MEASURING RESEARCH INTEREST SIMILARITY AMONG AUTHORS USING COMMUNITY DETECTION

HARITHA AKKINENI[1], MYNENI MADHU BALA[2], VENKATASUNEETHA TAKELLAPATI[3], MADHURI NALLAMOTHU[4], SURESH YADLAPATI[5]

[1]Associate Professor, Department of Information Technology, PVP Siddhartha Institute of Technology, Vijayawada, Andhra Pradesh, India

[2]Professor, Computer Science and Engineering, Institute of Aeronautical Engineering, Hyderabad, Telangana, India.

[3]Assistant Professor,Department of CSE,Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, Telangana, India

[4]Assistant Professor, Department of CSE, Dhanekula Institute of Technology, Vijayawada, Andhra Pradesh, India.

[5]Assistant Professor, Department of Information Technology, PVP Siddhartha Institute of Technology, Vijayawada, Andhra Pradesh, India

[1]aharithapvpsit@gmail.com, [2]baladandamudi@gmail.com, [3]takkellapati9@gmail.com,
[4]nallamothu.madhuri@gmail.com,[5]sureshyadlapati@gmail.com

ABSTRACT

Social dynamics that govern human phenomenon are the real need of the hour to access the community structures in social networks. In the present world, online social networks provide huge data that includes the objects information and comments which are analyzed and lead to discovering information and relationship among the networks. Finding community detection is an existing and attracting the researchers where they use different algorithms, one is mathematically based which work on connections in the community and the other one is the graph the structure which shows the output and it is similar to the topological structure. These traditionally followed algorithms and structures are having their limitations. This article attempts to overcome these drawbacks by identifying communities in social networking sites using density-based clustering technique DBSCAN. The identification and removal of such noisy nodes in the identified communities improves the quality. The method's ability to detect different community structures has been demonstrated in studies on synthetic and real-world networks such as research gate, where scientists communicate and share their work and build their reputations.

Keywords: *Density Based Clustering, DBSCAN, Community Detection, Fast Greedy, Visualization*

1. INTRODUCTION

Social Networks and online communications between people have increased significantly and important part of a social network is its connections and these are some kind of relationship between the users. A Group of users who are more strongly connected to each other with other users in the network forms a community. Detecting such communities are hard and There are many general methods applied to detect communities there are several algorithms and approaches available to detect communities. One of the significant methods for community detection are Grivan newman algorithm also known as Edge Betweenness, Fast Greedy, Lable propagation, Louvain, Walktrap, Infomap. None of these methodologies can recognize noise as nodes are not members of any community that can confront the problem density-based community. The detection algorithm is relevant since they provide the best to leave specious connected nodes such as noise from the detected community.

A network is a collection of objects that are connected to one another. A common category is the social network, in which the objects are people and the social relations that exist between them can be considered connections. Social relationships can be based on some form of commonality, such as common friendships, interests, and so on. The most important property to investigate in many networks is community structure, which can be simply defined as the division of networks into groups known as clusters that exhibit strong interconnectedness and welfare interconnectedness. This type of observation, which divides network data into a number of clusters, can provide useful insights into how the structure of ties affects individuals and their relationships. This kind can be applied for research gate data to detect clusters of different research interests.

The DBSCAN algorithm was used to detect outliers, which identifies communities in a social network. Outliers, also known as "noisy nodes," are excluded from network graph. The proposed method in the paper enables the detection and removal of noisy nodes in detected communities, resulting in an improvement in quality.

The graph uses the well-known DBSCAN Algorithm, which is a density-based approach that could be used to detect communities. Similarly to DBSCAN, it has two parameters: a density-based level eps and a lower bound Min Pts for the number of nodes that form a community.

2. RELATED WORKS

There have been many community detection algorithms developed in the past, but only a few of them are massive scale algorithms that can be used in social media graphs.

By removing edges from the original network, the [1]Girvan–Newman algorithm detects communities. The communities are the remaining network's connected components. We need to build an indicator that informs us that the edges are indeed the central communities; this method has a lot of edges that are probably "between" communities. Which has the edges that define "edge betweenness" of the edge which contains the number of shortest paths between both the nodes that will run in it If the network contains communities that are only loosely attached by inter-group edges, the shortest paths between them should follow a few of these edges. As a result, the edges linking communities have a high degree of edge betweenness. Then By removing these edges, the groups are separated from one another and the community structure of the network is revealed.

Algorithm for community detection: The first step is to create a graph of N nodes and edges, which will be used as the inbuilt graph. The betweenness of all live edges in the network is formulated in the following step. The edge with the greatest betweenness is removed first. The removal is then recalculated after the betweenness of all the edges is affected. Steps 3 and 4 are repeated until there are no more edges. It is recognized that by adjusting the betweennesses and removing each edge, the least one of the remaining edges between the two communities will always have the greatest benefit. The dendrogram is the outcome of the Girvan–Newman algorithm. The Girvan–Newman algorithm generates a dendrogram.

Fast greedy algorithm [2] the problem-solving which make it as optimal choice at each stage. It has many problems and greedy strategy which does not produce optimal result, Greedy may have optimal solutions that have best solution in a maximum amount of time. Fast Greedy algorithm, It has a set in which a solution is generated After the selection function it has the best candidate to be added into the solution Then after the feasibility function it is used to determine either a candidate can be used to have a solution or not objective function assign values to the result. Result function will indicate when we get the entire solution. But for many other problems the fast greedy algorithms fail to produce the best result, and may also produce the worst result.

The Label Propagation Algorithm [3] is used to assign labels to unlabeled nodes by propagating labels through various datasets. The edge connecting two nodes has few similarities with the connection between other algorithms label propagation can have different community structures that have starting condition. The solutions are reduced when some nodes are given with preceding labels and while others are unlabelled. And these unlabelled nodes will be more likely to adopt the labeled ones. This algorithm uses the labels of previously labeled nodes as its foundation and attempts to anticipate the labels of unlabeled nodes. As an example, if the initial labeling is incorrect, it can disrupt the label propagation procedure and cause labels to be propagated. The Louvain method [4]for community detection is to extract communities from large social networks. This is an unsupervised algorithm and it does not require the input of the number of communities or size before execution and it is divided into two phases: Modularity Optimization, Community Aggregation, After the first step is completed, the second will be executed, and both are repeated until there are no modifications in the network and maximum modularity is achieved.

Walktrap [5,7]is a ordered clustering algorithm which has an idea of this method which has the short distance walk and likely will stay in the same community. Distances between nearest neighbors are computed starting with a non-clustered partition.

Infomap algorithm [6] reduce the cost that is based on the flow that was created by the pattern of connections in a given network. Another way to choose the same path in a more incisive way is by Huffman coding approach. This approach also shows that the community finding algorithms can be also used to solve the compression problems and this approach also shows that the community finding algorithm can be also used to solve compression problems.

In this paper [8] the authors Madhu Bala Myneni, Rohit Dandamudi stated the sentiment analysis of tweets given by railway passenger using novel social graph clustering approach. Here the sentiment analysis is performed on every detected cluster to predict the people’s opinion and also helps in improving customer experience.

From these different algorithms, DBSCAN is a best-unsupervised algorithm that is done to accentuate community detection in the social networks. By removing the outliers the dataset will be noise-free.

Density-based clustering algorithm has ability to extract the clusters without the prior knowledge on number of clusters, also in the case where there is noise. The clustering is based on two parameters eps and MinPts, which are by the density level eps and a lower bound and the number of points in a MinPts[9].

The Louvain algorithm [10] finds the community structure with the maximum modularity by moving every node to the neighbourhood of its neighbour with the highest modularity rise, compacting the community as a supernode, and repeating the prior processes till the modularity is the highest.

The community structure can be obtained by deleting the inter-community edges with a large distance. Attractor [11] is a network dynamics-based method for controlling the interaction between nodes and the distance between them.

LPA [12] is a high-efficiency community detection algorithm depending on the structure of disseminating information. Each node is initially assigned a specific label, and it updates its own label to be one that occurs the most frequently in its neighborhood. The label update procedure is repeated until the label of each node is the most frequently used among its neighbors.

To identify communities in large scale networks, greedy modularity optimization was used by Clauset et al[13] . The method has a running time of (mdlogn) for a network structure with m edges and n vertices, where'd' indicates the depth of the dendrogram.

Blondel et al. [14] use the Louvain method to calculate the modularity gain of moving a node I from one community to another. During the first phase, all nodes are assigned to different communities, and then gains are obtained by rearranging them.

The Markov Clustering Algorithm (MCL)[15] is a graph simulation model for detecting clusters in a graph. This method is comprised of two alterative processes: 'expansion' and 'inflation.'

3. METHODOLOGY

Clustering algorithms look for similarities and differences between data points. Partition-based clustering algorithms such as k-means and k-median are examples of clustering algorithms. Agglomerative and Divisive clustering are examples of hierarchical clustering. DBSCAN, for example, is a density-based clustering algorithm. DBSCAN algorithm makes it a perfect fit for outlier detection. Algorithms like K-Means Clustering lack the property and has clusters that are very sensitive to outliers.

DBSCAN is that which belongs to a cluster if it is close to many points from that cluster, so there are two key parameters of DBSCAN eps it has the distance that specifies the neighbors. If the distance between two points is less than or equal to eps, they are considered neighbors. MinPts has the smallest number of data points required to define a cluster based on two parameters. These points are classified as core points, border points, and outliers. A core point is a point where there are at least MinPts number of points that surround the point with radius eps. Border points are non-core points that are outliers and cannot be reached by any other core points. As DBSCAN does not stress the number of cluster before hand, it is much useful for the type of input data we have chosen. i.e research gate data. There is a scope for us to arbitrarily find the number of clusters as opposed to k-means

A network is mathematically defined as G(N,E) where N is the number of nodes and E is the number of edges E € { {e1,e2} | e1,e2 € N and e1≠e2}.

A community is defined as a cluster of nodes N where the connections are dense and these nodes are connected by edges E.

DBSCAN starts with an arbitrary Node or Edge which hasn't been visited then its neighbourhood information is rescue from the eps parameter. If it contains MinPts within eps neighbourhood, community formation starts. Otherwise the aim is labeled as noise. The above process continues until the density-connected cluster is totally found. The approach of DBSCAN is used in three different ways such as Perform DBSCAN to detect noise points. Perform DBSCAN to remove edges which are marked as noise. Perform DBSCAN to remove nodes which are marked as noise. The Output of DBSCAN algorithm depends on values on MinPts. The optimal epsilon value is found using. Varying the MinPts helps us in detecting communities. Figure 1 depicts the overall block diagram of the proposed methodology.

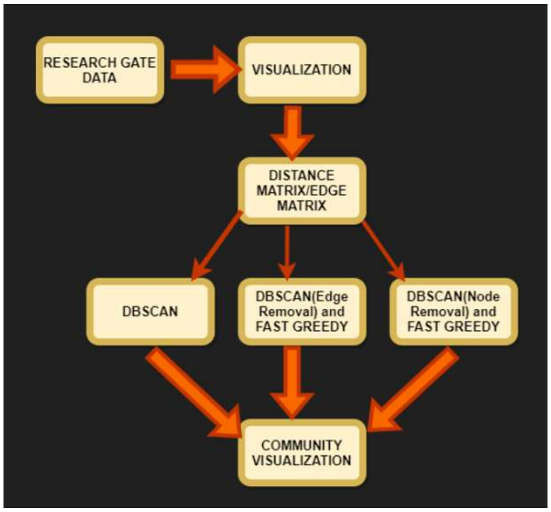

RESEARCH GATE DATA
VISUALIZATION
DISTANCE MATRIX/EDGE MATRIX
DBSCAN
DBSCAN(Edge Removal) and FAST GREEDY
DBSCAN(Node Removal) and FAST GREEDY
COMMUNITY VISUALIZATION

Figure 1: Block diagram of proposed DBSCAN

3.1 Algorithm: Proposed DBSCAN

```
Import the Libraries
Load the Dataset
Perform DBSCAN to detect communities and noise points
for each point p in dataset do
    if p is equal to noise then
        remove p
Perform Fast Greedy on the new data
Visualizing newly detected communities
End
```

4. EVALUATION

Research Gate dataset is used in this paper. Initially the network is visualized. The edge data is converted into distance matrix and edge matrix. DBSCAN is performed on the matrix. The outliers are removed from the network. The algorithm Fast Greedy is applied on the network after removing outliers. Finally the communities formed are visualized. The communities are evaluated using modularity score.

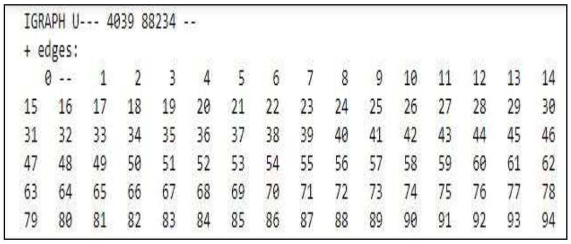
The most popular evaluation metrics to assess the quality of communities in community structure detection algorithms is modularity [16]. Modularity is stated as.

$$Q = \frac{1}{2M}\sum u,v\left(A_{uv} - \frac{k_u k_v}{2M}\right)S_{uv} \qquad (1)$$

where A is the adjacency matrix of the network, k_u and k_v are the inward and outward degrees of node , and s_{uv} is 1 if nodes u and v have the same community membership, and 0 otherwise. M is the number of edges in the graph. Modularity is the gold standard for assessing the goodness of a network's community structure, but it works best when all nodes are clustered into their perfect communities.

5. RESULTS

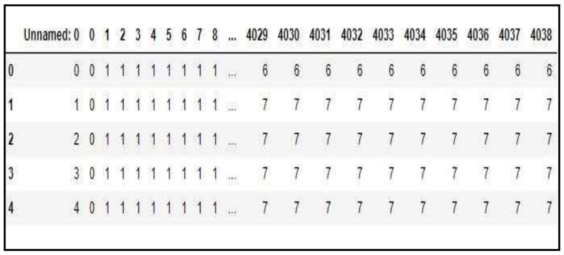
ResearchGate data was collected. This dataset consists of 4038 nodes and 88234 edges and dataset includes node features, circles and ego-networks. Ids have been replaced with a new value for each user. The sample input is shown in Figure 2.



```
IGRAPH U--- 4039 88234 --
+ edges:
   0 --  1  2  3  4  5  6  7  8  9 10 11 12 13 14
  15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
  31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46
  47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62
  63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78
  79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94
```

Figure 2: Sample Input Data

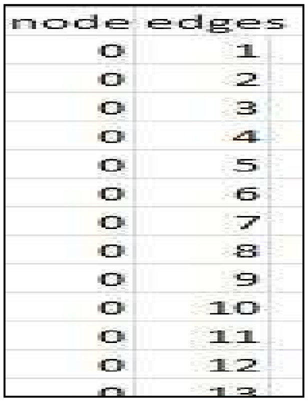
The distance matrix on taken input by using Euclidian distance is shown in Figure 3.



```
   Unnamed: 0  0  1  2  3  4  5  6  7  8  ...  4029  4030  4031  4032  4033  4034  4035  4036  4037  4038
0           0  0  1  1  1  1  1  1  1  1  ...     6     6     6     6     6     6     6     6     6     6
1           1  0  1  1  1  1  1  1  1  1  ...     7     7     7     7     7     7     7     7     7     7
2           2  0  1  1  1  1  1  1  1  1  ...     7     7     7     7     7     7     7     7     7     7
3           3  0  1  1  1  1  1  1  1  1  ...     7     7     7     7     7     7     7     7     7     7
4           4  0  1  1  1  1  1  1  1  1  ...     7     7     7     7     7     7     7     7     7     7
```

Figure 3: Distance Matrix

The edge matrix is shown in Figure 4. It gives the weightage of each edge among nodes.



```
node edges
   0     1
   0     2
   0     3
   0     4
   0     5
   0     6
   0     7
   0     8
   0     9
   0    10
   0    11
   0    12
   0    13
```

Figure 4: Edge Matrix

Epsilon Value: The optimal epsilon value will be found at the maximum point of curvature. The optimal epsilon values taken in DBSCAN is shown in Figure 5.

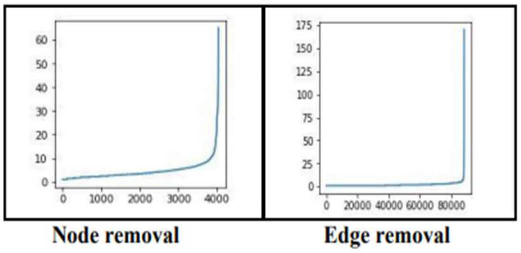


Figure 5: Optimal epsilon value for DBSCAN

5.1 Communities Detected Using DBSCAN

The red color nodes are marked as Noise by DBSCAN. The data was evaluated for different MinPts. These are the communities detected by DBSCAN.

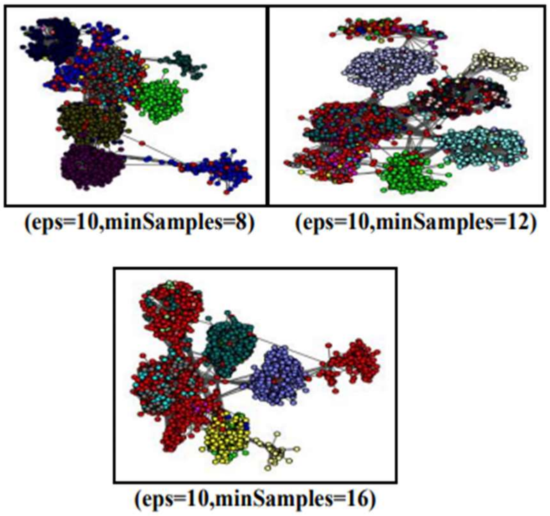


Figure 6: Communities Detected Using DBSCAN

Figure 6 shows the community detection visualizations with optimal epsilon and variant MinPts. It is observed that, among all eps 10 and min samples 8 are highly appropriate for the taken data set to visualize the community clusters. Table 1 shows the analysis of DBSCAN algorithm in community detection with variant minimum points is shown. It gives various parameters response with respect to the epsilon points.

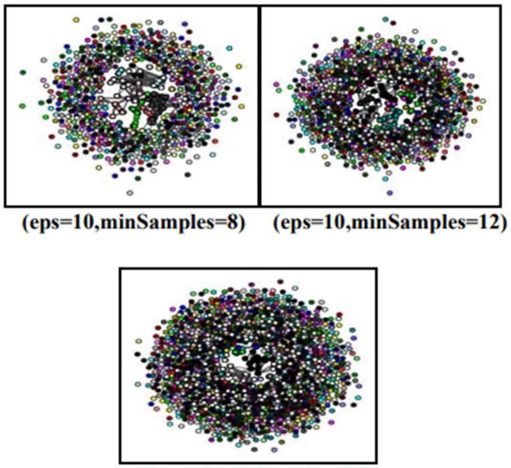
Table 1: Analysis of DBSCAN

DBSCAN (*eps=10, MinPts)*	Clusters	Modularity	Noise Points	Edges Removed	Nodes Removed
8	26	0.58	603	0	0
12	45	0.47	1105	0	0
16	17	0.46	2144	0	0

5.2 Communities detected using DBSCAN (Node removal) and Fast Greedy

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

In this approach the communities detected are large although they have a good modularity score.


(eps=10,minSamples=8) (eps=10,minSamples=12)

(eps=10,minSamples=16)

Figure 7: Communities Detected Using DBSCAN (Node removal) and fast greedy

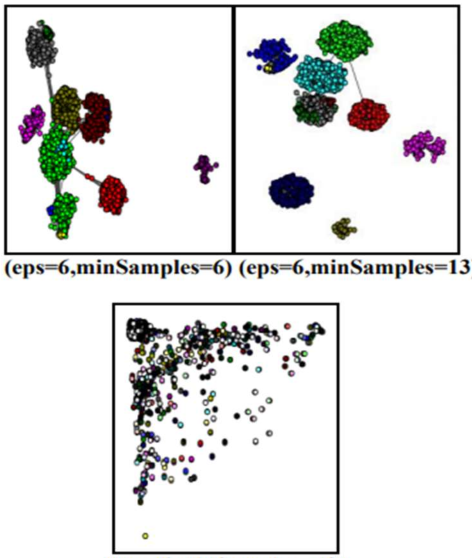
Figure 7 shows the community detection visualizations with optimal epsilon and variant MinPts. It is observed that, among all eps 10 and min samples 8 are highly appropriate for the taken data set to visualize the community clusters. The modularity score observed is high and noise is less for these values. Table 2 gives analysis on various parameters of DBSCAN with node removal and fast greedy to represent clusters with respect to the variant minimum samples.

Table 2. Analysis Of DBSCAN(Node Removal) And Fast Greedy

Node removal *(eps=10, MinPts)*	Clusters	Modularity	Noise Points	Edges Removed	Nodes Removed
8	757	0.83	603	33,308	603
12	1243	0.83	1105	53240	1105
16	2284	0.77	2144	68845	2144

5.3 Communities detected using DBSCAN (Edge removal) and Fast Greedy

The communities detected are clearly visible in this approach also the formed communities have a good modularity score.


(eps=6,minSamples=6) (eps=6,minSamples=13)

(eps=6,minSamples=18)

Figure 8: Communities detected using DBSCAN (Edge removal) and Fast Greedy

Figure 8 shows the community detection visualizations with optimal epsilon and variant MinPts. It is observed that, among all eps 6 and min samples 6 are highly appropriate for the taken data set to visualize the community clusters. Table 3 gives the analysis on various parameters of DBSCAN edge removal and fast greedy approach.

Table 3. Analysis of DBSCAN(edge removal)and Fast Greedy

edge removal *(eps=6, MinPts)*	Clusters	Modularity	Noise Points	Edges Removed	Nodes Removed
6	12	0.80	8,507	8507	0
13	11	0.76	43,315	43,315	0
18	2202	0.68	62194	62194	0

5.4 Analysis of DBSCAN variants

There is no ground truth data available for the dataset. We used modularity score to evaluate how well the clusters are formed. Modularity is a network or graph structure metric that measures the strength of a network's division into modules (also called groups, clusters or communities). High modularity networks have dense connections between nodes within modules but sparse connections between different nodes in different modules - this is known as a high modularity network. The DBSCAN node removal tends to obtain high-modularity partition and get results more handily, and the other two can partition the networks as accurately as possible even when the community structure is not clear.

TABLE 4. Analysis of DBSCAN Variants

Algorithm	Clusters	Modularity	Noise Points	Edges Removed	Nodes Removed
Dbscan	26	0.58	603	0	0
Node Removal	757	0.83	603	33,308	603
edge removal	11	0.76	43,315	43,315	0

6. CONCLUSION

This proposed community detection using different DBSCAN approaches with Fast Greedy shows high performance. Community detection of Research Gate network is successfully performed using different DBSCAN approaches with Fast Greedy. Our approach was able to detect communities quickly and efficiently. Community detection techniques help us understand more of users' collective behavior by clustering similar users based on common research interests. Certain groups were found like Mathematics, Computer science, Physics, Medicine, Social science and astrophysics. We were able to detect communities in the network with a good modularity value. The proposed models were able to detect communities in complex networks efficiently. Further optimizations in the code can achieve better results. This model can achieve better results in sparse networks and networks which do not have dense connections.

REFERENCES:

[1] Despalatovic, L., Vojkovic, T., & Vukicevic, D. (2014). Community structure in networks: Girvan-Newman algorithm improvement. 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). doi:10.1109/mipro.2014.6859714

[2] Bakillah, M., Li, R.-Y., & Liang, S. H. L. (2014). Geo-located community detection in Twitter with enhanced fast-greedy optimization of modularity: the case study of typhoon Haiyan. International Journal of Geographical Information Science, 29(2), 258–279. doi:10.1080/13658816.2014.964247 S. Papadopoulos, Y. Kompatsiaris, A. Vakali, and P. Spyridonos,(2012) "Community detection in social media," Data Mining and Knowledge Discovery, vol. 24, no. 3, pp. 515–554.

[3] Garza, S. E., & Schaeffer, S. E. (2019). Community detection with the Label Propagation Algorithm: A survey. Physica A: Statistical Mechanics and Its Applications, 22058. doi:10.1016/j.physa.2019.122058

[4] Que, X., Checconi, F., Petrini, F., & Gunnels, J. A. (2015). Scalable Community Detection with the Louvain Algorithm. 2015 IEEE International Parallel and Distributed Processing Symposium. doi:10.1109/ipdps.2015.59

[5] Seunghyeon Moon, Jae-Gil Lee, & Minseo Kang. (2014). Scalable community detection from networks by computing edge betweenness on MapReduce. 2014 International Conference on Big Data and Smart Computing (BIGCOMP). doi:10.1109/bigcomp.2014.6741425

[6] Yu-Liang, L., Jie, T., Hao, G., & Yu, W. (2012). Infomap Based Community Detection in Weibo Following Graph. 2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control. doi:10.1109/imccc.2012.286.

[7] Hu, F., Zhu, Y., Shi, Y., Cai, J., Chen, L., & Shen, S. (2017). An algorithm Walktrap-SPM for detecting overlapping community structure. International Journal of Modern Physics B, 31(15), 1750121. doi:10.1142/s0217979217501211

[8] Madhu Bala Myneni, Rohit Dandamudi, (2020) Harvesting railway passenger opinions on multi themes by using social graph clustering, Journal of Rail Transport Planning & Management, Volume13,100151,https://doi.org/10.1016/j.jrtpm.2019.100151.

[9] Rahmah, N., & Sitanggang, I. S. (2016). Determination of Optimal Epsilon (Eps) Value on DBSCAN Algorithm to Clustering Data on Peatland Hotspots in Sumatra. IOP Conference Series: Earth and Environmental Science, 31, 012012. doi:10.1088/1755-1315/31/1/012012

[10] V. D. Blondel, J.L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," Journal of Statistical Mechanics: Theory and experiment, vol. 2008, no. 10, Article ID P10008, 2008.

[11] J. Shao, Z. Han, Q. Yang, and T. Zhou, "Community detection based on distance dynamics," in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1075–1084, Sydney, Australia, August 2015.View at: Publisher Site | Google Scholar

[12] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," Physical review. E, Statistical, nonlinear, and soft matter physics, vol. 76, Article ID 036106, 2007.

[13] Clauset A, Newman ME, Moore C. Finding community structure in very large networks. Physical review E 2004, 70 (6):066111. doi:10.1103/PhysRevE.70.066111.

[14] Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 2008. doi:10.1088/1742-5468/2008/10/P10008

[15] Dongen SV. Graph Clustering by Flow Simulation, PhD thesis, University of Utrecht. 2000.

[16] Newman, M.E.J. (2006) Modularity and Community Structure in Networks. Proceedings of the National Academy of Sciences of the United States of America, 103, 8577-8858. http://dx.doi.org/10.1073/pnas.0601602103