Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL ALGORITHM BASED ON THE NODE TIGHTNESS DEGREE FOR COMMUNITY DETECTION IN LARGE GRAPHS

WAFAA OUBAALLA[1], LAILA BENHLIMA[2]

[1]Mohammadia School of Engineers, Mohammed V University, Rabat, Morocco.
[2] Mohammadia School of Engineers, Mohammed V University, Rabat, Morocco.
E-mail: [1]oubaalla.wafaa@gmail.com, [2]benhlima@emi.ac.ma

ABSTRACT

Community detection is one of the most important research topics in the complex network area. The identification of community structure in large graphs analyze the information unrevealed in the exterior data relationships, explore the internal structure and the function of networks and improve their efficacity. A lot of approaches and methods have been proposed to identify communities based on network structure. However, the majority of them focus on topologies of nodes but ignore the relevance of interactions between them. In this paper, we propose a novel algorithm especially focused on identifying the initial communities then expanding them by using a new node tightness degree based on the edge clustering coefficient and the shared neighbour's similarity of nodes. The proposed approach is evaluated based on different small and large datasets corresponding to different contexts. The experiments prove good results in terms of modularity and computation time while using the new node tightness degree

Keywords: *Graph, Node, Edge, Community, Modularity*

1. INTRODUCTION

In the last decade, the complex system network has become the main form of representing data relations and analyzing data information in diverse fields. The network in the real world mostly represents a high level of organization. Consequently, the processing of such real networks that can reach millions of nodes becomes a challenging task. Much research has proved that graph theory is a very helpful tool to model this type of network, by considering the nodes as elements and the edges as the relationships between them. One of the most important challenges when studying graphs is the identification of community structure [1]. Community detection in complex networks has attracted the attention of several researchers due to its efficiency and reliability in different fields and different types of networks such as social networks, biological networks, geographical networks... Community detection can be defined as a significant and logical partitioning of a set of nodes (subgraphs). This partitioning is based on function quality and network topology, in the objective of understanding the structure of such networks, revealing information, etc. community structure is the most widely studied structural feature of complex networks. It identifies the dense groups of nodes that have a higher density of edges inside them and a lower density of edges between them. Recently, the problem of community detection in graphs has attracted a big attention due to the great availability of the data sets of the large networks. Discovering the community structure in these networks has attracted much attention in recent years. Due to its growing utilization, it becomes one of the most studied subjects in complex networks analysis. furthermore, the objective of discovering communities is detecting the partition of a graph whose nodes are strongly connected compared to the nodes of the rest of the graph. The objective of community detection is to identify a significant organization of the graph and to identify the particular relations between the nodes which can help to understand the function and the structure of the networks. Moreover communities allows to have a mesoscopic view of the complex networks and helps to understand its structure. It also helps to carry out more complex operations on the networks like visualization, compression, or parallelization.

For example, the communities in friendship networks represent the groups of friends. In the World Wide Web network, the communities represent the web pages having the same subjects.

Many approaches have been proposed to detect communities in large graphs, but they are not satisfactory considering the computation time and quality of partitions.

Nowadays, the modularity optimization algorithms are the only approach able to detect communities in large graphs. However, it is demonstrated in [2, 3] that the resolution limit of these methods weakens its performance. In this work, a novel approach is proposed to solve this problem, this approach employs a new metric based on the edge clustering coefficient and the shared neighbours' similarity to optimize the quality of the constructed partitions. This new metric is used to extract the initial communities, then extend them to have the final communities. In the following sections we explain in detail the concept of the proposed community detection method. The approach is assessed using the modularity and the computation time. The experiments show good results for small and large graphs.

The remaining sections of this article are structured as follows: Section 2 presents some related works of community detection. Section 3 is devoted to a presentation of the problem statement, the related concepts and the proposed metric. Section 4 details the proposed algorithm. Section 5 deals with the evaluation of our proposed approach, and the presentation of the obtained results. Section 6 presents some conclusions and perspectives for future work.

2. RELATED WORKS

The identification of community structure in graphs allows us to understand the network performance. In the last years, various methods have been proposed through a lot of research and experiments to undertake the topic of community detection in large graphs. The Girvan-Newman algorithm [4] uses a global information parameter called the edge betweenness that gives the number of the shortest path between all pairs of nodes, with O(n3) as time complexity. This method has initiated the field of community detection due to its great success. Although, it has a high computational complexity. So that it does not apply to large graphs. Afterward, Rachidi Algorithm [5] was proposed. Based on the edge clustering coefficient It has upgraded the Girvan-Newman algorithm by reducing computational complexity. But it also has a high computational complexity because it seeks the edge with the minimum edge clustering coefficient in the global area.

Kongwen Li and Qing Gu [6] proposed a local community detection algorithm based on the edge clustering coefficient. It begins from a node and agglomerates the local community based on the node clustering coefficient. Then it merges the initial local community to get the final community structure. However, this algorithm does not specify the way of selecting the initial node, and the size of the local community is smaller. These factors influence the accuracy of merging community in later periods.

Another famous approach introduced by Clauset et al. [7] based on the modularity maximization process applied to hierarchical agglomerative methods [8]. This method considers each node as a community, then the communities that give the higher increase in the modularity value step by step until it remains just a single community containing all nodes. This method is one of the best community detection methods because of its low time complexity O(n.log2)(n).

Blondel et al. [9] proposed an algorithm to discover communities in large graphs based on two steps: In the first one, it begins from a set of given nodes and merge them using greedy optimization to achieve a maximum modularity. In the second step, it considers each obtained community as a single node and release the global maximum value of modularity. Then, it iterates the first step. This algorithm is very fast and hands a big value of modularity. However, however, it requires a large storage space.

The InfoMap [10] is one of the most important methods of community detection based on the information theory tools. It uses the random walk technique as an intermediate for the information flow and then it seeks the best partition by using the compression of information. This algorithm has O(m) as time complexity.

The Random walk algorithm proposed by [11] based on the hierarchical agglomerative approach it detects the communities through the distance between the nodes. Moreover, the distance between nodes belonging to the same community is very short. Nevertheless, this method has a large time complexity which makes it difficult to apply in the large graphs.

The Graph partitioning [12] is another way to detect communities by dividing the graph into predefined number of clusters, such that the number of edges in a cluster is higher than the number of edges between them.

The Partitional clustering approach [13, 14, 15] divides a dataset into a fixed number of disjoint groups. The purpose is to partition the graph into clusters in order to minimize the cost function based

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

on dissimilarity measure between nodes. This approach requires a predefined number of clusters of the network and it has a high complexity when fixing the level of the cut of the dendrogram, also they suffer from the high computation time.

The Spectral clustering methods based on the eigenvectors of matrices to divide the nodes in the graph using the pairwise similarity between these nodes [13].

Divisive algorithms separate communities having low similarity to each other [16], by removing the inter cluster edges in a network. The main instances of this type include Radicchi et al. method [17] based on the edge clustering coefficient to remove edges iteratively.

Another class of community detection methods is the evolutionary algorithms, they are characterized by their effective local learning and global research capabilities, we distinguish two categories in this class; the first one is based on single objective optimization [18] MLAMA-Net [19], MLCD [20], etc. The second one is a multi-objective optimization that include COMBO [21], I-NSGAII [22].

3. PROBLEM FORMULATION

Using different techniques of graph theory, many real-world networks can be modelled as a graph G (V, E), where V (nodes) denotes the set of elements in the graph and E (edges) represents interactions between them. Our objective in this work is to identify the communities C = {c1, c2, …, ck} in the graph G, namely, the set of partitions where the internal connections are denser than the other connections between them. In the following sections, we detail the used metrics and the new proposed function

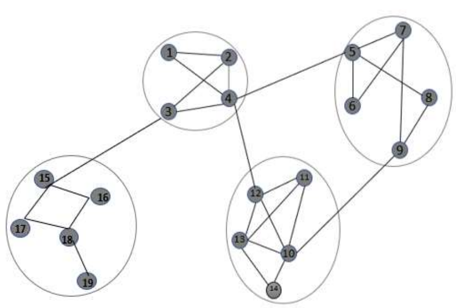

Figure 1: A Graph With Four Communities

3.1 Related Concepts

We devote this section to detail the different measures used to design our new metric and to evaluate the quality of the constructed partitions.

3.1.1 Shared neighbour's similarity measure

It is a widely used technique in graph theory, that allows to calculate the similarity between nodes based on the network properties. The vertices that share a high number of shared neighbours are usually considered to be similar to each other and they are more likely to be in the same community. In the literature, there are many similarity measures based on the principle of shared neighbour's similarity measure such as cosine similarity and Jaccard similarity [23,24,25]. In this work, we use the simple formula Bellow:

$$S(u,v) = |\text{neighbours}(u) \cap \text{neighbours}(v)| \quad (1)$$

3.1.2 Edge clustering coefficient

Constitutes one of the most important functions to estimate the quality of communities [26]. The edge clustering coefficient identifies the proximity between an edge's two connecting nodes and other nodes that surround them. The edges with a maximum value of the clustering coefficient tend to belong to the same community in the graph. This metric is mainly considered to quantify the importance of edges and describes the embeddedness of nodes in the network [27]. In this Article, we explore the equation (2) that represents, for an edge E(1,4), the proportion of the number of triangles within the neighborhoods set of nodes u and v, divided by the number of edges connecting them [22,2].

$$ECC\ (u,\ v) = \frac{Z(u,v)}{\min\ (\text{neighbours}(u)-1, \text{neighbours}(u)-1)} \quad (2)$$

Z(u,v) represents the number of triangles that include the edge E(u,v) in the graph. The min (neighbours(u)-1,neighbours(u)-1) represents the number of triangles which the edge E(u,v) may constitute at most. For example, in Fig. 2, the neighbours of the end nodes n1 and n3 of edge E(1,3) are both 4. Consequently, this edge could compose min $(4-1, 4-1) = 3$ triangles at most in theory. But in reality, there are only 2 triangles Δ_{135}, Δ_{134}, so ECC (1, 3) = 2/3=0.67.

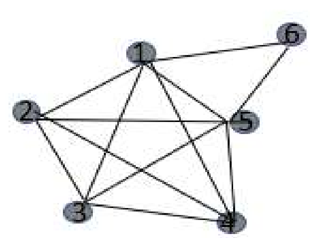

Figure 2: Example of The Edge Clustering Coefficient

Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.1.3 Modularity

This broadly used quality metric in the community detection area, introduced by [7] supplies a very efficient method to evaluate the goodness of the resulted communities. The modularity Q denotes the density of edges inside a community compared with edges between other communities in the graph. Besides, the communities having a high value of modularity are more accurate than the others with a low modularity. It is defined as follows:

$$Q = \sum_i e_{ii} - a_i^2 \qquad (3)$$

Where $\sum_i e_{ii}$ denotes the fraction of edges inside a partition i and a_i represents the fraction of edges connecting nodes in the community i.

3.2. The Proposed Node Tightness Degree

Authors in [28] have demonstrated that detecting small communities by using a similarity function then merging them is more effective than the modularity optimization method. Furthermore, the researchers in [29] analyze how the modularity maximization has the resolution limit, indeed, the graph is divided into communities in order that large communities are subdivided, and small communities are merged. In addition, the similarity measures only reveal the importance of nodes in the network but cannot indicate the importance of edges. Thus, we aim to seek the edges which are more likely to involve in the same community. For this reason, we propose a new measure based on the edge clustering coefficient measure combined with the shared neighbour's similarity measure. In the following part, we detail the new metric that describes the tightness of a node u with a community C.

Theorem. We hypothesize that we have an unweighted and undirected graph G. If two vertices u and v are connected by an edge E(u,v) with a significative value of the edge clustering coefficient and they share a large number of neighbours, these nodes will be in the same community.

Proof of theorem. In fact, recent researchers have proven that the existing centrality metrics only show the importance of nodes in the network but can not demonstrate the importance of the edges in the network. Furthermore, the measure of the edge clustering coefficient involves the tight relation between nodes connecting this edge, this property assures the density around them. In the other hand the large number of shared neighbours between two nodes ensures the strength tightness between them, which conducts to construct communities with high quality of accuracy and density.

Proposed node metric. We assess the importance of the edge clustering coefficient and the shared neighbours' similarity functions to propose a new measure combining these two metrics described as follows:

$$\mathrm{Ntd}\ (u, C) = \frac{\sum_{v \in N(u) \& v \in C} \mathrm{SoECC}(u) + S\,(u,v)}{\sum_{v \in N(u)} \mathrm{SoECC}\,(u) + S(u,v)} \qquad (4)$$

Where:

N(u) represents the set of all the neighbours of the node u.

SoECC(u) denotes the total value of edge clustering coefficients of node u such as:

$$\mathrm{SoECC}(u) = \sum_{v \in N_u} \mathrm{ECC}(u, v) \qquad (5)$$

Remarks.

i. To calculate the shared neighbours 'similarity in equation (4), we can use different measure similarity such as Jaccard, cosine.

ii. The node tightness degree of a node u; Ntd (u, C) = 1 if all the neighbours of node u are in community C, if not 0 ≤ Ntd (u, C) ≤ 1.

4. THE PROPOSED APPROACH: THE NODE TIGHTNESS DEGREE BASED COMMUNITY DETECTION ALGORITHM

In this paper, we propose a novel community detection algorithm based on a new node tightness degree. The process of this algorithm begins by identifying an initial community composed of the node having a highest tightness and its neighbours. The nodes that are not strongly tight witch nodes in the initial community C are removed. In other words, the nodes in C having a value of the node tightness degree inferior of 0.4 are removed from C, the other ones are used in to expand the community. The process of the proposed algorithm is constituted by two main phases: the first is finding the initial communities and the second one consists of expanding the discovered communities.

Phase 1: Detecting the initial community

In this phase, we use the node tightness degree Ntd (u; C) to find the initial communities. At the beginning, we select the node u with the highest tightness in the graph using this formula : $K_u = \sum_{v \in V} w_{uv}$ Therefore, the selected node u and its neighbours make an initial community C. At the next iteration, we calculate the node tightness degree Ntd (v; c) for each node v in community C. if the value of Ntd is less than 0,4 the element v is removed from community C. we repeat the last iteration until the node tightness degree of all nodes in community c is Ntd (v; c) ≥ 0,4.

Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

We note that the value of threshold can be configured according to the density of the network, to select only nodes having high tightness with the nodes of community C.

Algorithm 1: Constructing the initial community

```
1.  begin
2.  input: undirected unweighted network G(V, E)
3.  output: n Communities C = {C1, C2, C3, …, Cn}
4.  C ← empty list of communities
5.  T ← list of nodes in G
6.  repeat
7.      h ← nodes with highest node tightness degree
8.      ini ← list of h and it's neighbours
9.      repeat
10.         foreach r in ini do:
11.           bl ← calculate the node tightness degree of r
12.           if bl < 0.4 do:
13.                 remove r from ini
14.           end if
15.         end foreach
16.   until the size of the initial community remains stable
```

Phase 2: Expanding the initial community

After having constructed the initial community and identified its nodes, the second phase of the proposed approach is the expansion process. In this step, we select the neighbours of the nodes in community C and compute their node tightness degree, if the value of the Ntd is superior of 0,3 we Add these nodes to the initial community and get a larger community. This iteration is repeated until the size of community *C* remains constant. The implementation procedures of the the expansion process are as follows:

Algorithm 2: Expanding the initial community

```
1.  repeat
2.    x ← list of neighbours of the initial
community's  nodes
3.      foreach r in x do:
4.        bl ← calculate the node tightness degree of r
5.            if bl >= 0.3 do:
6.              add r to the initial community (ini)
7.            end if
8.      end foreach
9.   until the size of the initial community remains
stable
10.  add ini to C
11.  remove the selected nodes in ini from T
12. if ini contains no elements
13.      remove h from T
14.  end if
15.  until T is empty
```

5. IMPLEMENTATION

In this section, we present the evaluation we realized to evaluate our approach. The implementation of our local algorithm was developed using python programming language. This prototype was executed on a computer 32-bit architecture, Intel(R) Core (TM) i5 with a 2.30 GHz CPU, 8GB RAM, and the Windows 10 operating system. Therefore, we use the modularity to assess the algorithm performance, besides, we compute the computational time in small and large real graphs. We choose to test our approach with different real-world datasets described as follows:

5.1. Real Network Dataset

The present case study is based on a dataset of user generated profiles in a tourism context, which represents tourist preferences for tourist attractions. This dataset represents tourist id, attraction and rating. The user expresses his preferences to the visited tourist attractions giving a rating value from 1 (dislike) to 10 (like). The dataset is composed by 1200 elements, 1000 tourists and 1005 ratings.
At first, we Create the similarity matrix based on similarity measures by calculating pair-wise cosine similarity. A part of the user similarity matrix is given in Table 1.

Table 1: Example of a Similarity Matrix.

	T1	T2	T3
T1	1.000000	0.876280	0.859155
T2	0.876280	1.000000	0.935109
T3	0.859155	0.935109	1.000000

Then, we create the similarity graph by fixing the similarity threshold to 0.8. in order to retain just users having great similar preferences. Consequently, we got an undirected graph of users that the description is shown in table 2.

Table 2: Graph Description.

Number of nodes	Number of edges
1100	302104

Eventually, we use our proposed method to identify the communities in the obtained graph such as each community represents users with higher similar preferences.

Results

We use the modularity metric to evaluate the goodness of the resulting communities. The figure 3 shows the high value found by the proposed algorithm which demonstrate the goodness of its communities

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

comparatively with the other popular methods that are: CNM, WalkTrap and InfoMap. We conclude that the proposed algorithm obtains a good value of modularity compared to other methods.

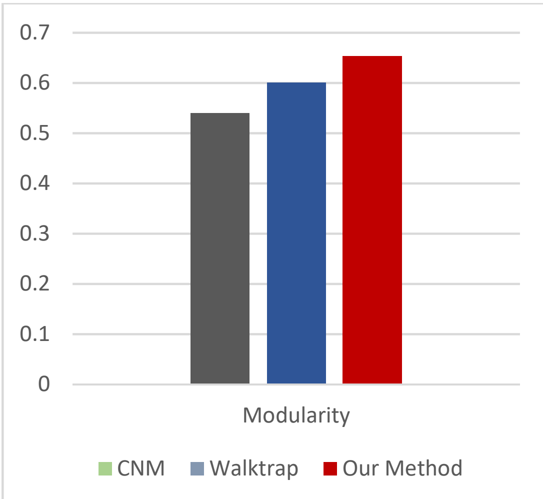

Figure 3: The evaluation Using Modularity

5.2 Real-World networks

For the second time, we evaluate our algorithm using the real-world networks. We select 4 datasets downloaded from [30,31]. These instances include the Dolphin network [32], which represents the social network of associations between 62 dolphins. The American college football network [4] which illustrates an American college football games between Division IA colleges. And the network of sold books about US politics. The last instance is the Standford Network which is a large real network. Table 3 describes the characteristics of each network.

Table 3: The Dataset Description.

Network	Number of nodes	Number of edges
Dolphin Network	62	159
American college football	115	613
Politics network	105	441
Citation	34,546	421,578

We use the modularity to compare the communities obtained by our proposed algorithm with those obtained by two other community detection methods (CNM, InfoMap). As a result, we observe that our algorithm gives good values of modularity for small and large real networks. Consequently, we conclude that the proposed community detection algorithm identifies partitions with a high quality compared to the other ones (CNM and InfoMap).

Table 4: The Modularity Values and The Computation Time of The CNM, InfoMap and Our Algorithm.

Network	Our Algorithm	CNM	InfoMap
Dolphin Network	0.567 (0.04)	0.49 (0.019s)	0.532 (0.2s)
American college football	0.601 (0.09)	0.57 (0.05s)	0.6 (0.28s)
Politics	0.52 (0.03s)	0.5 (0.04s)	0.52 (0.15s)
Citation	0.638 (1992s)	0.556 (225.95s)	0.579 (2879s)

In table 4, we can observe, as a result, that the computation time of the proposed algorithm takes the second one after the CNM method, which prove the rapidity of the new community detection algorithm.

6. CONCLUSION

The identification of community structure in large graphs helps to understand the network performance, communities have been used to reduce the computational complexity of several operations on complex networks, they have been also the subject of several works for systems recommendation, etc.
Many approaches have been proposed to detect communities in large graphs, but they are not satisfactory considering the computation time and quality of partitions.

The proposed algorithm in this paper improves the local community detection algorithm purely based on the node clustering. Its novelty is considering a new metric which is the node tightness degree based on the edge clustering coefficient and the shared neighbours' similarity. There are many algorithms using standards based on the node clustering coefficient and the edge clustering coefficient to weigh the compact degree between the nodes and the communities and then agglomerate the communities continuously, but the modularity of discovered partitions is not satisfactory. Our algorithm draws lessons from these methods and invokes much more powerful metrics than modularity to improve the quality of the discovered community structure. This new metric allows to detect better communities than the communities resulted from the well-known methods.

To test our approach, an evaluation is performed on small and large datasets of different networks to prove the effectiveness of the proposed algorithm in terms of modularity and time execution. The results were successful, moreover, our proposed approach can be applied for any type of networks. We infer that our algorithm uncovers the communities with better quality.

In a future work, we will seek to improve our method by optimising the algorithm complexity to reduce the computation time for massive graphs.

REFERENCES:

[1] Javed, M. A.; Younis, M. S.; Latif, S.; Qadir, J.; Baig, A. Community detection in networks: A multidisciplinary review, *Journal of Network and Computer Applications*, 108, 2018, pp. 87-111.

[2] Fortunato, S, Community detection in graphs, *Physics reports* 486(3), 2010, pp. 75-174.

[3] Porter, M.A.; Onnela, JP.; Mucha PJ. Communities in networks. *Not AMS* 2009, *56(9),* pp. 1082–1097.

[4] Girvan, M.; Newman, ME. Community structure in social and biological networks. *PNAS, 99*, 2002, 7821–7826.

[5] Radicchi, F.; Castellano, C.; Cecconi, F.; Loreto, V.; Parisi, D. Defining and identifying communities in networks. In Proceeding of National Academy of Sciences, USA, (23, 02, 2004).

[6] Li, K.; Qing, G.; Rao Z.; Daoxu C. A local community partition algorithm based on the clustering coefficient. IEEE Access,7, 2019**,** 121586 – 121598.

[7] Clauset, A.; Newman, MEJ.; Moore C. Finding community structure in very large networks. *Phys Rev*, 70(6):066111*,* 2005.

[8] Morvan, A.; Choromanski, K.; Gouy-Pailler, C.; Atif, J. Graph sketching-based massive data clustering. In Proceedings of the 2018 SIAM International Conference on Data Mining. San Diego, California, USA, 3-5 May 2018.

[9] Blondel,V.D.; Guillaume, J.L.; Lambiotte, R.; Lefebvre, E. Fast unfolding of communities in large networks, Journal of Statistical Mechanics Theory and Experiment, 10, 2008.

[10] Rosvall, M.; Bergstrom, CT. Maps of random walks on complex networks reveal community structure, PNAS, 105, 2008, 1118–1123.

[11] Pons, P.; Latapy, M.; Computing communities in large networks using random walks, J Graph Algorithms Appl, 10, 2006, pp. 191–218.

[12] Newman, M. E. J. Community detection and graph partitioning. EPL,103, 2013.

[13] Dhumal, A.; Kamde, P. Survey on Community Detection in Online Social Networks, International Journal of Computer Applications*,*121*,* 2015.

[14] Furht, B. Handbook of social network technologies and applications. In Springer Science & Business Media. 2010**,** 331-346.

[15] Slaninová, K.; Martinovič, J.; Dráždilová, P.; Obadi, G.; Snášel, V. Analysis of social networks extracted from log files, In Handbook of Social Network Technologies and Applications: Springer 2010, pp. 115-146.

[16] Murata, T. Detecting communities in social networks. In Handbook of social network technologies and applications: Springer 2010, p.p: 269-280. doi:10.1002/widm.1178.

[17] Madani, F. Technology Mining' bibliometrics analysis: applying network analysis and cluster analysis. *Scientometrics* 2015**,** 105(1), 323-335.

[18] Li, Z; Liu, J. A multi-agent genetic algorithm for community detection in complex networks. *Physica A: Statistical Mechanics and its Applications* 2016, *449*, 336-347. DOI: 10.1016/j.neucom.2016.06.030.

[19] Mirsaleh, M.R.; Meybodi, M.R. A Michigan memetic algorithm for solving the community detection problem in complex network. *Neurocomputing (214)* 2016**,** pp. 535-545.

[20] Ma, L.; Gong, M.; Liu, J.; Cai, Q.; Jiao, L. Multi-level learning based memetic algorithm for community detection. *Applied Soft Computing (19)* 2014**,** pp.121-133.

[21] Sobolevsky, S.; Campari, R.; Belyi, A.; Ratti, C. General optimization technique for high-quality community detection in complex networks. *Physical review E* 2014, 90(1), 012811.

[22] Deng, K.; Zhang, J.-P.; Yang, J. An efficient multi-objective community detection algorithm in complex networks. Tehnicki vjesnik/Technical Gazette *22(2)*, 2015, pp.319-328.

[23] Charu, C. A.; Yan, X.; Philip, S.Y. Towards community detection in locally heterogeneous networks. In Proceedings of the 2011 SIAM international conference on data mining. SIAM, 3. Arizona, USA, 28-30 April 2011.

[24] Liu, Z.; Peng Li.; Yabin, Z.; Maosong, S. Community detection by affinity propagation. Technical Report. 2008.

[25] Partha, B.; Siddhartha, A.; Diksha, R.S.; Khusbu. M.; Sanjoy, K.S. Detection of communities in social networks using spanning tree. *In Advanced Computing, Networking and* I*nformatics (2)* 2014, pp. 589–597.

[26] Soumita, S.; Debnath, B.; Kim, T.H. Constraint-Based Clustering to Find Communities in a social network, In Proceedings of international conference on information science and industrial applications (ISI), Cube, Philippines, May 2012.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[27] Mariá, C.V.; Nascimento. Community detection in networks via a spectral heuristic based on the clustering coefficient, Discrete Applied Mathematics (176), 2014, pp. 89–99.
[28] Arab, M.; Afsharchi M. Community detection in social networks using hybrid merging of sub-communities. *J Netw Comput Applic*, 40, 2014, pp.73–84.
[29] Fortunato, S.; Barthelemy, M. Resolution limit in community detection. *PNAS* 2007, 104(1), pp. 36–41.
[30] Leskovec, J.; Rok, S. SNAP: A General-Purpose Network Analysis and Graph-Mining Library, *ACM Transactions on Intelligent Systems and Technology (TIST)(8)* 2016, pp.1-20.
[31] Newman, MEJ. Available online: http://www-personal.umich.edu/ mejn/netdata. accessed on 6 March 2019.
[32] Lusseau, D.; Karsten, S.; Oliver, J.; Boisseau, P.H.; Elisabeth, S.; Steve,M.D. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations, *Behavioral Ecology and Sociobiology* (3), 2003, pp.396–405.