Implementation of Community Detection using DBSCAN Algorithm

Myneni Madhu Bala[1], M Sai Manikanta[2], R Sai Keerthana[3]

Department of Computer Science and Engineering
Institute of Aeronautical Engineering, Hyderabad – 500043, INDIA
m.madhubala@iare.ac.in

Abstract.

In the present world, online social networks are rich in multimodal data sources includes various objects, URLs, and comments. which are real time dynamic sources for analyzing and that lead to discovering facts and hidden relationship among the groups in networks. Finding a closed community in social networks is an existing and attractive area for the researchers. Broadly, these algorithms are based on dynamic networks as humans as key players and the other one is the graph structure similar to the topological structure. These two are being used for past years but these algorithms and structures have their own limitations. This article aims to overcome the said limitations by using a Density-based clustering technique called DBSCAN to detect communities. Detection and deletion of these noisy nodes in the communities lead to the development of the quality. The comparison of the experimental results will assure the removal of noisy nodes will impact the quality of the community detection. This is done by using the fast greedy approach by fusing with DBSCAN.

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 to each other users in the network forms a community. Detecting such communities are hard. 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 the Grivan Newman algorithm also known as Edge Betweenness, Fast Greedy, Lable propagation, Louvain, Walktrap, and Infomap. Neither one of these algorithms are apt to identify the noise, as nodes are not the members of the community so as to manage the problem DBSCAN algorithm is relevant since they provide the best to leave specious connected nodes such as noise, out of the detected community.

This Density-Based dimensional Collection of Applications with Noise (DBSCAN) algorithm has been taken for detecting the outliers(noise) and this detects the com-

munities 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 leads to the development of the quality.

The graph has a prevalent DBSCAN algorithm where it has a density-based approach that is used for community detection. Likewise, to DBSCAN, it has two variables, the density-based level is eps and a lower bound is MinPts for the number of nodes 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 are appeared in the past, but only some among them are extensive algorithms that are relevant in social media graphs.

The [1] Girvan–Newman algorithm detects communities by separating edges from the network. The components of the remaining network are the communities. We need to compose that reveal that the edges are the central communities, this algorithm has edges that are likely "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. If the network has loosely connected communities by an inter-group edge, then the shortest paths between different communities should go through with one of these edges. 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 the betweenness of all edges in the network are calculated. The edge with the highest betweenness is removed first. Then the betweenness of all other edges are afflicted then the removal is calculated again. Steps 3 and 4 are repeated until no edges remain. This shows that the least one of the remaining edges between the two communities will always have sustainable value. By the end of the algorithm, the result is the dendrogram. As the Girvan–Newman algorithm runs, the dendrogram is produced from the top down.

Fast greedy algorithm [2] the problem-solving which make it as the 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 either an applicant can be used to have a result that are 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.

Label Propagation algorithm [3] is which that allocate labels to the unlabeled nodes by propagating labels through the different kind 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 condition. The solutions are reduced when some nodes are given with preceding labels and 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 and it is divided into two phases: Modularity Optimization, Community Aggregation, After the beginning step is done then 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 the 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 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 [9] the authors Madhu Bala Myneni, Rohit Dandamudi 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 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 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 data points that are similar and these data points are assembled. So, clustering algorithms look for likeness or alikeness among data points. There are different algorithms to perform clustering which are Partition-based clustering such as k-means, k-median Hierarchical clustering such as Agglomerative, Divisive Density-based clustering such as DBSCAN. DBSCAN algorithm makes it a perfect fit for outlier detection. Algorithms like K-Means Clustering lack the property and have clusters that are very sensitive to outliers.

DBSCAN belongs to a cluster and is close to many points from that cluster so there are two variables of DBSCAN eps it has the distance that specifies the neighbors. There are two things that are to be considered to be neighbors i.e if the distance between them are less than or equal to eps. MinPts has the minimum number of data points, which are based on two variables, these points are classified as core point, border point, and noise point.

The core point is where if there are at least MinPts that has the point inside in its surrounding with radius eps. 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. Outlier is a point that is not a core point and 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

$$E \in \{ \{e1,e2\} \mid e1,e2 \in N \text{ and } e1 \neq e2\}.$$

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 neighborhood information is a rescue from the eps parameter. If it contains MinPts within eps neighborhood, 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 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.

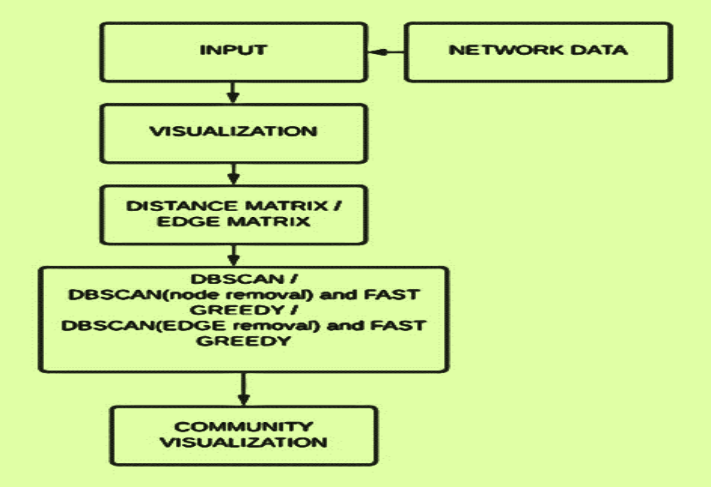

Fig. 1. Block diagram of proposed community detection framework

Algorithm 1: Proposed DBSCAN

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

IV. RESULTS and DISCUSSIONS

A. *Experimental set-up*

Stanford Facebook survey network dataset is used in this paper. Initially, the network is visualized. The edge data is converted into a distance matrix and 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-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
```

Fig. 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

Fig. 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

Fig. 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 are shown in figure 5.

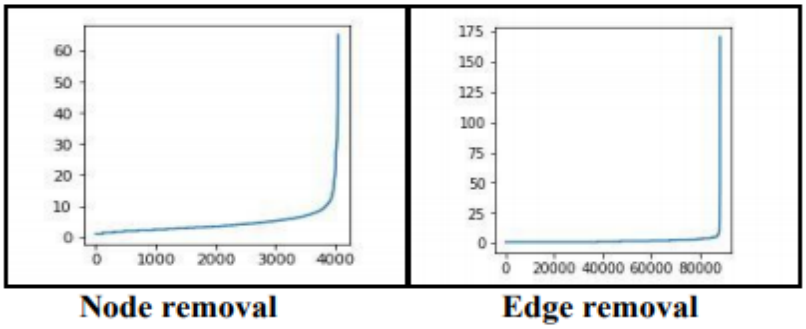

Fig. 5. Optimal epsilon value for DBSCAN

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.

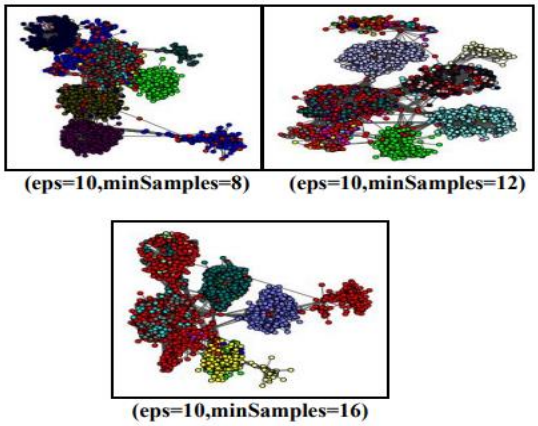

Fig. 6. Communities Detected Using DBSCAN

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. 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 4 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**	**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

ACKNOWLEDGEMENT

The authors greatly acknowledge the computational facility created in the college under DST's FIST program (SR/FST/College-2017/28(c)) laboratory. Which helps them to carry out the work. The authors thank the management of IARE for their support and kind encouragement. This document is prepared in support of Applied Computer Technology, a research-oriented technology-based company.

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 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] 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

[9] 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.