3, where the computational complexity is analyzed as well. Then, the proposed method is experimentally evaluated by conducting synthetic data set and the real data set. Finally, the article is concluded in section 5.

2 Related works

The first generation of community detection methods applies links such as friendship edges or contents of nodes. Some of the popular community approaches are: graph partitioning, devise methods, spectral methods, Bayesian models and clustering [11-16]. More approaches are reviewed in [17-19]. Since there are various traditional types of clustering algorithms such as simple linkage, complete linkage, *k*-means, this approach is applied widely in community detection methods. Two common clustering approaches, namely hierarchical and partitioning approaches, are applied for this purpose. Hierarchical methods detect communities without prior knowledge on the number of communities, while the best level of hierarchy should be selected, which needs a criterion to determine the appropriate level. One of the drawbacks of the hierarchical method is: members will be grouped at each iteration level, but after first assignment given to each member in its community, the member cannot change its community even if it has a greater similarity to other sub clusters [20]. Partitioning methods need the number of communities and initial nodes as inputs. The common criteria that used to find communities are: Normalized cut [21], maximum flow [22], modularity [14], matrix factorization [23] and edge betweeness [23, 24]. Some of these criteria for mining the relationships among nodes in social network analysis are recently proposed, which seems to be more effective for finding communities [25, 26].

Beside the mentioned approaches, the content analyses are applied in special cases such as topic models and citation graphs. For detecting communities a change in each one of the documents into some frequent keywords and the document- word matrix should take place for further analysis [5, 6]. This preprocessing is a costly step in community detection methods. In the topic model studies, each community has one or more topics for the users to write about. Bayesian models are adopted for community detection algorithms; hence, most topic models are vulnerable to words that are irrelevant to the target topic [1]. Moreover, there are a few scalable Bayesian approaches in community detection in graph [10] such as most extend Latent Drichlet Allocation applied in [27, 28]. Citation graphs construct the social network based on references and keywords available in scientific articles. Experimental results indicate that the accuracy of community detection is enhanced through these two major constituents of an article [29].

Neither link information nor content information is sufficient to decide on the member of the community. Combining link and content named hybrid for community detection usually leads better performance. There are some advantages and disadvantages using hybrid methods as follows:

Using other sources: According to [30], communities are users with common topics. Since the document collections are large, the authors here found a solution to make a scalable community detection method by applying text contents as well as relations thereof. Their proposed solution, first, decomposes the data collection into smaller units by exploring the relations in a rapid manner. With respect to the textual attributes, relations among documents consistent topics are conveyed, since they are constructed deliberately by their creators. Handling relations may first reduce the overhead in interpreting and summarizing texts significantly, so that the solution has a high scalability regardless of the data set size. Through a relation topology analysis, a set of preliminary community cores is generated and later expanded into communities.

Combining links and attributes: Attribute data, such as demographic information and correlation data can yield more accurate results than classical algorithms, where either only attributes or only relationships are applied. JointClust algorithm applying these two data sources determine cluster atoms in the first phase, which are then merged in a bottom-up strategy in the second phase. The results of experimental tests indicate that their algorithm indeed indicates meaningful and accurate clustering without requiring the user to specify the number of clusters [7].

Combining different approaches: A Hybrid Community Discovery Framework (HCDF) is a frame work presented by Henderson et al, which applies Latent Dirichlet Allocation on Graphs (LDA-G) as the core Bayesian method for community detection. The key aspect of HCDF is its effectiveness in incorporating hints from a number of other community detection algorithms and generates results that outperform the constituent parts. Furthermore it generates algorithms that can predict links [10].

Complementary information: Complementary information such as nodes' Influence is used to find the centroids of communities. Researchers apply centroids to detect the communities. Khorasgani et al. proposed a new method, that formed communities by influential nodes named leaders and assigned other nodes to these leaders as followers. This complementary information is extracted from link sources while yielding more accurate communities [31]. Huang et al. proposed an unsupervised analysis to detect the best cores, centroids, via all other members. The cores are identified through the weights of relevant relations they have [12]. Liang et al. proposed algorithms to predict some tags. They believed approximately 50% of social networks' lost users leave due to a lack of people to follow. In that study some accounts and related tags are extracted and ranked. The predicted tags with central accounts related to those tags are given to users [32].

The entire mentioned hybrid studies including the above four applied just one data source such as [31] or they used Latent Dirichlet Allocation in their methods [10, 27, 28] which is a voluminous task. There are different languages and many written comments available in online social networks. Moreover, what is common in online social networks is when users make friends with many other users, who really do not communicate with them even once a year; and yet there are some users who are similar in their interests and their attributes but since they do not know one another so they do not have any connection as well with. To solve the above weaknesses of hybrid methods, a new hybrid method named *k*-indicators is proposed.

3 *K*-indicators method

Each one of the communities in the social network consists of nodes; one of these nodes is the centroid or indicator of that community. There exists a similarity among the users and their indicator in their community. In social networks, these community indicators could be the most influential nodes. There exists k communities formed through their indicators; therefore the proposed method in this article is named *k*-indicators. As shown in Fig. 1, the *k*-indicators method consists of three main phases as follows: Phase one provides a new solution in determining the k, indicating the number of communities. Similar to other clustering and community detection algorithms, the k should be properly clarified because its selection has direct influence on accuracy of the results. In phase two, two data sources consisting of users' relations and users' attributes go through fusion. It should be mentioned that these attributes lead to finding the self-descriptive communities; that is, the semantic of communities can be extracted by these attributes. In phase three, algorithms are proposed to detect the self-descriptive communities.

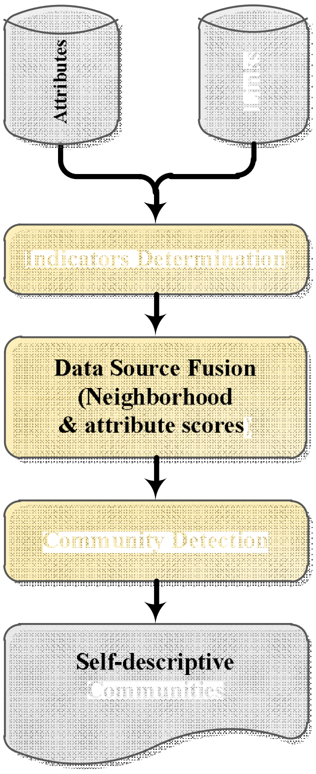


Fig. 1: Main phases of the *k*-indicators method

3.1 Indicators determination

Selection of the initial indicators is important because similar to hill climbing algorithms, usually the initial centroids are restricted to local optimum instead of being restricted in global optimum. Since the appropriate selection of initial indicators have a direct effect on the speed of algorithm convergence and the quality of the results, new search methods [32, 33] and algorithms [12, 34] are recently introduced to find the best indicators of the social networks.

In order to determine k, there are several solutions such as repeating the algorithm by different k numbers and then accepting the k which has the maximum accuracy regarding the detected communities and applying the ground truth of data which is not always available.

In this article the social influence concept is adopted in determining k. To the best of authors knowledge there is no study where this concept is adopted. Change a person's behavior due to the perceived relation with other people, organizations, and society in general is social influence. Social influence is a widely accepted phenomenon in social networks [35]. Node and edge measurements are used to compute the social influence. For choosing proper indicators which would represent the k, only node measurements are used in this study. Node-based centrality is defined in order to measure the importance of a node in the network. A node with high centrality score is usually considered more influential than other nodes in the network. Many centrality measures are proposed based on the precise definition of this influence.

The Katz centrality counts the number of walks starting from a node, while penalizing longer walks. The Katz centrality C_i^{KATZ} of node i is defined by Equation (1).

$$c_i^{KATZ} = e_i^T(\sum_{j=1}^{\infty}(\beta A)^j 1 \qquad (1)$$

where, e_i is the column vector that the ith element is 1, and the rest are 0. β is a positive penalty constant between 0 and 1.

The simplest and most popular measure in Equation (1) is that of the Degree centrality. Let A in Equation (1) be the adjacency matrix of a network, and deg (i) is the degree of node I, then it follows that the Degree centrality c_i^{DEG} of node i is defined to be the degree of the node presented in Equation (2). The Degree centrality is interpreted as: it counts the number of paths of length 1 which starts from a node.

$$C_i^{DEG} = deg(i) \qquad (2)$$

In this study, the centrality of every node is measured and then sorted. The nodes with maximum centrality in sorting function have the potential of being a community indicator that are called candidate indicators. It is obvious that if a community has more than one central node, k will be the upper bound for the number of communities. Accordingly, each candidate central node must be proved as the most influential node in its community; therefore, other nodes would be omitted from the candidate list. Finally, the total number of central nodes determines the k. As mentioned above, the *k*-indicators method detects communities based on two data sources. Accordingly, the central nodes would be in a specific community provided that they have similar attributes and common neighbors. In order to omit some candidate central nodes in this attempt, two thresholds are introduced and applied: α representing the similarity of attributes and β representing the number of shared neighbors. The manner in which the influential nodes are determined expressed in Algorithm 1. In this algorithm, the extracted number of the influential nodes determines k.

Algorithm 1. Finding indicators

INPUT: Candidate indicators, α, β

OUTPUT: indicators, number of indicators

METHOD: similar indicators trough attributes similarity and common neighbors are removed.

```
indicators ←candidate indicators
for all Candidate indicators c do
      CN←compute common neighbors of each 2 Candidate indicators c
      DS←compute dissimilarity of each 2 Candidate indicators c
        if DS < α OR CN>β
            remove one of c from indicators' list
      end if
end for
k← number of real indicators
```

3.2 Data source fusion

The *k*-indicators method is capable to detect hybrid communities through users' attributes (categorical features) and links. The nodes of this graph have their own features which describe their characteristics. These features can relate to the kind of relation among them or not related to the graph. Links can represent any relation among the nodes of the graph or can even represent a separate social network. These attributes and links must go through fusion in order to be applicable in the hybrid community detection algorithm.

Attribute score: Attribute data can be represented as an n×m entity-attribute matrix, where n is the number of users or nodes and m represents the number of attributes. The dissimilarity between two categorical objects are computed by simple mismatching [36]. Simple mismatching between two nodes with d features is presented in Equation (3).

$$D\left(x_i, x_j\right) = \sum_{l=1}^{d} \delta\left(x_{il}, x_{jl}\right)$$

$$\delta\left(x_{il}, x_{jl}\right) = \begin{cases} & x_{il} \neq x_{jl} \\ & x_{il} = x_{jl} \end{cases} \quad (3)$$

Recently Cao et al. introduced a new function shown in Equation (4) to compute the dissimilarity by which the accuracy of *k*-modes clustering algorithm can be enhanced [37].

$$NDis_p\left(z_l, x_i\right) = \sum\nolimits_{a \epsilon P} NDis_a\left(z_l, x_i\right)$$

$$NDis_a\left(z_l, x_i\right) = 1 - Sim_a\left(z_l, x_i\right) \times m_a$$

$$m_a = \frac{\left|x_i \middle| f(x_i, a) \equiv f(z_l, a), x_i \epsilon c_l\right|}{\left|c_l\right|} \quad (4)$$

$$Sim_a(x, y) = \frac{f(x, a) \equiv f(y, a)}{\sum_{z \epsilon U} f(x, a) \equiv f(z, a)},$$

$$f(x, a) \equiv f(y, a) = \begin{cases} & \\ & \end{cases}$$

$$f(x,a) \equiv f(y,a) = score_{i,j}(attributes)$$

$$= 1 - dissimilarity(node_i, indicator_j)$$

Based on the Equation (4), the score of the similarity of each one of the nodes and the available indicators must be calculated trough Equation (5).

$$score_{i,j}(attributes) = 1 - dissimilarity(node_i, indicator_j) \tag{5}$$

Similar to *k*-modes algorithm each node should be assigned to the community or cluster with the most similar indicator.

Neighborhood score: the correlation data is modeled by the adjacency matrix [38] where, each element in the matrix is computed as Equation (6).

$$m_{ij} = 1 \quad iff\ relation\ (x_i, x_j) = True \tag{6}$$

The neighbors of every one of the nodes are compared with all of the available indicators, then that node will be assigned to the community with the maximum neighborhood score with its indicator. The score of common neighbors is computed as presented in Equation (7).

$$score_{i,j}(neighborhood) = \frac{|common\_neighbors(node_i, indicator_j)|}{N} \tag{7}$$

Normalization and fusion of both the scores: The following example reveals why the normalization is necessary.

Example: Let us consider object o with the three i_1, i_2, i_3indicators. Scores of attributes and neighborhood are tabulated in Table 1. It is not clear that object o should be assigned to which community. By min-max normalization process both of the scores will be in the same range of values; hence a rational summation could be applied. Each score v is normalized through min-max normalization[39], Equation (8). In this study the new minimum and maximum values are zero and one, hence all the normalized scores will be between zero and one.

$$v' = \frac{v - min_A}{max_A - min_A}\left(new_{max_A} - new_{min_A}\right) + new_{min_A} \qquad (8)$$

The normalized values are presented in the second row of Table 1. At this point the similarity of object o, which is 1, with the second community becomes clear.

Table 1: An artificial Data sources

indicators	i_1		i_2		i_3	
parameters	Similarity	Common neighbors	Similarity	Common neighbors	Similarity	Common neighbors
Real scores	2	10	0	25	1	6
Normalized scores	0	0.78	1	1	0.5	1

Now that the score is normalized, Algorithm 2 will be adopted for fusion of the scores.

Algorithm 2. Compute_Total_Score

```
INPUT: node x, community C, indicator(C)
OUTPUT: normalized total score of node x
METHOD: Total score is computed through summation of normalized two scores of attribute and neighborhood.
for all nodes m ϵ C do
        influence (m) ← Compute neighborhood_Score (m, indicator(C))
    similarity (m) ←Compute attribute_Score (m, indicator(C))
    Find min, max of both scores
end for
//normalization
Ninfluence(x) = [influence (x) – min (influence)] / [max (influence) – min (influence)]
Nsimilarity(x) = [similarity (x) – min (similarity)] / [max (similarity) - min (similarity)]
Total_score(x) = Nsimilarity(x) + Ninfluence(x)
```

3.3 *K*-indicators algorithm

The *k*-indicators method apply k representative nodes or indicators, where each node in the data set will be assigned to one of the k indicators with the maximum score with respect to that indicator. First, communities are formed by attribute source. After this step, an iterative process where the communities are refined and the indicators are updated, begins until the indicators are stabilized. The basic idea of this method is inspired by *k*-modes clustering algorithm. The significant difference between this and *k*-modes algorithm is: here the two attributes and links go through a fusion process. The major steps of *k*-indicators are highlighted in Algorithm 3.

The major steps in this algorithm include forming the communities and updating the indicators. The first assignments of the nodes to their indicators are based on similarity between nodes and indicators; while, for the iterative process both the similarity and common neighbor scores are considered. Common neighbors of each node and an indicator should be computed. If the node and the indicator have neighbors in other communities, these foreign members should be ignored. The two sources, attributes and links, make updating step more complex in comparison with algorithms such as *k*-modes [36] where one source is applied.

Algorithm 3. *k*-indicators

INPUT: data set (attributes, adjacency matrix), *k*

OUTPUT: detected communities

METHOD: communities are detected trough their indicators by an iterative process.

```
indicators selection through Algorithm 1
// first assignments
for all nodes n ϵ data set do
    assign n to the community which has minimum dissimilarity with its real indicator   and      make first communities
end for
for all communities do
    Update the indicator
end for
//  iterative section
Repeat
    for all nodes n ϵ data set do
```

```
        assign n to the community which has maximum total score (Algorithm2) with its indicator
    end for
    for all communities do
        Update the indicator
    end for
until there is no change in communities' indicators
```

3.4 Indicators updating

An indicator in a community is the most similar node to all of the other nodes in its own community. Each community has the choice to select its mode as its indicator. Mode of a cluster or a community is a vector of attributes that can either be a specific node or not an available node; therefore, mode is not selected as the indicator. The node with the maximum influence in the community can be another choice as an appropriate indicator. The influence of node can be measured by Degree centrality of the best and the simplest measurements [35], through Equation (9).

$$DC(n) = \frac{deg(n, c)}{N - 1} \tag{9}$$

The updating process of each indicator in a community is expressed through Algorithm 4.

Algorithm 4. Update the indicator

INPUT: community C

OUTPUT: The most proper indicator of community C

METHOD: the node with maximum score is select to be C’s indicator for iterative section of Algorithm 3.

```
new_mode ← mode(C)
for all nodes $c \epsilon C$ do
    influence (c) ← Compute influence (c)
    similarity (c) ←Compute similarity (c, new_mode)
    Find min, max of both scores
end for
```

```
//normalization
for all nodes c ϵ C do
    Ninfluence(c) = [influence (c) – min (influence)] / [max (influence) - min (influence)]
    Nsimilarity(c) = [similarity (c) – min (similarity)] / [max (similarity) - min (similarity)]
    score(c)=Nsimilarity (c)+Ninfluence (c)
end for
indicator(C) ← node c with maximum score
```

3.5 Computational Complexity Analysis

In referring to the pseudo-code of the above algorithms, the computational complexity of *k*-indicators method is as follows: The major computational steps consist of community formations in iterative section and updating indicators. With respect to the total score computed by Algorithm 2, the computational complexity for assigning the nodes to their related communities is $o(nk(p+|C|))$, where n is the number of nodes, p is the number of attributes and |C| is the number of nodes in its own community. The iteration time assumed as t, and the whole computational cost of Algorithm 3 is $o(t(n(p+|C|)+k|C|^2)$. The computational complexity of updating all the indicators is $|C|p+|C|(|C|+p)+|C|$. Since t and p can be considered as constants and |C| is equal to n in the worst case, the computational complexity of the method is equal to $O(n^2)$.

4 Experimental Evaluation

These proposed algorithms are coded in Matlab 7.10.0 programming language. Other related results and illustrations are computed by NodeXL 10.0.1.229. The accuracy of the *k*-indicators method is evaluated through the Facebook social network and Soybean diseases network.

4.1 Facebook Social network

In this section some Facebook users' profiles are gathered as a data set.

4.1.1 Data set Preparation based on Facebook users' profiles

Facebook is one of the most popular online social networks. Users in this website have their own profiles. Consistently, to evaluate the proposed method, a data set is made from profile of users in which there are three users as seeds. Two of seeds are born and reside in Isfahan city, located in Iran, they are female and male bachelors and aged under 30. The third user is an Iranian woman who has migrated to U.A.E., and then she is divorced and is about 45 years old.

These users have 600 friends in total generating 2362 inter-friendship relations. The adjacencies of these users are stored as a matrix. As shown in Fig. 2, social networks have sparse links among their users.

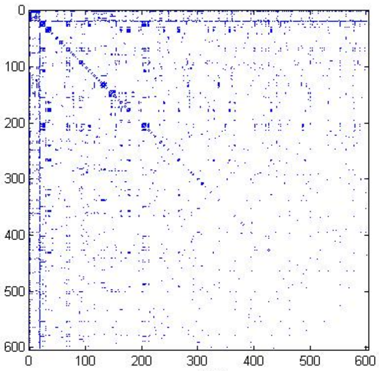


Fig. 2: Scarcity pattern of Facebook adjacency matrix

The attributes of the statistical population are tabulated in Table 2. Regarding the attribute religious their profiles are blank; hence, this attributes is ignored. In each profile there are eight textual fields that, in which users can write whatever they want about their interests. Here two attributes from these textual fields: the writing language and the number of filled fields are extracted.

In order to register in Facebook, responding to some attributes is optional while to others is mandatory. The optional fields can have a null value. With regard to the privacy policy of Facebook, some attributes are not exposed; thus, they are null too. Appling both the link and attribute sources can decrease the effect of null values. The image of the link source is illustrated in Fig. 3. Almost 30% of the nodes have just one friend in this data set. For this data set, the number of communities is set to 3 and initial indicators are the seeds.

Table 2: Attributes and domains of Facebook data set

Attribute	Categorical Domain	Some examples
Gender	(1,2)	Female, male
Locale	(1,8)	fr_Fr ,fa_Ir ,en_US ,en_UK
Marital status	(1,11)	Single, married, complicated
Cities, states, Counties	(1,145)	Tehran, Dubai, Isfahan
Language	(1,3)	English, Persian, others

Number of filled textual fields	(1,8)	1 to 8
Religion	-	-

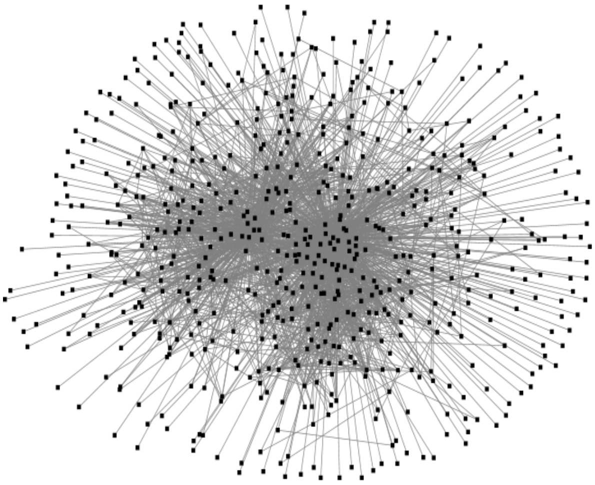


Fig. 3: Facebook social network, proposed the statistical population

4.1.2 Determining the indicators

Some statistics about the population of Facebook social network data set is tabulated in Table 3. Based on the average network diameter, the Degree centrality is applied to measure the nodes influence.

Table 3: The Facebook social Network properties

Property name	Maximum value	Average value
Network diameter	5	3
Node's degree	318	8
Node's degree(without outlier nodes)	100	9.47

The nodes degree distribution is showed in

Fig. 4. There are just two nodes in influential set 10, The node degree value of which is more than 100; therefore these nodes can assumed as outliers. The degree values of the set 9 which consists of four nodes range from 50 to 100. This set is

selected as the first candidate nodes; where maximum number of communities would be 4. The set 8 has 22 nodes and is the last selected set.

In order to determine indicators from the above mentioned two sets, the α, β of Algorithm 1 are applied.

α selection: To select a proper number of common neighbors, nodes distribution is applied. In this data set, almost 30% of the nodes have just one friend. Moreover, nodes with more than 50 friends constituted just 1% of all the nodes. Here, these two groups of nodes are considered as outliers; and nodes with 2-50 friends are involved in determining the proper α.

β selection: The data set attributes are: gender, locale, marital status, the city, state and country of hometown and current location of users, language and number of filled textual fields. 6 attributes of these 11 attributes are assigned to users' location. This information involved in determining the proper β.

Table 4 indicates the values obtained for α, β. Moreover, it shows value 3 as representative number of the communities which are detected by applying Algorithm 1. To prove the extraction of 3 the two algorithms applied in [24] and [40] are used (Fig. 5). For the next experimental tests, the fixed values of 10 for α and 4 for β are applied.

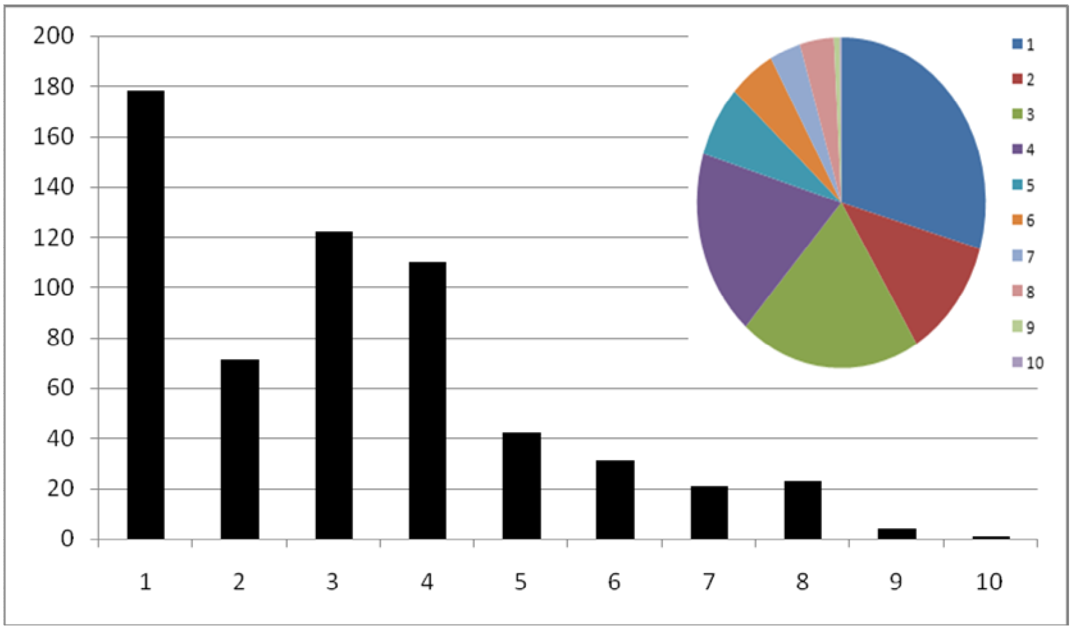
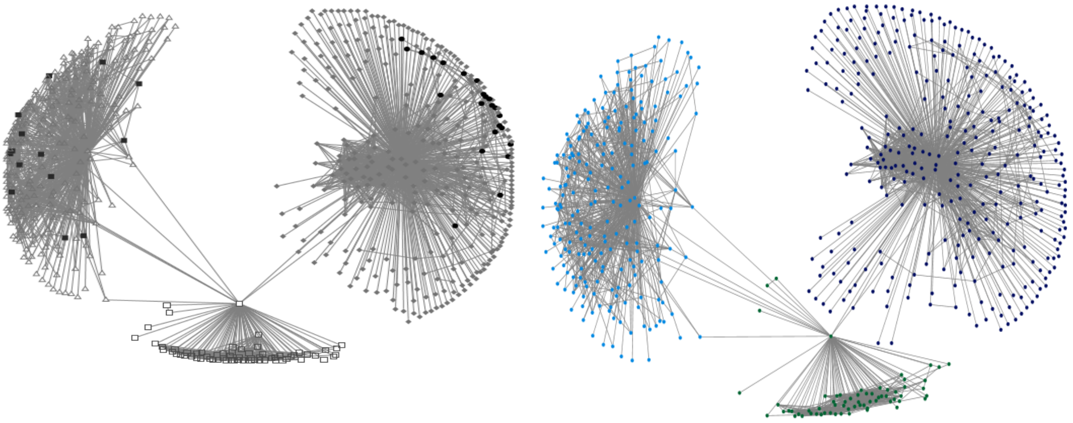


Fig. 4: Nodes degree distribution



(a)

(b)

Fig. 5: Illustrations of detected communities. (a) Detected communities by Clauset algorithm, (b) Detected communities by Newman-Girvan algorithm

Table 4: Proper α and β selection

Node degree ranges	α	β	Number of communities
(51,100)	(10,40)	(1,4)	3
(26,50)	10	(5,6)	3
(26,50)	(15,19)	(3,4)	5

4.1.3 Detecting the communities

In order to detect communities, three approaches of applying the sources are tested, the results of which are presented in Table 5.

As observed in Table 5, selection of the sources yields different members for communities. The By applying attribute source, *k*-indicators method detects self-descriptive communities. Indicators of each community can describe the most frequent values for each attribute. Here, the description of most of the detected communities' users is: the first and second members of communities are live in their hometown cities and are single. The members of third community are female who are born in Iran, live in U.A.E, and they refuse to respond to their marital status (

Table 6).

Table 5: Different sources yield different community memberships

Data sources	1st community	2nd community	3rd community
Links	318	216	69
Attributes	260	201	142
Hybrid sources	282	123	198

Table 6: Description of communities' indicators

	Hometown location			Current location			Gender	relation	locale	language	Filled interests
	City	State	Country	City	State	Country					
1st Community	Isfahan	Isfahan	Iran	Isfahan	Isfahan	Iran	female	single	fa_IR	unknown	0
2nd Community	Isfahan	Isfahan	Iran	Isfahan	Isfahan	Iran	Male	Single	fa_IR	Persian	2
3rd Community	Shiraz	Fars	Iran	Dubai	Dubai	U.A.E	Female	null	en_US	Persian	5

4.1.4 Accuracy evaluation

The accuracy of the *k*-indicators method is evaluated through Equation (10) where a_i is the number of nodes that are correctly assigned to their community, and k is considered equal to 3.

$$Accuracy = \frac{\sum_{i=1}^{k} a_i}{n} \tag{10}$$

The *k*-indicators method detects communities through two sources of links and attributes. Since there does not exist an algorithm to compare the results obtained here with, it is necessary to compare the accuracy of this proposed method with the accuracy of other well-known algorithms that either use links or categorical attributes. Here, in the link-only case, the Newman-Girvan's algorithm [24] is applied.

The number of members in each community is expressed in Table 7. Each one of the three seeds, applied for data set preparation in section 4.1.1, has many friends, nine of which are common in each ones' 'friendship list by the one; therefore, they are considered as a member for the community which has a greater degree in relation to the members of communities.

Table 7: Number of Facebook communities' members, and the accuracy of method

	1st community	2nd community	3rd community	Accuracy	Computational Complexity
Newman-Girvan	318	220	65	$\frac{601}{603} = 0.996$	)
k-indicators	318	216	69	$\frac{600}{603} = 0.995$	$O(n^2)$
Friends of seeds	320	219	73	-	-

4.2 Soybean diseases network

In the second experiment set, the accuracy of proposed method is experimentally evaluated by conducted benchmark data set Soybean. The results reported were computed by the average of multiple runs.

4.2.1 Data set Preparation based on Soybean diseases network

Soybean disease data set is frequently applied for categorical clustering algorithms. This data set has 47 instances, each being described by 35 attributes. Each instance is labeled as one of the four diseases. Except for the fourth disease which has 17 instances, all other diseases have 10 instances each. To make this data set as a social network, links are added to instances with a common disease. By adding this kind of relations to categorical attributes, communities can be found. Fig. 6 illustrates that there are four communities in this data set, same as the number of diseases. For this data set, the number of communities is fixed to 4 and two groups of initial nodes are applied.

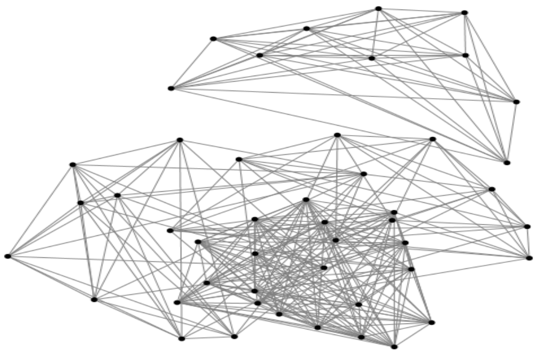


Fig. 6: Soybean network

4.2.2 Accuracy evaluation

In this study, since the extracted information of the two applied sources is consistent, the accuracy of *k*-indicators can be evaluated by comparing detected communities with instances which have the same label of disease. The accuracy of Soybean diseases network is tested and the results are expressed in Table 8 (a), (b).

Table 8: Accuracy vs. link source completeness

Number of links	First indicator selection method	Accuracy	Table section
0	Randomly	0.357	a
61	Randomly	0.6	
271	Randomly	0.71	
271	Specified nodes	0.893	b

As observed in Table 8 section (a), the first three tests are random in selecting the first indicators where the number of links is not constant, leading to difference in values of accuracy. It is found here that an increase in the links would enhance accuracy. In section (b), the selection of first indicators is not random. Here, each indicator is an instance of one of the four kinds of diseases. This test clears that indicators determination has effect on the quality of results.

5 Conclusion

In this article, a new hybrid method named *k*-indicators is proposed for community detection. This method is capable of determining the most influential nodes as the indicators of communities which declaring the number of communities. Two data sources of links and attributes are applied here in order to detect online social networks where the data in each source is incomplete and inconsistent. Fusion of sources can increase the quality of results through reducing the negative effect of incomplete inconsistent data. Moreover, by applying the attributes from users' profiles, the most frequent values of each community describe what is common among them. The experimental evaluations indicate that the accuracy of the *k*-indicators method is comparable with Newman-Girwan algorithm’s which detects communities based on links as a source. In addition, the accuracy gained by *k*-modes algorithm is improved by the use of links as an additional source within data source of categorical features.

References

[1] T. Yang, R. Jin, Y. Chi, and S. Zhu, "Combining link and content for community detection: a discriminative approach," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 927-936.

[2] K. Yang, "Combining Text-and Link-based Retrieval Methods for Web IR," in *TREC*, 2001.

[3] Y. Wang and M. Kitsuregawa, "On combining link and contents information for web page clustering," in *Database and expert systems applications*, 2002, pp. 902-913.

[4] J. Li and O. R. Zaïane, "Combining usage, content, and structure data to improve web site recommendation," in *E-Commerce and Web Technologies*, ed: Springer, 2004, pp. 305-315.

[5] S. Zhu, K. Yu, Y. Chi, and Y. Gong, "Combining content and link for classification using matrix factorization," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 487-494.

[6] N. F. Chikhi, B. Rothenburger, and N. Aussenac-Gilles, "Combining link and content information for scientific topics discovery," in *Tools with Artificial Intelligence, 2008. ICTAI'08. 20th IEEE International Conference on*, 2008, pp. 211-214.

[7] F. Moser, R. Ge, and M. Ester, "Joint cluster analysis of attribute and relationship data withouta-priori specification of the number of clusters," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2007, pp. 510-519.

[8] C. Wang, Z.-y. Guan, C. Chen, J.-j. Bu, J.-f. Wang, and H.-z. Lin, "On-line topical importance estimation: an effective focused crawling algorithm combining link and content analysis," *Journal of Zhejiang University SCIENCE A*, vol. 10, pp. 1114-1124, 2009.

[9] Y.-M. Li and C.-W. Chen, "A synthetical approach for blog recommendation: Combining trust, social relation, and semantic analysis," *Expert Systems with Applications*, vol. 36, pp. 6536-6547, 2009.

[10]K. Henderson, T. Eliassi-Rad, S. Papadimitriou, and C. Faloutsos, "HCDF: A Hybrid Community Discovery Framework," in *SDM*, 2010, pp. 754-765.

[11]A. Borg, M. Boldt, N. Lavesson, U. Melander, and V. Boeva, "Detecting serial residential burglaries using clustering," *Expert Systems with Applications*, vol. 41, pp. 5252-5266, 2014.

[12]H. Huang, Y. Gao, K. Chiew, Q. He, and B. Zheng, "Unsupervised analysis of top-k core members in poly-relational networks," *Expert Systems with Applications,* vol. 41, pp. 5689-5701, 2014.

[13]B. Kernighan, Lin, S, " An efficient heuristic procedure for partitioning graphs," *Bell system technical journal,* vol. 49, pp. 291-307, 1970.

[14]M. E. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences,* vol. 103, pp. 8577-8582, 2006.

[15]H.-W. Shen and X.-Q. Cheng, "Spectral methods for the detection of network community structure: a comparative analysis," *Journal of Statistical Mechanics: Theory and Experiment,* vol. 2010, p. P10020, 2010.

[16]D. Zhou, E. Manavoglu, J. Li, C. L. Giles, and H. Zha, "Probabilistic models for discovering e-communities," in *Proceedings of the 15th international conference on World Wide Web,* 2006, pp. 173-182.

[17]C. C. Aggarwal, *An introduction to social network data analytics*: Springer, 2011.

[18]S. Fortunato, "Community detection in graphs," *Physics Reports,* vol. 486, pp. 75-174, 2010.

[19]S. Parthasarathy, Y. Ruan, and V. Satuluri, "Community discovery in social networks: Applications, methods and emerging trends," in *Social Network Data Analytics,* ed: Springer, 2011, pp. 79-113.

[20]R. Xu and D. Wunsch, *Clustering* vol. 10: Wiley. com, 2008.

[21]J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, pp. 888-905, 2000.

[22]S. M. van Dongen, "Graph clustering by flow simulation," 2000.

[23]F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding, "Community discovery using nonnegative matrix factorization," *Data Mining and Knowledge Discovery,* vol. 22, pp. 493-521, 2011.

[24]M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences,* vol. 99, pp. 7821-7826, 2002.

[25]H. Zare, A. Mohammadpour, and P. Moradi, "A random projection approach for estimation of the betweenness centrality measure," *Intelligent Data Analysis,* vol. 17, pp. 217-231, 2013.

[26]G. Li, Z. Pan, and B. Xiao, "Community discovery and importance analysis in social network," *Intelligent Data Analysis,* vol. 18, pp. 495-510, 2014.

[27]T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 50-57.

[28]S.Lacoste-Julien, F. Sha, and M. I. Jordan, "DiscLDA: Discriminative learning for dimensionality reduction and classification," in *Advances in neural information processing systems*, 2008, pp. 897-904.

[29]D.Greene and P. Cunningham, "Multi-view clustering for mining heterogeneous social network data," presented at the 31st European Conference on Information Retrieval 2009.

[30]H. Li, Z. Nie, W.-C. Lee, L. Giles, and J.-R. Wen, "Scalable community discovery on textual data with relations," in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008, pp. 1203-1212.

[31]R. R. Khorasgani, J. Chen, and O. R. Zaïane, "Top leaders community detection approach in information networks," in *Proceedings of the 4th Workshop on Social Network Mining and Analysis,* 2010.

[32]B. Liang, Y. Liu, M. Zhang, S. Ma, L. Ru, and K. Zhang, "Searching for people to follow in social networks," *Expert Systems with Applications,* vol. 41, pp. 7455-7465, 2014.

[33]M. Zhang, H. Hu, Z. He, and W. Wang, "Top-k similarity search in heterogeneous information networks with x-star network schema," *Expert Systems with Applications,* vol. 42, pp. 699-712, 2015.

[34]D. Rhouma and L. B. Romdhane, "An efficient algorithm for community mining with overlap in social networks," *Expert Systems with Applications,* vol. 41, pp. 4309-4321, 2014.

[35]J. Sun and J. Tang, "A survey of models and algorithms for social influence analysis," in *Social Network Data Analytics*, ed: Springer, 2011, pp. 177-214.

[36]Z. Huang, "Extensions to the *k*-means algorithm for clustering large data sets with categorical values," *Data Mining and Knowledge Discovery,* vol. 2, pp. 283-304, 1998.

[37]F. Cao, J. Liang, D. Li, L. Bai, and C. Dang, "A dissimilarity measure for the *k*-Modes clustering algorithm," *Knowledge-Based Systems,* vol. 26, pp. 120-127, 2012.

[38]S. Zhou, A. Zhou, W. Jin, Y. Fan, and W. Qian, "FDBSCAN: a fast DBSCAN algorithm," *RUAN JIAN XUE BAO,* vol. 11, pp. 735-744, 2000.

[39]J. Han, M. Kamber, and J. Pei, *Data mining: concepts and techniques*: Morgan kaufmann, 2006.

[40]A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," *Physical review E,* vol. 70, p. 066111, 2004.