Implementation of DBSCAN Algorithm for Closed Community Detection in Social Networks

Myneni Madhu Bala, Jhansi Lakshmi Bai K
VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad– 500080, INDIA

ABSTRACT

In the present world, online social networks are rich in multimodal data sources with various objects, URLs, and comments. These are the real-time dynamic sources for the analysis which will lead to the discovery of facts and hidden relationships among the closed community groups in networks. Finding a closed community in online social networks is a challenging task for various purposes of applications. A closed community is formed with a group of similar-minded people and may be related to political, ethnic, or religious. The governance of such groups consciously applies limitations on the network links with outside communities. Broadly, two concepts of algorithms viz., clustering and network partitioning are used for the detection of such groups. These algorithms are based on dynamic networks with humans as key players and the other one is the graph structure similar to the topological structure. However, these algorithms suffer from limitations such as these communities provide no knowledge of groups in advance, the requirement of an extensive analysis of all possible partitions, etc. This article aims to overcome the said limitations by using the fast greedy approach by fusing with a Density-based clustering technique called DBSCAN for the detection of such communities. Detection and deletion of these noisy nodes in the communities lead to the development of quality. The comparison of the experimental results proved that the removal of noisy nodes will impact the quality of the community detection.

Keywords: Social Networks, Density-based clustering, community detection, Fast Greedy, Facebook, MinPts.

I. INTRODUCTION

Social Networks and online communications between people have increased significantly and an important part of a social network is its connections and these are some kinds of relationships between the users. A group of users who are more strongly connected users in the network forms a community. Detecting such communities is hard. There are many general approaches, algorithms, and methods available and applied to detect communities. One of the significant methods for community detection is the Grivan Newman algorithm also known as Edge Between-ness, Fast Greedy, Lable propagation, Louvain, Walktrap, and Infomap. Neither one of these algorithms is apt to identify the noise, as nodes are not the members of the community to manage the problem DBSCAN algorithm is relevant since it provides the best to leave specious connected nodes such as noise, out of the detected community.

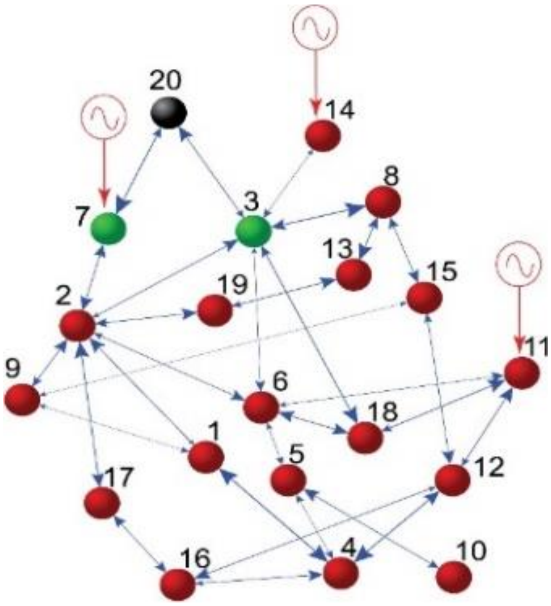


Fig. 1. An illustration of a network with the complexity of noise sources and hidden nodes [11]

The network in Fig.1. shows the complexity of a network in which the measuring of time series is possible from all nodes but not from the hidden node numbered 20. It can be identified or detected only with the identification of the nodes numbered 3 and 7 is possible without any ambiguity. The sources of local noise drive the nodes numbered 7, 11, and 14.

This Density-Based dimensional Collection of Applications with Noise (DBSCAN) algorithm has been taken for detecting the outliers(noise) and this detects the communities in a social network and noise nodes are also separated from the network graph. The main algorithm in the paper focuses on the Detection and deletion of these noisy nodes in the communities leading to the development of quality. It has two variables, the density-based level with a lower bound is MinPts and eps for the nodes' number that forms a community.

The paper is organized as follows: Section II describes the related works, III describes the proposed Density-

Based Spatial Clustering of Applications with Noise algorithm and Section IV provides results and the Discussions. The concluding remarks are given in Section V.

II. RELATED WORKS

There are many community detection algorithms that have appeared in the past, but only some among them are extensive algorithms that are relevant to social media graphs.

The algorithm of [1] Girvan–Newman separates edges from the network for the detection of communities. Communities are the components of the left-out network. We need to compose such that the central communities can be revealed from the edges, and which has edges similar to "between" communities. It has the edges that define the edge-betweenness it has the shortest paths in the middle of nodes that will run in it. The shortest paths between different communities should go through one of these edges if the network has loosely connected communities by an inter-group edge. Therefore, the communities connecting with edges have high ‘edge-betweenness’, by removing these edges, they are separated, and the structure of the community network is released. Algorithm for community detection: In the first step we need to create a graph of N nodes and edges are taken as the in-built graph in the next step for the calculation of the betweenness of all edges in the network. The first removed edge is that which has the highest betweenness. Then affliction will be done for the between-ness of all other edges and hence the removal is calculated again. Till no edges remain, the steps 3 and 4 are repeated. This shows that sustainable value can be achieved for the least one of the remaining edges between the two communities. The result will be the dendrogram at the end of the algorithm. As the Gir-van–Newman algorithm runs, the dendrogram is produced from the top down.

Fast greedy algorithm [2] the problem-solving which makes it an exceptional choice at each stage. It has many problems, and a greedy strategy does not produce a better result, Greedy may have optimal solutions that have the best solution in a maximum amount of time. Fast Greedy algorithm, has a set in which a result is generated after the function it has the best applicant to be added into the result then after the practical function, it is used to determine whether an applicant can be used to have a result that is or not objective function assign values to the result. The result function will indicate when we get the entire solution. But for many other difficulties, the Fast greedy algorithms did not make the best result and may also give the worst result.

The label Propagation algorithm [3] allocates labels to the unlabeled nodes by propagating labels through the different kinds of 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 conditions. The solutions are reduced when some nodes are given with preceding labels while others are unlabeled. And these unlabeled nodes will be more likely to adopt the labeled ones. This algorithm has the labels of the already labeled nodes as their ground and they try to predict the labels of the unlabeled nodes. As, if the first labeling is wrong this can affect the label propagation process and labels may get 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 it is divided into two phases: Modularity Optimization and community Aggregation, After the beginning step is done the next follows later both will be executed until there are no changes in the network and then the greatest modularity is reached.

Walktrap [5] is an ordered clustering algorithm. Which has an idea of this method which has a short distance walk and likely will be in the same community. In the non-clustered partition, the distances between the adjoining nodes are calculated.

Infomap algorithm [6] reduces the cost 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 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 paper [9] the authors stated the sentiment analysis of tweets given by railway passengers using a novel social graph clustering approach. Here the sentiment analysis is performed on every detected cluster to predict the people’s opinions and also helps in improving customer experience.

From these different algorithms, DBSCAN is the best unsupervised algorithm that is done to accentuate community detection in social networks. The results specify that the large bias members by core, less bias by the border, and members with no effect in the groups are considered as Noise. By removing the Noise, the dataset will be noise-free.

DBSCAN algorithm has the capability to remove the clusters without the initial recognition on a number of clusters, also where there are outliners. The clustering is based on two variables eps and MinPts, which are by the density level eps and a lower bound and the number of points in a MinPts.

III. DBSCAN ALGORITHM

Clustering is a set of similar assembled points of data. So, algorithms of clustering seek likeness or alikeness among the points of data. Clustering can be performed by various algorithms which are Partition-based, hierarchical and density-based clustering. Partition-based algorithms include k-median and k-means clustering. Hierarchical algorithms include agglomerative, divisive clustering. Density-based clustering such as DBSCAN. DBSCAN can be considered a perfect algorithm for the detection of outliers. Algorithms like K-Means Clustering lack the property and have clusters that are very sensitive to outliers. The principle of DBSCAN is that if any data point is nearer to many points of data in a cluster, then that point is considered as a part of that cluster. It considers vitally two parameters viz., minPts and ε for the determination of the minimum distance. The distance which specifies the neighbors is ε. Any two points of data are considered as neighbors only when the distance between them is either equal or less than that of ε. The minimum number of points of data for defining the cluster is minPts. Taking into consideration of these two vital parameters the points of data are classified as core point, border point, and noise point as shown in the Fig.2.

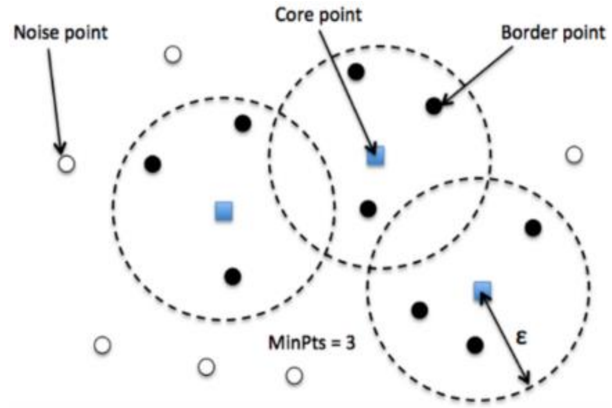


Fig. 2. Determination of the parameter minPts in the DBSCAN algorithm [12]

The core point is where there are at least MinPts that have the point inside in its surroundings with radius eps. The border point is where it is accessible from a core point and if there are less than actual required MinPts inside in its surrounding area. An outlier is a point that is not a core point and is not reached by any other core points.

A network is mathematically defined as G (N, E) where N is the number of nodes and E is the number of edges

$$E \text{ € } \{ \{e1, e2\} \mid e1, e2 \text{ € } N \text{ and } e1 \neq e2\} \qquad (1)$$

A community is defined as a cluster of nodes N where the connections are dense and these nodes are connected by edges E. The authors proposed a novel approach for the identification of the research community based on similarity indexes for assessing research interest is available in [14]. That, the key approach DBSCAN social network is elaborated in further sections.

DBSCAN starts with an arbitrary Node or Edge that hasn't been visited then its neighborhood information is a rescue from the ε parameter. If it contains MinPts within the eps neighborhood, community formation starts. Otherwise, the aim is labeled as noise. The above process continues until the density-connected cluster is found. The approach of DBSCAN is used in three different ways such as Perform DBSCAN to detect noise points. Perform DBSCAN to remove edges that are marked as noise. Perform DBSCAN to remove nodes that are marked as noise. The Output of the DBSCAN algorithm depends on values on MinPts. The optimal epsilon value is found using [8]. Varying the MinPts helps us in detecting communities.

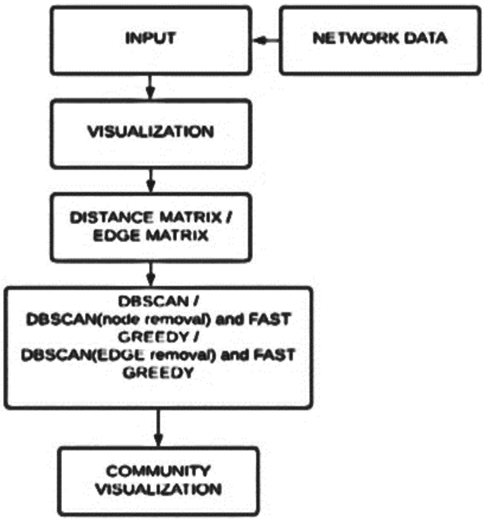

INPUT
NETWORK DATA
VISUALIZATION
DISTANCE MATRIX / EDGE MATRIX
DBSCAN / DBSCAN(node removal) and FAST GREEDY / DBSCAN(EDGE removal) and FAST GREEDY
COMMUNITY VISUALIZATION

Fig. 3. Block diagram of proposed community detection framework

Algorithm 1: Proposed DBSCAN

```
1. Import the Libraries
2. Load the Dataset
3. Perform DBSCAN to detect communities and noise points
4. for each point p in the dataset do
5.     if p is equal to noise, then
6.         remove p
7. Perform Fast Greedy on the new data
8. Visualizing newly detected communities
   End
```

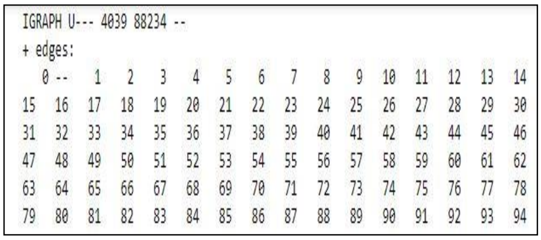
IV. RESULTS and DISCUSSIONS

A. Experimental set-up

Stanford Facebook survey network dataset is used in this paper as shown in Fig.3. Initially, the network is visualized. The edge data is converted into a distance matrix and an edge matrix. DBSCAN is performed on the matrix. The outliers are removed from the network. The algorithm Fast Greedy is applied to the network after removing outliers. Finally, the communities formed are visualized. The communities are evaluated using a modularity score.

B. Social Network Data set

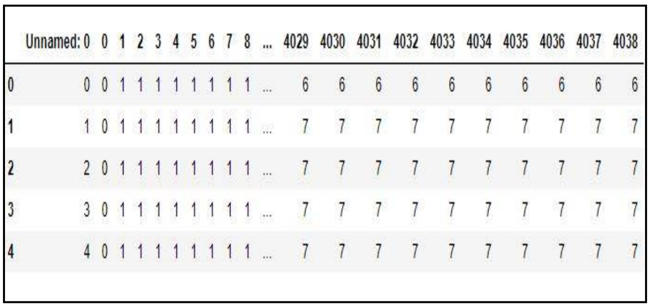
Facebook data was collected from survey participants using the Facebook app. This dataset consists of 4038 nodes and 88234 edges and the dataset includes node features, circles, and ego-networks, Facebook has been replaced with a new value for each user. The sample input is shown in Fig.4.



```
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
```

Fig. 4. Sample Input Data

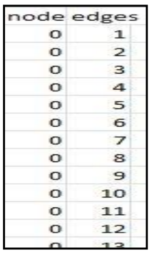
The distance matrix on taken input by using Euclidian distance is shown in Fig. 5.



	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

Fig. 5. Distance Matrix

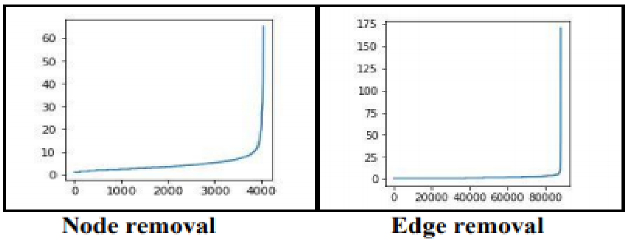
The edge matrix is shown in Fig. 6. It gives the weightage of each edge among nodes.



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

Fig. 6. Edge Matrix

Epsilon Value: The optimal epsilon value will be found at the maximum point of curvature. The optimal epsilon values taken in DBSCAN are shown in Fig. 7.



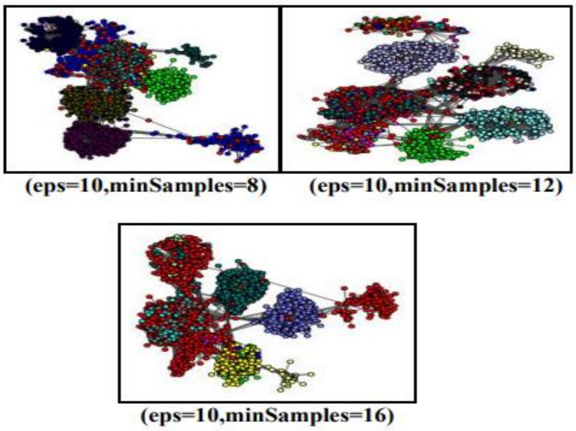
Node removal

Edge removal

Fig. 7. Optimal epsilon value for DBSCAN

C. Communities Detected Using DBSCAN

The red color nodes are marked as Noise by DBSCAN. The data were evaluated for different *MinPts.* These are the communities detected by DBSCAN shown in Fig.8.



(eps=10,minSamples=8)

(eps=10,minSamples=12)

(eps=10,minSamples=16)

Fig. 8. Communities Detected Using DBSCAN

Figure 8 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. ANALYSIS OF DBSCAN WITH EPS =10

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

Table 1 shows the analysis of the DBSCAN algorithm in community detection with variant minimum points is shown. It gives various parameters response with respect to the epsilon points.

D. Analysis of DBSCAN variants

There is no ground truth data available for the dataset. We used the modularity score to evaluate how well the clusters are formed. Modularity is an estimation of the formation of networks and graphs which calculate the strength of the separation of a network into modules. Networks with high modularity have good connections between the nodes inside the modules but rare connections between nodes in different modules. Table 2 gives the comparison of variants of the DBSCAN algorithm with node and edge removal. The results are showing with the removal of noisy edges prominent clusters are found as 11 with high density.

TABLE 2. ANALYSIS OF DBSCAN VARIANTS

Algorithm	**Clusters**	**Modularity**	**Noise Points**	**Edges 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

V. CONCLUSIONS

This proposed community detection using different DBSCAN approaches with Fast Greedy shows high performance. Community detection of Facebook - ego network is successfully performed using different DBSCAN approaches with Fast Greedy. Our approach was able to detect communities quickly and efficiently. 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 that do not have dense connections.

REFERENCES

[1] Despalatovic, L., Vojkovic, T., & Vukicevic, D. (2014). Community structure in networks: Girvan-Newman algorithm improvement. 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

[3] 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.

[4] 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.

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

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

[7] 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.

[8] 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

[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] 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.

[11] Su, RQ., Lai, YC., Wang, X. et al. Uncovering hidden nodes in complex networks in the presence of noise. Sci Rep 4, 3944 (2014). https://doi.org/10.1038/srep03944

[12] https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html

[13] Akkineni, Haritha, Myneni Madhu Bala, Venkatasuneetha Takellapati, Madhuri Nallamothu, and Suresh Yadlapati. "Measuring Research Interest Similarity Among Authors Using Community Detection." *Journal of theoretical and applied information technology*, 100, No. 11 (2022).