Azerbaijan Journal of High Performance Computing, Vol 5, Issue 1, 2022, pp. 72-86
https://doi.org/10.32010/26166127.2022.5.1.72.86

A Model for Multi-Layer Dynamic Social Networks to Discover Influential Groups Based on a Combination of Meta-Heuristic Algorithm and C-Means Clustering

Lida Naderlou[1] and Zahra Tayyebi Qasabeh[2]

[1] Non-Profit Higher Education Institutions Roozbeh Zanjan Branch, Zanjan, Iran. lidanaderlou@gmail.com
[2] Payame Noor University of Guilan, Guilan, Iran, tayyebi.shiva@gmail.com

*Correspondence:
Zahra Tayyebi Qasabeh,
Payame Noor University of
Guilan, Guilan, Iran, tayyebi.shiva@gmail.com

Abstract

Science and technology are proliferating, and complex networks have become a necessity in our daily life, so separating people from complex networks built on the fundamental needs of human life is almost impossible. This research presented a multi-layer dynamic social networks model to discover influential groups based on a developing frog-leaping algorithm and C-means clustering. We collected the data in the first step. Then, we conducted data cleansing and normalization to identify influential individuals and groups using the optimal data by forming a decision matrix. Hence, we used the matrix to identify and cluster (based on phase clustering) and determined each group's importance. The frog-leaping algorithm was used to improve the identification of influence parameters, which led to improvement in node's importance, to discover influential individuals and groups in social networks, In the measurement and simulation of clustering section, the proposed method was contrasted against the K-means method, and its equilibrium value in cluster selection resulted from 5. The proposed method presented a more genuine improvement compared to the other methods. However, measuring precision indicators for the proposed method had a 3.3 improvement compared to similar methods and a 3.8 improvement compared to the M-ALCD primary method.

Keyword: Multi-layer Dynamic Social Networks, Influential Groups, Meta-Heuristic Algorithm, C-means Clustering.

1. Introduction

A social network is a social structure formed by a group of social activities, and ties are created among these activists. Social network creates many methods to analyze the structure of social features with patterns that are produced based on the theory of each social network (Dasgupta, S., & Prakash, C., 2016, March).

This research examines a more complex social network called a multi-layered social network. Recently, the concept of multi-layer networks has emerged from the field of

1. Introduction

A social network is a social structure formed by a group of social activities, and ties are created among these activists. Social network creates many methods to analyze the structure of social features with patterns that are produced based on the theory of each social network (Dasgupta, S., & Prakash, C., 2016, March).

This research examines a more complex social network called a multi-layered social network. Recently, the concept of multi-layer networks has emerged from the field of complex networks under the domain of complex systems, and it is a fertile ground for new visualization research. In big data, simple and multi-layered social networks can be found everywhere and in any field. Estimating the degree of importance of each node in this network is not the same, and giving weight to the nodes is necessary to control the network. For this purpose, the relationship between the characteristics of the nodes and the relationship with the network structure should be investigated. Reliability, control, and power should be considered to find the degree of each node in the system function (Yang, Y., & Xie, G., 2016)

It is these days. Many methods have been proposed to identify essential nodes (influencers) in simple social networks. In contrast to the results of recent research in identifying effective distributors in multi-layered dynamic social networks, it has provoked a broad debate and is a hot topic. The main issue and a correct solution to identify essential nodes in multi-layer networks are to consider a parameter for that node. Identifying the influential (important) node can be done using evolutionary algorithms inspired by nature. The main goal is to solve the problem and maximize users' priority in social networks (Zhou, J., Zhang, Y., & Cheng, J., 2014). Therefore, in this research, the combination of evolutionary algorithms and data mining is used to calculate the user preferences and their influence on the network, and as a result, their maximization and detection, and it answers the question of how to provide a model for social networks. Is a dynamic layer to discover influential groups based on the combination of meta-heuristic algorithm and C-means clustering possible?

2. Related Work

In 2022, Noori et al. proposed a new Method for Detecting Influential Nodes in Social Network Graphs Using Deep Learning Techniques. They stated that one of the most critical issues in large social networks is identifying influential users to maximize the diffusion of news and messages, popularly known as the Social Influence Maximization Problem (SIM Problem). The success of the diffusion process in these networks depends on the effective user selection mechanism. On the other hand, with the increase in growth rate and data size in the graph of large social networks, one of the main challenges is a large number of nodes and edges, which makes any processing on it face many problems. Implementing traditional methods on large graphs with high-dimensional data is difficult and time-consuming, and more efficient methods must be used. In this paper, they propose a new method to reduce the graph size of social networks using deep learning and then try to provide a new and effective solution to the Social Influence Maximization Problem by considering the minimum overlap between nodes. In the following, simulation results in the real world show better performance of the proposed method in terms of execution time and Influence spread than traditional techniques (Noori, A., 2022)

In 2022, Wang et al. proposed a model for Influential node identification using

In 2022, Wang et al. proposed a model for Influential node identification using network local structural properties. They stated that with the rapid development of information technology, the scale of complex networks is increasing, making spreading diseases and rumors harder to control. Identifying the influential nodes effectively and accurately is critical to predicting and controlling the network system pertly. Some existing influential node detection algorithms do not consider the impact of edges, resulting in the algorithm effect deviating from the expected. Some consider the global structure of the network to result in high computational complexity. To solve the above problems, based on the information entropy theory, they propose an influential nodes evaluation algorithm based on the entropy and the weight distribution of the edges connecting it to calculate the difference of edge weights and the influence of edge weights on neighbor nodes. They select eight real-world networks to verify the effectiveness and accuracy of the algorithm. They verify the infection size of each node and top-10 nodes according to the ranking results by the SIR model. Otherwise, the Kendall τ coefficient is used to examine the consistency of our algorithm with the SIR model. Based on the above experiments, the performance of the LENC algorithm is verified (Wang, B., Zhang, J., Dai, J., & Sheng, J., 2022).

In 2021, Qian et al. proposed a model for Data-Driven Influential Nodes Identification in Dynamic Social Networks. They started The identification of influential nodes in social networks has significant commercial and academic value in advertising, information management, and user behavior analysis. Previous work only studies the network's simple topology without considering the network's dynamic propagation characteristics, which does not fit the actual scene and hinders wide application. To solve the problem, We develop a data-driven model for identifying influential nodes in dynamic social networks. Firstly, we introduce an influence evaluation metric BTRank based on user interaction behavior and topic relevance of the information. We construct a multi-scale comprehensive metric system by combining BTRank, LH-index, and betweenness centrality. Secondly, we use simulation data to train a regression model and obtain the metric weights by Gradient Descent Algorithm to optimize the metric weights calculated by the entropy weight method. Thirdly, the weights obtained from training are used in weighted TOPSIS to sort the influence of nodes and identify influential nodes among them. Finally, We compare our model with existing models on four real-world networks. The experimental results have demonstrated significant improvement in accuracy and effectiveness achieved by our proposed model (Qian, Y., & Pan, L., 2021, October)

In 2020, Jalil et al.; proposed An incremental approach to updating influential nodes in dynamic social networks. They stated that Detecting the influential nodes in dynamic social networks is a recent field that has gained considerable interest from researchers. One interesting approach is to update influential nodes incrementally, considering the social network's structural evolution. However, most existing methods can only be used to identify influential nodes in static rather than dynamic social networks. We propose an incremental approach for detecting influential nodes by inspecting social network evolution to solve this problem. First, we identify the influential nodes in the original network. Then, we propose a method for finding the changed elements. Finally, we present our algorithm for updating influential nodes in dynamic social networks. Experimental results on three real dynamic social networks prove that our approach achieves better performance in terms of both influence degree and computational time (Hafiene, N., Karoui, W., & Romdhane, L. B., 2020).

In 2019, Li et al. proposed a multi-layer network community detection model based on attributes and social interaction intensity. They stated that With the rapid development of mobile communications and electronic technology, relevant network systems have become more prominent in scale and exhibited different hierarchical relationships. The complex relationships and links among different users ultimately complicate obtaining data on a whole system. Even with the availability of powerful aggregation tools, most companies cannot afford the associated human and financial costs. However, existing multi-layer network community detection methods are well suited for attribute-based community detection. Therefore, this paper proposes a multi-layer local community detection model based on attribute and structure information. This model can effectively utilize node attribute information and the similarity strength information revealed by social exchanges to improve the accuracy of community detection in multi-layer networks. Unlike classical multi-layer and global community detection algorithms, this algorithm is robust on most datasets because of its modularity and computational efficiency (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

In 2019, Lei et al. a model for Overlapping communities' detection of social networks based on a hybrid C-means clustering algorithm. They stated that community detection has become increasingly important in social network analysis as an essential part of social computing. Overlapping communities detection, one of the important topics, is beneficial to understanding the properties of knowledge-sharing organizations in social networks. Because of uncertainties inherent in knowledge-sharing organizations, good results are hard to gain by using traditional community detection technologies. Through a complement of both fuzzy sets and rough sets, this paper proposed a novel hybrid clustering method, which uses a fuzzy partitioning technique to replace a traversal search method for discovering overlapping community structures. The final representation leads to an efficient description of overlapping regions among communities and uncertainties in class boundaries.

Meanwhile, considering local and global structural features of knowledge-sharing organizations in complex networks, a meaningful similarity measure is designed for each pair of objects. As a result, our proposed method can effectively and efficiently detect communities whose boundaries are not easily separated from each other. Further, experimental results on complex synthetic and real-world networks demonstrate that the proposed method works well in detecting overlapping community structures in a knowledge-sharing organization of complex networks (Lei, Y., Zhou, Y., & Shi, J., 2019).

In 2019, Mittal et al. a model for Classifying influential individuals in multi-layer social networks. They stated that Nowadays, social media is one of the popular modes of interaction and information diffusion. It is commonly found that some entities do the primary source of information diffusion, and such entities are also called influencers. An influencer is an entity or individual who can influence others because of their relationship or connection with their audience. This article proposes a methodology to classify influencers from multi-layer social networks. The proposed methodology is a fusion of machine learning techniques (SVM, neural networks) with centrality measures. A multi-layer social network is the same as a single-layer social network, which includes multiple properties of a node and is modeled into multiple layers. We demonstrate the proposed algorithm on real-life networks to validate the approach's effectiveness in multi-layer systems (Mittal, R., & Bhatia, M. P. S., 2019).

3. The proposed method

In this Section, for an entire multi-layer network with scattered connections, many disturbances, and significant noises, the characteristics of all network nodes and the strength of social similarity were jointly considered. This method proposes the first local community detection model based on multi-layer characteristics and community structure (AM-ALCD). This model, which is a development of regional community diagnosis, relies on a new measure of the intensity of community similarity based on the community structure similarity. A different criterion is defined to describe the characteristics of the group and the strength of social similarity. With the help of the effective use of this new criterion, an algorithm is proposed for the detection of local communities. The proposed method determines the node similarity using specific real-life node characteristics. The frog jump algorithm and C-Means clustering are used to obtain the integrated similarity level of the connection between the nodes. In the proposed method, first, the multi-layer network graph model is described. Then the AM-ALCD model is proposed to detect the local community in a multi-layer network. Finally, the framework of the algorithm based on input seed nodes is presented.

Attribute similarity between nodes in the same layer: The nodes in a social network are typically associated with a wide variety of attributes, such as age, gender, place of residence, shopping preferences, and behavioral descriptions. Each of these attributes can be regarded as forming one attribute layer. The attributes of the core nodes in each layer are critically crucial to describing the node features. Therefore, in addition to a structure-based social similarity, the similarity between nodes in the same layer, which arises from the attribute-based similarity, should also be considered. Attribute-based node classification can be achieved by classifying nodes with the same attribute into the same category. During local community detection in a network layer, the attribute similarity between a node and its neighbors in the corresponding structure is used to define the criterion for selecting a seed node. The set of all seeds fulfilling this criterion is determined, which are then expanded to establish the communities (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

Let $G_{\mathcal{L}} = (V_{\mathcal{L}}.E_{\mathcal{L}}.V.\mathcal{L})$ denote the attribute-based network model, where VL denotes the nodes in layer L, with V belonging to the set $\{V_1.V_2.\ldots\ V_n$and n denotes the number of nodes in the network, and EL denotes the edges connecting the nodes in layer L. Let $S_{\mathcal{L}} = (S_1.S_2.\ldots.S_n)$ denote the node attributes, where SL denotes the set of attributes in layer L, and S1 denotes the attribute of node V1. Consider a network with L layers; each layer has n attributes, which N corresponding keywords {can represent$X_1.X_2.\ldots.X_n\}$. Then, the set of attributes for node Vi can be denoted by a vector $S_i = (W_{i1}.W_{i2}.\ldots.W_{in})$. The element Wir is 1 if the keyword Xr appears in node Vi; otherwise, it is 0. Although each node in a community has various attributes, not all of these attributes are prevalent. Therefore, when updating the attributes of a node Vi, when two or more attributes occur most frequently among its neighbors, the attribute similarity between node Vi and each corresponding neighboring subsystem is computed as the basis for label selection. The attribute set of a neighboring subsystem can be represented in terms of the attributes of all nodes within that subsystem. For node Vi in layer X, the attribute set of its neighboring subsystem Cim can be computed as follows (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019):

$$S_x C_{im} \frac{1}{n} (\textstyle\sum_{j=1}^{n} W_{j1} . \sum_{j=1}^{n} W_{j2} . \dots . \sum_{j=1}^{n} W_{jn}) \tag{1}$$

Where N denotes the number of neighbors in Cim. The attribute similarity $S_{imX}(S_i . S_{cim})$between node Vi and its neighbor set C_{im} in layer X can be calculated as the cosine of the included angle between the two corresponding vectors (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019; Interdonato, R., Tagarelli, A., Ienco, D., Sallaberry, A., & Poncelet, P., 2017):

$$S_{xim}(S_i . S_{Cim}) = \frac{\sum_{j=1}^{N} s_{ij} \times s_{Cim}}{\sqrt{\left[\sum_{j=1}^{N} s_{ij}\right]^2 \left[\sum_{j=1}^{N} s_{Cimj}\right]}} \tag{2}$$

Here, Sij and Scim denote the jth elements of the attribute vectors of Vi and Cim, respectively, in layer X, that is, Wij and $\frac{1}{n}\sum_{k=1}^{n} W_{kj}$. A higher value of $S_{xim}(S_i . S_{Cim})$ indicates a greater level of attribute similarity between the nodes and a more significant possibility of the node belonging to its neighboring subsystem Cim. If two or more attributes occur most frequently, the attributes of the node are used to compute the similarity between the node and its neighboring subsystems. The attribute of the neighboring subsystem for which $S_{xim}(S_i . S_{Cim})$ is maximized and selected as the seed node's attribute, V0 (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

Strength of social interaction between nodes in the same layer: The information-based local community clustering algorithm does not need the number of communities in the network in advance, nor does it need the locations of nodes of interest. All the algorithm needs are each local node's information to cluster communities rapidly. However, for a local community detection method, a maximum number of communities is usually pre-defined as the condition for terminating the algorithm. This pre-set number considerably influences the clustering performance (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

In this Section, the level of similarity in terms of social information is used to determine the final node added to a community during community expansion. Community expansion is terminated when the level of similarity between the current node and the core node of the neighboring subsystem with the same attribute is less than the level of similarity between the current node and the core node of a neighboring subsystem with a different attribute (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

$$S_x tr(S_i . S_{Cim}) = \frac{\sum_{i \in \Gamma i \cap \Gamma cim}^{N} s_{ij} \times s_{Cim}}{\sqrt{\sum_{i \in \Gamma i} \frac{1}{D(t)}} \sqrt{\sum_{i \in \Gamma cim} \frac{1}{D(t)}}} \tag{3}$$

Here, $S_x tr(S_i . S_{Cim})$ denotes the level of similarity between node i and its neighboring subsystem in layer X; Γi and Γcim denote the set of nodes to which node i points and the set of nodes pointing to subsystem Cim, respectively; and D(t) denotes the node degree (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

Levels of attribute similarity and social interaction similarity in a multi-layer network: The discussions above focus on defining the levels of node attribute similarity and social interaction similarity in a single-layer network. However, human relationships in social networks are multi-faceted, and social networks have multiple layers. Here, we consider the multiple layers of a social network and study how to measure the attribute and social interaction similarities within a multi-layer network. The attributes of a set of neighbors can be represented in terms of the attributes of all nodes in the system. For node Vi, let C_{xyim} denote a set of its neighbors in a multi-layer network; then, the set of attributes of C_{xyim} can be computed as follows (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019):

$$S_{xycim} = \frac{1}{n}(\sum_{ix.iy \in \mathcal{L}} \sum_{i.j=1} W_{i.j2} \ldots . \sum_{ix.iy \in \mathcal{L}} \sum_{i.j=1} W_{i.jn} \tag{4}$$

Where N denotes the number of nodes in the set C_{xyim} and XX denotes the similarity between node Vi in layer X and the set of its neighbors in layer Y. The attribute similarity between node Vi in layer X and node Ji in layer Y can be measured as the cosine of the included angle between the two corresponding vectors (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019)

$$S_{xy}im(S_i.S_{Cim}) = \frac{\sum_{j=1}^{N} s_{ij} \sum_{ix.iy \in \mathcal{L}} s_{Cim}}{\sqrt{[\sum_{j=1}^{N} s_{ij}]^2 [\sum_{ix.iy \in \mathcal{L}}^{N} s_{Cim} j]}} \tag{5}$$

Social interaction similarity between nodes in different layers: In this Section, the social interaction similarity in a multi-layer network is used to determine the final node added to a community during community expansion. That is, the community expansion process terminates when the level of similarity between the current node and the core node of the neighboring subsystem with the same attribute is less than the level of similarity between the current node and the core node of a neighboring subsystem with a different attribute (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

$$S_{xy}tr(S_i.S_{Cim}) = \frac{\sum_{ix.iy \in \mathcal{L}} \sum_{ix.iy \in \mathcal{L}} \frac{1}{D(t)}}{\sqrt{\sum_{t \in \Gamma i} \frac{1}{D(t)}} \sqrt{\sum_{ix.iy \in \mathcal{L}} \sum_{t \in \Gamma cim} \frac{1}{D(t)}}} \tag{6}$$

Here, $S_{xy}tr(S_i.S_{Cim})$ denotes the level of similarity between node i in layer X and its neighboring subsystems in all other layers; Γi and Γcim denote the set of nodes to which node i points and the set of nodes pointing to subsystem Cxyim, respectively; and D(t) denotes the node degree (Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W., 2019).

Frog Leaping Algorithm and Improvement of Affective Features in Finding Similarity: Memetic algorithms are a particular category of meta-heuristic search methods that result from matching models of natural systems. With its flexibility and power, the frog leaping algorithm has been used to determine the impact and importance of the feature. The Frog Leaping Algorithm (SFLA) is also a global search algorithm that mimics the evolutionary behavior patterns of a group of frogs when they seek to find a location with the most considerable amount of food. The optimization results prove that the SFLA algorithm performs much better than modern meta-heuristic and conventional methods and the obtained results show that this algorithm is more optimal than other algorithms in load distribution (Huynh, T. H., 2008, April).

Network clustering: Fuzzy C-Means is used for this operation. FCM is one of the most common clustering algorithms. FCM clustering is constructed based on the definition of cluster centers through iterative adjustment of their locations and minimization of the objective function as the K-means (KM) algorithm (Bezdek, J. C., Ehrlich, R., & Full, W., 1984).

The FCM algorithm divides L-dimension targets into fuzzy sets. N represents the number of goals, and C represents the number of sets. The objective function is minimized by repeatedly updating the membership of targets and cluster centers to cluster the data. The objective function is as follows:

$$J(U.V) = \sum_{i=1}^{N} \sum_{c=1}^{C} u_{ic}^{m} d^{2}(x_i.v_c) \tag{7}$$

The degree of Uic membership is subject to the following restrictions:

$$\sum_{c=1}^{C} u_{ic} = 1 . \forall\, i \tag{8}$$

Therefore, the formulas for calculating the degree of membership and cluster center are as follows:

$$u_{ic} = \frac{1}{\sum_{c=1}^{C} \left(\frac{d(x_i.v_c)}{d(x_i.v_c)} \right)^{m-1}} \tag{9}$$

$$v_c = \frac{\sum_{i=1}^{N} u_{ic}^{m} x_i}{\sum_{i=1}^{N} u_{ic}^{m}} \tag{10}$$

The performance of the FCM method is based on the K-Means method, one of the clustering methods. Two main goals are followed in this algorithm, the first goal is to obtain a node as the center of the clusters, which is the average value of the nodes belonging to each cluster, and the second is to assign each node to a cluster so that the node with the shortest distance to its center have the cluster. Consider the n-dimensional problem space; the nodes in this space can be defined as D={xi|i=1,2,...,n} in K-means, each cluster with a single node with the same center or value. Cluster averages are displayed. The set C={cj|j=1,2,...,k} represents the center of clusters. A vector called M is also considered to store the cluster number assigned to each node, where each mi is the cluster number for the data Xi. In the k-means algorithm, the default criterion for measuring data similarity is the Euclidean distance, and the algorithm seeks to minimize the sum of the powers of two Euclidean distances between each Xi and Cj attributed to it (Bezdek, J. C., Ehrlich, R., & Full, W., 1984). The following function is considered the target function.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| X_i^{(j)} - c_j \right\|^2 \tag{11}$$

where $\left\| X_i^{(j)} - c_j \right\|^2$ is the distance criterion between the points and cj, the center of the jth cluster.

4. Results

In this Section, the proposed method has been tested, and the results of different stages of evaluation have been analyzed. The previous chapter used Matlab and Weka software to simulate the presentation method. To evaluate the proposed method, the wiki4HE dataset located in This collection has 913 examples, and in each example, different personal and social characteristics of a social network express the extent of Wikipedia usage. It is mainly located in it along with individual features of interest, sharing, usefulness, effectiveness, image, interaction, and relevance. There are 53 features of the mentioned categories in this data set.

In the AM-ALCD method, the frog jump algorithm is used to improve similarity discovery and help final clustering in extracting the impact of nodes and the proximity of influential factors to each other. Weighting these factors and choosing the appropriate parameters to specify the most critical parameters, reducing the time of conclusion, and increasing the quality are used. The considered parameters for this algorithm are shown in Table 1.

Table 1. Parameters used in the frog jump algorithm

Value	Description	Parameter Name	Row
equal to the number of features	The number of decision variables	nVar	1
-10	The lower limit of variable values	VarMin	2
10	The upper limit of variable values	VarMax	3
300	Maximum iteration of the algorithm	MaxIt	4
50	The number of frogs	nPop	5
10	Memeplex size	nPopMemeplex	6
5	Number of Memeplex	nMemeplex	7
3	Alpha value	alpha	8
5	Beta value	beta	9
2	Sigma value	sigma	10

The fixed values used in the algorithm have been obtained from repeated repetitions and have shown the best output with these values. At the beginning of the algorithm, the best cost value is high, and the algorithm tries to minimize its tendency toward zero. This event is approached as the number of iterations progresses. Of course, in evolutionary and optimization algorithms such as the frog leaping algorithm, choosing the maximum number of repetitions is very important because a small number may result in an unfavorable cost, and a large number may increase the response time. Therefore, choosing the correct value is very important, and usually, This value is obtained by repeating the execution and observing the result. The algorithm has been implemented with different iterations in the proposed method to achieve the desired results. The results are shown in Table 2.

In the frog jump algorithm, there are three different cost functions, Rosenbrock, Ackley, and depending on the problem; there are three different cost functions to choose from. According to the three mentioned functions, there is the calculation of the best cost of the algorithm, which executes the algorithm with the same conditions and the number of repetitions of 300 with all three functions. As shown in Table 2, the results in different executions and the value of the best cost are decreasing. This reduction continued until repetition 500, and the increase in the number of repetitions was 100 units in each step. This decrease in value was slight in rounds 400 and 500 and increased after that. For this reason, the value of 500 has been used as the maximum number of repetitions of the frog leaping algorithm in the proposed method. The results of its best marginal cost are shown in Figure 1.

To evaluate the clustering of the AM-ALCD method presented at the beginning, using the introduced data set, we run FCM on it. Data collection and preprocessing

Table 2. Comparison of the value of the best cost in different implementations of the frog leaping algorithm

best cost	Iteration
13.3753	100
12.868	200
12.7251000000000	300
12.2449	400
11.9684	500
12.0736	600
12.0022	700
12.2364	800

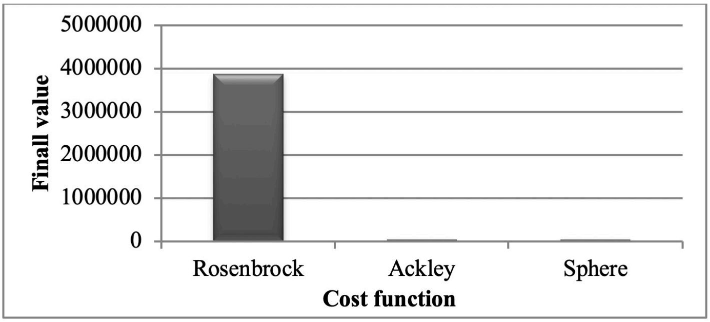

Fig 1. Comparison of the functions that can be used in the frog leaping algorithm in terms of the value of the best final cost

steps have been applied to the dataset. We compare the results of the proposed method with the K-means algorithm. As shown in Figure 3, the Rosenbrock function showed the worst value, and the Ackley function showed the best value at the end of the execution and after 300 iterations. As shown in Table 3, the K-means method has an imbalance in the number of K equal to 2, and cluster one has 67% of the members; as the value increases, this trend becomes more balanced. In the number 5, this value reaches an acceptable relative balance.

Table 3. Results of the K-means method and the proposed method

Cluster number	K=2	nC=2	K=3	nC=3	K=4	nC=4	K=5	nC=5	K=6	nC=6	K=7	nC=7
	Percentage of members		Percentage of members		Percentage of members		Percentage of members		Percentage of members		Percentage of members	
0	67	61	45	41	33	30	28	28	22	25	20	21
1	33	39	27	31	22	26	21	21	18	19	18	18
2			27	28	25	26	20	20	22	21	19	17
3					20	19	17	17	19	17	16	15
4							14	14	8	9	8	9

5									11	10	11	11
6											9	9

As shown in Table 3, the proposed method has results close to the K-means method; therefore, these results can be accepted as acceptable. Also, the amount of balance in the number of 5 clusters in both methods on the proposed data set can be The title suggested the number of clusters. In the FCM implemented in the proposed method, values are considered for the crucial parameters such that for the parameter m and theta value of 2, the maximum repetition is equal to 100, and the term threshold limit is equal to 0.00001. In the following, the clusters obtained from the number equal to 5 were compared with different classification methods to obtain the strength and improvement of the proposed method compared to the known and widely used methods. The following criteria were used for evaluation.

True negative(TN): This criterion means that the node is ineffective and the model predicts the same; the proposed model correctly classifies a normal node.

True Positive (TP): This criterion means that the node is effective, and the model also predicts the same result and is correctly classified by the proposed model.

False positive(FP): This criterion means that the node is not influential, but the model predicts that the node is influential and wrongly classified by the proposed model (also called type 1 error).

False negative(FN): This criterion means that the node is influential, but the model predicts that the node is not influential, and it is wrongly classified by the proposed model (this is also called type 2 error).

Accuracy is the most used and maybe the first choice for evaluating an algorithm's performance in classification problems. It can be defined as the ratio of accurately classified data items to the total number of observations (formula (12)). Despite the widespread usability, accuracy is not the most appropriate performance metric in some situations, especially when target variable classes in the dataset are unbalanced (Vakili, M., Ghamsari, M., & Rezaei, M., 2020).

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN) \tag{12}$$

Recall: It presents "what number of relevant data items are selected." In fact, out of the positive observations, how many of them have been predicted by the algorithm? According to formula (13), the recall equals the number of true positives divided by the sum of true positives and false negatives (Vakili, M., Ghamsari, M., & Rezaei, M., 2020):

$$\text{Recall} = TP/(TP + FN) \tag{13}$$

Precision simply shows "what number of selected data items are relevant." In other words, out of the observations that an algorithm has predicted to be positive, how many of them are positive? According to formula (14), the precision equals the number of true positives divided by the sum of true positives and false positives (Vakili, M., Ghamsari, M., & Rezaei, M., 2020):

$$\text{Precision} = TP/(TP + FP) \tag{14}$$

F1-score: This metric, also known as f-score or f-measure, considers precision and recall to calculate an algorithm's performance. Mathematically, it is the harmonic mean of precision and recall formulated as follows (Vakili, M., Ghamsari, M., & Rezaei, M., 2020):

$$\mathrm{F-measure} = \frac{2PR}{P+R} \quad (15)$$

Criterion and ROC curve: ROC analysis is used to determine the ability to detect test power, compare different test techniques, and determine the appropriate positive threshold. The area calculated by ROC analysis is one of the most important analytical methods used to evaluate the performance of classification algorithms.

J48, SMO, algorithm based on Bayes theory is used to evaluate K-nearest neighbor classification algorithms. An accuracy test using the K-Fold method with K=10 was used in all experiments. In this type of validation, the data is divided into K subsets. From these K subsets, each time, one is used for validation, and another K-1 is used for training. This procedure is repeated K times, and all data are used precisely once for training and once for validation. Finally, the average result of these K validation times is chosen as a final estimate. Table 4 shows the results of all the mentioned criteria for testing the proposed method compared to other algorithms.

Table 4. The results of the test criteria of the proposed method compared to other algorithms

	KNN=3	KNN=5	J48	NaiveBayes	SMO	AM-ALCD
Accuracy	67.908	69.989	52.793	74.2607	84.4469	88.7
TP Rate	0.679	0.7	0.528	0.743	0.844	0.887
FP Rate	0.098	0.91	0.133	0.073	0.044	0.775
Precision	0.696	0.71	0.522	0.742	0.845	0.754
Recall	0.679	0.7	0.528	0.743	0.844	0.887
F-Measure	0.679	0.7	0.522	0.741	0.844	0.736
ROCArea	0.902	0.09	0.867	0. 927	0.956	0.965

As shown in Figure 2, the AM-ALCD (suggested) has better results than other methods. In the F-Measure criterion, the AM-ALCD method has a lower value than the SMO and Bayesian methods. The reason for this is the high value of both the TP and FP, which means that according to the proposed method, it correctly detects both the correct and the incorrect This has caused the reduction of this measure because this measure is obtained from the combination of Precision and Recall. However, the SMO method has a shallow FP value, which has caused an increase in the F-Measure. Also, comparing the critical criterion of accuracy shows that the AM-ALCD method has higher accuracy than all the compared methods. After the proposed method, with a difference of 3.3%, the SMO method is placed, and Bayesian is the following method in this order. KNN methods with K equal to 5 and K equal to 3 are also in the following ranks. Since the AM-ALCD method has a higher classification accuracy, it has more effective group detection and good weighting than other methods. It is worth mentioning that the comparison of the proposed method and others is shown in the diagram related to the ROC criterion.

As it is apparent in the figure, the proposed method in the ROC criterion is closer to the value of one than the others and is higher than all the compared methods. The closest method is the proposed method of SMO and NaiveBayes, among the best classifiers introduced in Table 4 for evaluation.

Finally, the final evaluation was compared with the primary method (Li, X., Xu, G.,

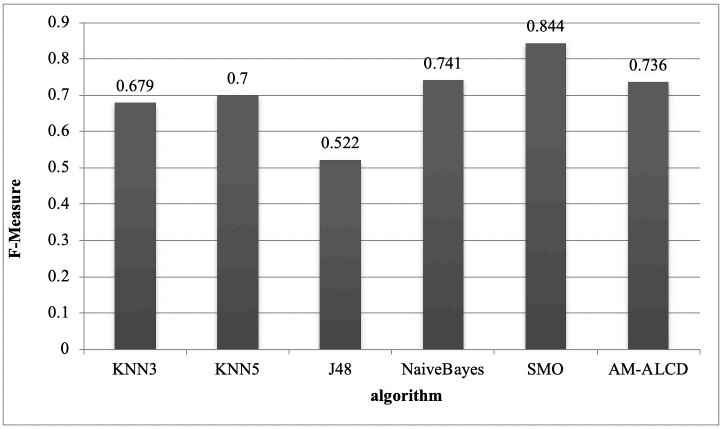

Fig2. Results of ROC criteria of the proposed method compared to other algorithms

Jiao, L., Zhou, Y., & Yu, W., 2019), shown in Figure 7. As shown in Figure 3, based on the equalization of the input parameters and evaluation based on the previous simulations, the proposed AM-ALCD method has recorded an improvement of 3.8% compared to the primary M-ALCD method.

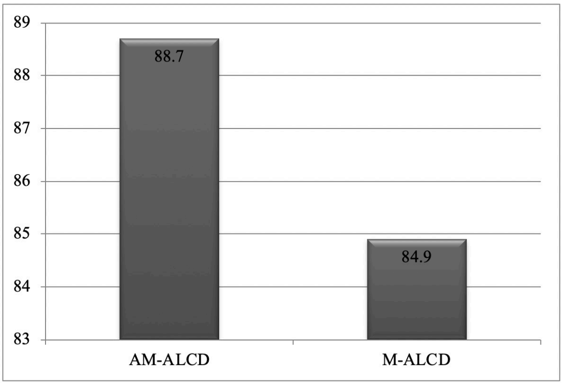

Fig 3. Comparison of the proposed method and the primary method in terms of accuracy criteria

5. Conclusions

In this article, a multi-layer local community detection model is proposed, which is based on structure and feature information. This model can exploit the information on the characteristics of the nodes and the information of the similarity strength that is revealed by the social exchanges and thereby improve the accuracy of community detection in multi-layer networks. Due to its modularity and computational efficiency, this algorithm works powerfully on most datasets, unlike the classic multi-layer and global community detection algorithms. This method has been used to reach influential people and groups in social networks, using the jumping frog algorithm to improve the detection of influential parameters. Based on the proposed method, after collecting the data, we clean and normalize the data, and with the desired data, we try to identify influential people and groups. For this purpose, a decision matrix is formed first and based on it, identification and clustering are made, and the groups' importance is also considered. Also, the importance of nodes is improved by the jumping frog algorithm. The presented method is based on fuzzy clustering. After describing the proposed method, simulation and evaluation were done, the necessary parameters were measured in the algorithms, and the best ones were found by simulation. In the clustering evaluation, the proposed method was compared with the K-means method, and the value of 5 clusters for the proposed method was presented as the value at which the method reached equilibrium. Then the obtained clusters were evaluated and checked with different methods, in which the proposed method showed improvement compared to the compared methods. The evaluation carried out in different criteria has shown the improvement of the proposed method, so that in the accuracy criterion of the proposed method, compared to the decision support vector method, which has the closest accuracy value to the proposed method, it has shown an improvement of 3.3. Also, the AM-ALCD method, compared to The primary M-ALCD method, has recorded an improvement of 3.8.

For improvement and future works regarding the proposed method, it is possible to mention the full implementation of this method in the real world so that the shortcomings of the proposed method are discovered and resolved in the real world. It is also possible to use evolutionary and exploratory methods other than the evolutionary algorithm of the jumping frog to show the best value in the number of clusters or use hybrid algorithms in classification and clustering.

Reference

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & geosciences, 10(2-3), 191-203.*

Dasgupta, S., & Prakash, C. (2016, March). Intelligent detection of influential nodes in networks. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (pp. 2626-2628).* IEEE.

Hafiene, N., Karoui, W., & Romdhane, L. B. (2020). An incremental approach to update influential nodes in dynamic social networks. *Procedia Computer Science, 176,* 781-790.

Huynh, T. H. (2008, April). A modified shuffled frog leaping algorithm for optimal tuning of multivariable PID controllers. *In 2008 IEEE International Conference on Industrial Technology (pp. 1-6)*. IEEE.

Interdonato, R., Tagarelli, A., Ienco, D., Sallaberry, A., & Poncelet, P. (2017). Local community detection in multilayer networks. Data Mining and Knowledge Discovery, 31(5), 1444-1479.

Lei, Y., Zhou, Y., & Shi, J. (2019). Overlapping communities detection of social network based on hybrid C-means clustering algorithm. *Sustainable Cities and Society, 47,* 101436.

Li, X., Xu, G., Jiao, L., Zhou, Y., & Yu, W. (2019). Multi-layer network community detection model based on attributes and social interaction intensity. *Computers & Electrical Engineering, 77,* 300-313.

Mittal, R., & Bhatia, M. P. S. (2019). Classifying the influential individuals in multi-layer social netw*orks. International Journal of Electroni*cs, Communications, and Measurement Engineering (IJECME), 8(1), 21-32.

Noori, A. (2022). A New Method for Detecting Influential Nodes in Socia*l Network Graphs Using Deep Learning* Techniques. Karafan Quarterly Scientific Journal.

Qian, Y., & Pan, L. (2021, October). Data-Driven Influential Nodes Id*entification in Dynamic Social Networks. In International Conference on Collaborative Computi*ng: Networking, Applications and Worksharing (pp. 592-607). Springer, Cham.

Vakili, M., Ghamsari, M., & Rezaei, M. (2020). Performance analysis *and comparison of machine and deep learn*ing algorithms for IoT data classification. arXiv preprint arXiv:2001.09636.

Wang, B., Zhang, J., Dai, J., & Sheng, J. (2022). Influential nod*es identification using network local structural properties.* Scientific Reports, 12(1), 1-13.

Yang, Y., & Xie, G. (2016). Efficient identification of node importance in s*ocial networks. Information* Processing & Management, 52(5), 911-922.

Zhou, J., Zhang, Y., & Cheng, J. (2014). Preference-based mining of top-K influential nodes in social networks. Future Genera*tion Computer Systems, 31, 40-47.*

Submitted: 18.02.2022
Accepted: 18.05.2022