International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Effective Way of Detecting Communities in Social Network

Mehjabin Khatoon[1]* **Wahab Aisha Banu[1]**

[1]*Department of Computer Science and Engineering,*
B. S. Abdur Rahman Crescent Institute of Science & Technology, Chennai, Tamil Nadu, India
* Corresponding author's Email: mehjabinkhatoon@gmail.com

Abstract: Social networking websites have become an easiest way to make the common people thoughts and reviews to become public. Among those websites, Twitter data's are in boom, because of heavy interests of people to update their information in that website. Detection of communities for Twitter data has already been done by the other authors, but still communities detected with high strength or quality are lagging behind. In this paper, the data collected from Twitter have gone through sentiment analysis and the final scores of that analysis have been used for the plotting of the graph which acts as an input to the community detection algorithm. The twitter data's communities were detected with the detection of noise too, and upon removal of those noisy data, the strength of the detected communities used to get increase. The detection of the outliers or noise has been done with the help of DBSCAN algorithm and the communities have been detected by Newman Girvan algorithm. In this study the proposed sentiment analysis algorithm and the community detection technique have been successfully implemented and evaluated. The results from the collected data sets from Twitter have shown the communities, which were properly detected with the help of the proposed methodology. The communities were actually grouped according the sentiment scores derived and the number of words, for each tweets. Each community shows the connection according the high and low sentiment scores. The quality of the detected communities has been measured by centrality, modularity and conductance and has been compared with four other community detection algorithms i.e. with Louvain, Walktrap, Leading Eigenvector and Fast Greedy algorithms. The results were positive in maximum times when compared on the basis of the considered metrics with the other community detection algorithms.

Keywords: Social network, Community structure, Community detection, Outliers.

1. Introduction

From a decade, social media has been in boom for the uploading of day to day ongoing information. The information content for a particular topic in the social networking websites, such as in Twitter, can be extracted and can be arranged in a network which contains the hidden communities, i.e. communities of people who have the similar thoughts and interests. The networks formed from the social network data have two levels i.e. (i) microscopic- i.e. the node level properties in a network (ii) macroscopic – i.e. the properties which are global, for example the distance of the network [1]. The social networks are also called as complex networks and the research for finding the communities in social network has been in highly focused. The community in a network can be defined as the group of vertices when arranged in a much closed manner in comparison with its sparse neighbourhood. The intermediate category between microscopic and macroscopic category is called as mesoscopic category, and the detection of communities in complex networks falls under mesoscopic category since the communities are very tricky to determine [1]. In a graph, the detection of communities is NP-complete [2] and the detection of the sub-graph from the given graph with a specific property is also NP-complete [3].

The community detection method is actually a process to determine the community structure- which are connected densely between the groups of the nodes and are also sparser between the connections. The quality of the detected communities can be measured by several metrics such as modularity, conductance and centrality. The quality of the detected communities can also be increased by removing some unnecessary nodes which are known as “outliers”, also called as “noisy nodes”. Several methods have been proposed already for the detection of “outliers”. Outliers are the data which are consider being as the inconsistent data when compared with the rest of the data. The outliers’ data can be of different types like noisy or unusual information, abnormal, novel, new [4]. The detection of outliers is significant in many fields [5]. There are many outlier detection algorithms, like DBSCAN, BIRCH, ROCK, STING, Wave Cluster and these algorithms main work is to detect clusters with outliers i.e. noise, in the perspective of clustering them[5].

The method we have approached for the detection of communities with outliers, for the collected data, is DBSCAN clustering algorithm with Newman Girvan algorithm i.e. the community detection algorithm. The data’s have been extracted from one of the famous social networking website i.e. Twitter. The detection of communities of the real time data sets have been done previously but the quality of the detected communities has not been focussed much. Thus to solve this previous issue, our proposed work have been focussed on the detection of outliers with the deletion of those outliers too. The proposed methodology consists of mainly four stages:

(1) In the first stage proposed sentiment analysis (SA) algorithm has been applied on the collected Twitter data sets. The sentiment scores of the data sets were derived by using the SA algorithm and have been used for the plotting of the graphs which acts as an input to the method for detecting communities.
(2) In the second stage the density based spatial clustering of applications with noise (DBSCAN) algorithm has been applied in the input graphs.
(3) In the third stage the outliers or the noisy nodes detected on the formed groups or the clusters, from the previous stage were deleted.
(4) In the final stage communities were formed after applying the Newman Girvan (NG) algorithm.

The data sets that have been collected and used for the implementation from the Twitter are the complex networks. The data sets are following the rules of complex network with the two properties of complex network, i.e. power law and clustering coefficient [1]. The complex networks in general, will follow the power law degree distribution and high clustering coefficient. The detected communities from the implementation done in this research work has been compared with the other community detection algorithms i.e. with Leading Eigenvector, Fast Greedy, Louvain and Walktrap community detection algorithms.

The remaining sections of this research article have been given below. Section 2 is the literature survey which has been done related to this paper work. Section 3 is about the proposed methodology that has been followed to the task of detecting proper communities. Section 4 is about the experimental analysis of the accomplished research work for this paper. The last section 5 is about the conclusion and future work of this paper.

2. Related work

Singh Vijendra and Pathak Shivani in 2014 have given a description about an approach, in which in the pre-processing step they have used Univariate Outlier detection and to analyze the outliers’ effect for the analysis of the clusters of the dataset they have used K-means algorithm [4].The authors have to consider more number of data sets to use the proposed approach for detecting the outliers. The author has used only one type of data set. The K-means algorithm is having one disadvantage that number of clusters should be given prior to the clustering process.

The detection of outliers is significant in many fields. Sheng-yi Jiang, and Qing-bo An, in 2008 presented a Clustering Based Outlier Detection (CBOD) method. The CBOD method consists of two stages, i.e. in the first stage it uses one-pass clustering algorithm by using the cluster dataset and in the second stage outlier factors have been used for determining the outlier cluster [5]. The labelling of the outliers should be more accurate and should have also considered the more real time data sets of any social networking websites.

In the field of detecting outliers DBSCAN is a powerful algorithm for clustering on the basis of density, but some difficulty lies in detecting its parameters value i.e. minpts and epsilon. Tran Manh Thang and Juntae Kim in 2011 have proposed a new way to detect and apply the parameters in DBSCAN [6]. The new type of algorithm has been named as DBSCAN-MP and in this algorithm every cluster may different values for epsilon and minpts. The results of the DBSCAN-MP algorithm should compared with other clustering algorithms based on more number of functions other than the false positive rate.

Till now different authors have used different clustering algorithms for the detection of outliers [5], and have also proved that some methods of detecting outliers have been very advantageous and proves to be better than the other methods in the field of detecting and cleaning outliers like Antonio Loureiro in 2004 have proposed a method for the detection of outliers using hierarchical clustering algorithm [7].The work in Antonio Loureiro paper is to detect the erroneous foreign trade transactions and is tested on the cleaning of official statistics data.The work can be tested by increasing the cluster size more than the considered size i.e. in this case it is 5.

Yomna M. ElBarawy in 2014 have implemented the DBSCAN clustering algorithm for detecting the communities from the real time data sets and the results represents the core which have high influence, borders which have low influence and the outliers which don’t have any influence [8]. The deletion of these outliers’ nodes will make the datasets, noise free to deal with. However the detected communities’ quality should also be determined.

In the field of text mining, sentiment analysis is the evolving field of research [11]. The main motive of sentiment analysis is to digitize the expressions and emotions of individuals [12]. Fazal Masud Kundiin in 2014 has presented a lexicon based framework for the classification of tweets into positive, negative or neutral sentiments. Slang words present in the tweets also gets detected and gets scores with the help of this framework. The more number of data sets need to be considering for this work.

Shri Bharathi in 2017 [25] proposed an approach in which correlation between the sentiments of Tweets , Really Simple Syndication (RSS) news feeds and stock market values were determined for doing the stock market prediction with high precision. This work focuses only on the type of words i.e. in nouns, verbs, adjectives by using a Part-of-Speech tagger. It should also consider the positivity or negativity of the words.

Haritha Akkineni in 2017 [26] worked on developing a hybrid method to create a readable summaries of the tweets with the help of classified positive and negative tweets, which acts as an input to the hybrid method. Instead of finding the semantically similar words by its root words, focus should be on different possible types of slang words.

3. Proposed methodology

The first thing to achieve the task of detecting communities from Twitter data is to collect the data from the social network website i.e. Twitter. The data collected then digitized with sentiment analysis scores by the proposed Sentiment Analysis (SA) algorithm. The sentences of tweets were fully analyzed including the slang words but except the stop words. The computation for deriving the sentiment scores was done using the proposed sentiment analysis algorithm for this methodology. There are four databases that have been collected from the web i.e. the databases of the positive words, negative words, positive slang words, negative slang words. The words of databases have been matched with the words of tweets sentences and the tweets sentiment scores were calculated according to the sentiment analysis algorithm. The implementation for SA algorithm has been done using JAVA programming language. The detection of noise free communities has been achieved by using the proposed DBSCAN & Newman Girvan (DBNG) algorithm. The implementation of DBNG algorithm has been done using R tool.

3.1 Proposed sentiment analysis algorithm

```
Input: Tweets
Calculation of Sentiment scores:
  If the word is a positive word then add to tweet score of +1
  Else If the word is a negative word then add to tweet score of -1
  Else If  the word is a positive slang word then add to tweet score of +1
  Else If the word is a negative slang word then add to tweet score of -1
  Else If the word is a positive word and contains a capital letter then add to tweet score of +2
  Else If the word is a negative word and contains a capital letter then add to tweet score of -2
  Else If the word is a positive word and contains repetition of any letter then add to tweet score of +2
  Else If the word is a negative word and contains repetition of any letter then add to tweet score of -2
Tweet_score = Tweet_score + score

Output: Tweets Score.
```

The tweets score which are actually the sentiment scores i.e. people sentiments regarding any event, product, movie etc. Size for the collected tweets has also been calculated by counting the number of words. The input table for plotting the graph is then prepared with two columns i.e. first

Table 1. Tweets size and tweets score for few collected data from DS1.

Sl. No.	Tweets size	Tweets score
1.	20	9
2.	15	0
3.	19	1
4.	12	0
5.	21	2
6.	23	3
7.	33	1
8.	22	1
9.	23	3
10.	18	1
11.	24	3
12.	26	1
13.	26	-4
14.	35	1
15.	15	0

column with tweets size and second column with tweets score. The table has been arranged in excel sheets for every collected data set. Table 1, has been shown for few collected data only from DS1 ,in which tweets size is the first column and tweets score is in the second column.

3.2 Proposed DBSCAN & Newman Girvan (DBNG) algorithm

The approach that has been followed for the detection of the noise free communities for the Twitter data is DBSCAN (Density Based Spatial Clustering of Applications with Noise) clustering algorithm with Newman Girvan community detection algorithm which we have named it as DBSCAN & Newman Girvan (DBNG) algorithm. DBSCAN is a clustering algorithm which is used for detecting clusters with outliers also and it can be applied on large datasets [9]. DBSCAN have several advantages in comparison with other clustering algorithms:

(1) In DBSCAN it's not required to mention the number of clusters in prior, to be formed and it can handle the outliers or noisy data [10].
(2) DBSCAN algorithm depends on the value of the epsilon i.e. the radius and the minpts i.e. the minimum number of points.
(3) The value of the radius and the minpts should be specified by the user itself.

The DBSCAN algorithm results in three types of data i.e. the core, border and noise points [13]. The core points are those nodes which should be lying within the given value of epsilon i.e. radius which has been specified by the user and by considering the minimum number of points i.e. minpts which has also been specified by the user. The border points are those points which used to fall on the neighbours of the several core points. The outlier points are those points which neither falls in core point category nor in border point category.

There exists so many other community detection algorithms but the reasons for selecting Newman Girvan algorithm are:

(1) The communities detected are stronger compared to other community detection algorithms, which has been proved in our previous research work.
(2) The Newman Girvan algorithm [14] depends on the "edge betweenness" factor which determines the all of the shortest paths that exists between a pair of nodes.

The DBSCAN & Newman Girvan (DBNG) algorithm takes input as a graph that has been plotted after applying the sentiment analysis algorithm. The inputs for the graph to be formed have been taken from the Twitter data sets that have been discussed in the experimental analysis section. The outliers and the clusters can be visualized after the DBSCAN algorithm is executed after which we can get the number of seed nodes, the border points and the number of outliers. The number of outliers can be more or less according to the epsilon value and the minpts value. The outliers are deleted from the graph, after which it's (i.e. of graph) quality can be determined through various metric functions, i.e. modularity, centrality, conductance. The Newman Girvan (NG) algorithm has been applied at last for the formation of communities. The DBNG algorithm complexity is *O(mn)* where as *m* is the number of edges and *n* is the number of nodes.

3.2.1. Steps of the DBNG algorithm

(1) First the graph formed from the input data sets should be given as an input with the value of the epsilon i.e. the radius whose value should be considered to connect the core points and the minpts i.e. the minimum number of points to form the clusters.
(2) Edges should be added between each pair of core points.
(3) The outlier points are marked according to the distance from the core points.
(4) Clusters used to be formed from each group of connected core points.
(5) Border points are assigned arbitrarily with its associated core points in its clusters.
(6) The clusters are formed in the full graph, according to the previous steps of the algorithm.
(7) The marked outlier points are then deleted from the whole graph.

DBSCAN & Newman Girvan (DBNG) Algorithm

```
Formation of the communities
Input:
g(V,E)                // graph formed from the collected data set
 MinPts , Eps         // MinPts is the minimum number of points and Eps is the epsilon (i.e. radius)

Output:
C                     // Formed communities with the set of nodes

Procedure:
Eps ← give epsilon value for the graph to be formed
MinPts ← give the minimum number of points for the graph to be formed
while (length(g))
{
cp ← add edges            //add an edge between each pair of core points
np ← mark noise points
cluster ← make clusters from each group of connected core points
bop ← arbitrarily border points assignments to the cluster which contains its associated core points
}

CG ← Formation of the clustered graph
CG ← CG - np     // deletion of the noise points (np) from the cluster
C ← Formation of communities after applying Newman Girvan algorithm
```

(8) Finally the communities are formed after applying the Newman Girvan algorithm. This algorithm works with the calculation of the edge betweenness, and then deleting the edge with high betweenness. After deletion betweenness score is recalculated for all the edges i.e. affected by the deletion and this process continues till it covers all the edges.

4. Experimental analysis

The data sets have been collected from the social network website i.e. Twitter for carrying out the implementations work and every data sets consists either of 100 or 200 number of data's . The tweets are actually the reviews given for a particular movie, a newly launched phone. There are total number of four data sets and these are (i) DATA SET 1(DS1): It is a set of tweets of an Indian movie named Baahubali, (ii) DATA SET 2(DS2): It is set of tweets for newly launched iphone7 mobile phone, (iii) DATA SET 3(DS3): It is set of tweets for newly launched MiA1 mobile phone, (iv) DATA SET 4(DS4): It is set of tweets of the review given for that Gst council meet ,which was conducted because of the imposed Gst rates by government. First the tweets collected were digitized by calculating its tweet score with the help of proposed Sentiment Analysis algorithm in the previous section. The tweets size have been calculated by calculating the number of words and then the table for each data set have been arranged by including the values of tweets size and tweets score. The graphs have been plotted from the table and then the DBNG algorithm was applied in which the outlier nodes from the plotted graph have been deleted. Then finally we got the noise free communities from the DBNG algorithm. The networks of the communities formed have been simplified because of which the multiple connections between the nodes have been deleted. As we have already discussed in the previous sections that the table formed from the collected data set consists of two columns - one for the tweets size and another of the tweets score formed from the SA algorithm, so the detected communities consists of two types of nodes one of the tweets size and another for the tweets score.

4.1 Graphs plotted and the proof of its complex network

The graphs formed from the collected data sets follows the property of complex network i.e. the network follows the property of power law degree distribution and high clustering coefficient [1].In this paper we have derived the proof of networks for all the four data sets which follows the property of the complex network i.e. power law and clustering coefficient. The distributions of power law are the decaying probability tail exponentially and its occurrence loosely represents the involvement of large values with a non-negligible probability [15].

In an undirected graph G, let for a randomly selected vertex has a probability $P_k$and k is its degree. The scale free property of graph G is proved to occur if it is having its node-degree distribution P_k as heavy tailed-if,

$$P_k \sim C_k^{-\alpha} \qquad (1)$$

Where, C > 0 is a constant and α ∈ (0, 2).

Clustering coefficient or network transitivity is also one of the properties of the complex networks. In social networks the two vertices are common neighbors when they are connected to a common third vertex and these two vertexes will know each other more than other vertex because of their common contacts. Clustering coefficient or network transitivity has measured this problem from the below Eq. (2):

$$C = \frac{3 \times (number\ of\ triangles\ on\ the\ graph)}{(number\ of\ connected\ triples\ of\ vertices)} \qquad (2)$$

The clustering coefficient or network transitivity value used to be 1 for the connected graphs and other than this for real-world networks it usually used to be between 0.1 to 0.5 [16].In this paper we have mentioned the details regarding the power law and clustering coefficient for two data sets i.e. for DS1 and DS2, for the graphs plotted from the collected datasets .The graphs of the DS1 and DS2 has been shown in Figs.2 and 4. The plots of the power law and the transitivity values are the proof for the networks to be complex networks. The transitivity value for DS1 is 0.01287 and the plot for the power law degree distribution has been shown in Fig. 3 for which the value of α is 0.572. The transitivity value for the DS2 is 0.01060071 and the plot for the power law degree distribution has been shown in Fig. 5 for which the value of α is 0.604.

4.2 The effects of using DBSCAN algorithm

The DBSCAN algorithm results after applying different epsilon values for the input graph for the DBNG algorithm. The DBSCAN algorithm results in clusters which has three types of nodes i.e. core, border and outliers, as we have already discussed in the previous section. The number of clusters formation and the quantity of outlier nodes used to get decrease by increasing the value of epsilon. The value of core nodes used to get decrease by increasing the value of epsilon. The effects of DBSCAN algorithm has been shown in table number 2 and 3 for the Data Set 1 and Data Set 3.

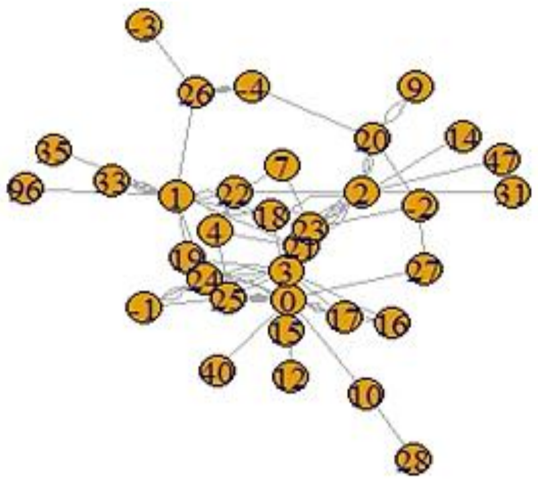

Figure.2 Graph formed from DS1

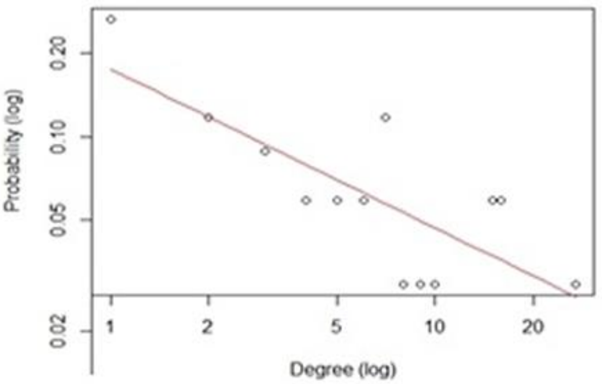

Figure.3 Plot for power law degree distribution for DS1 with α=0.572

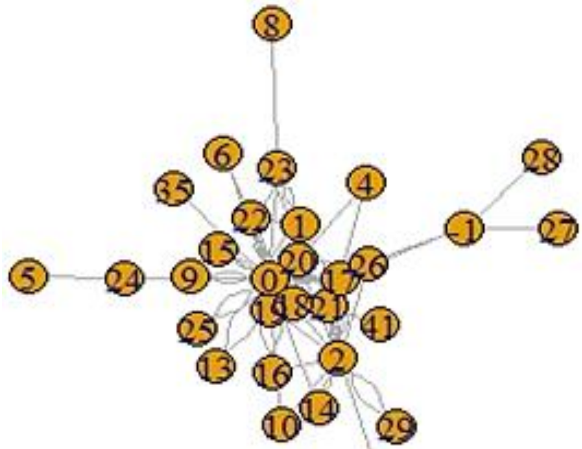

Figure.4 Graph formed from the DS2

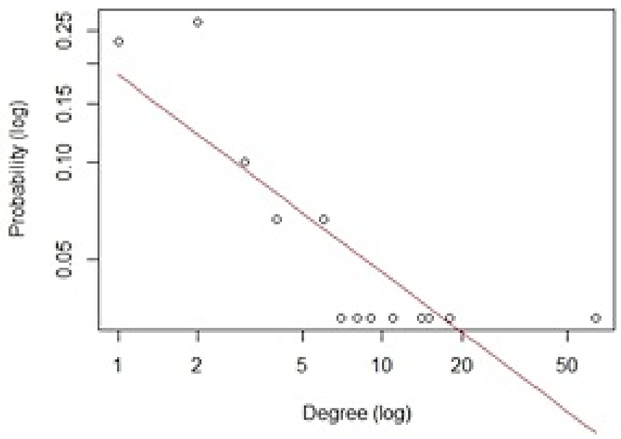

Figure.5 Plot for power law degree distribution for DS2 with α=0.604

Table 2. The effects of DBSCAN clustering algorithm in DS1

Epsilon	No. of clusters	Core	Border	Outliers
1	6	16	18	18
2	2	30	4	4
3	1	34	0	0

Table 3. The effects of DBSCAN clustering algorithm in DS3

Epsilon	No. of clusters	Core	Border	Outliers
1	2	16	11	11
2	2	25	2	2
3	2	26	1	1
4	1	27	0	0

4.3 Scoring functions

Modularity: The quality of a particular division of a network can be measured by *modularity,* which was proposed by Newman and Girvan in 2003[14]. Using modularity the community structure of a network can be determined, i.e. the statistical arrangement of edges in a graph [17]. Modularity measure can be quantified by the equation no. (3) in which $\sum_i e_{ii}$ determines the fraction of edges in the network which has been connected in between the vertices in the same community. The $a_i = \sum_j e_{ij}$ represents the sums of row (or column) which determines the fraction of edges that connect to vertices in community "i". We will have $e_{ij} = a_i a_j$ when edges falls in between the nodes without regarding the communities they belong to. The value of modularity lies between 0 & 1, where as 1 indicates the strongest community structure.

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

Conductance: The fraction of the total edges which point outside a cluster can be measure by *conductance* function [17]. The conductance can be measured by using the equation number (4) in which S is the total quantity of nodes, m is the total number of edges in S and c is the total number of edges which are present in the boundary of those nodes i.e. S.

$$f(S)=c/(2m+c) \tag{4}$$

Centrality: Centrality function in the graph theory is used for the identification of the most significant vertices and edges in the network. For the edge of a graph, centrality is the degree of global sensitivity of graph distance function (i.e. a graph metric) on the weight of the edge considered [18]. The three centrality measures that we have used for measuring the centrality of the graphs are degree centrality, betweenness centrality and closeness centrality.The *degree centrality* can be defined as total quantity of ties or links which are incident upon a node. For a graph G = (V, E) which have |V| number of vertices and |E| number of edges, the degree centrality can be defined as in Eq. (5).

$$C_D(v) = deg(v) \tag{5}$$

For graph centralization, the degree centrality of a vertex can be extended to the whole graph from the vertex level [19]. In graph G, let $v*$ be the node with highest degree centrality. In a graph, X= (Y, Z) be the |Y| vertex connected graph which maximizes the quantity given in equation no. (6), in which, $y*$ is the node with highest degree centrality. The degree centralization of the graph G is given in equation no. (7). When the graph X is star graph it contains one central vertex that connects to all other vertices, then the value of H is maximized like in Eq. (8).

$$H = \sum_{j=1}^{|Y|} [C_D(y^*) - C_D(y_j)] \tag{6}$$

$$C_D(G) = \frac{\sum_{i=1}^{|V|} [C_D(v^*) - C_D(v_i)]}{H} \tag{7}$$

$$H = n^2 + 3n + 2 \tag{8}$$

The *closeness centrality*, for a vertex in a connected graph is the average distance of the shortest path between the vertex and all other vertices in the graph. Thus the value of closeness centrality of a vertex depends on its closeness to all other vertices. The closeness centrality can be derived by the Eq.

(9), in which *d(y, x)* is the distance between the nodes *x* and *y*.

$$C(x) = \frac{1}{\sum_y d(y,x)} \quad (9)$$

The *betweenness centrality* of a graph is the centrality measure of a node within a graph. Betweenness centrality calculates the number of times a node acts a bridge between two other nodes in a shortest path. It can be derived by the equation no. (10), in which, $\sigma_{st}(v)$ is the total number of shortest paths that pass through *v*, and σ_{st} is the number of shortest paths between node *s* to node *t*.

$$C_B(v) = \sum_{s \neq v \neq t \epsilon V} = \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (10)$$

Graph density: The *graph density* function has been also for determining the quality of the extracted communities. The density of the graph in a mathematical way is the quantity of edges which are closer to the maximal number of edges. While the sparse graphs are graphs which have only few edges. So, for an undirected graph, density of the graph is computed by the Eq. (11) in which D is the graph density, E is the number of edges and V is the number of vertices.

$$D = \frac{2|E|}{|V|(|V|-1)} \quad (11)$$

4.4 Other community detection algorithms

Louvain algorithm (LV): LV algorithm was proposed in 2008 by Vincent D. Blondel [20] for extracting the community structure of large networks. The algorithm is based on modularity optimization method and it is a heuristic approach. The algorithm is divided into two phases, which used to get repeated in every iteration. In the first phase every nodes was assigned in a community, so the number of community is equivalent to the number of nodes. For each node *x*, first it's (i.e. of *x*) neighbors *y* are considered, then the gain of modularity is quantified for the removal of node *x* from its community and placing it in the community of *y*. The removal of node *x* to community *y* happens only when there used to gain in the modularity, otherwise node *x* used to remain in its own community. The first phase continues till the attainment of local maxima of modularity is achieved. In the second phase it builds a new network which consists of the nodes for which communities are found during the first phase. To achieve the task of second phase it attains the weight of the links in between the two nodes by considering the two nodes in the corresponding two communities and then the sum of the weight of the links is calculated.

Walktrap algorithm (WT): WT algorithm was proposed in 2005 by Pascal Pons [21] which applies a hierarchical agglomerative clustering approach and it uses a similarity based on random walks. So for determining the community structure efficiently it uses the agglomerative algorithm. The main intuition behind the walktrap algorithm is that random walkers usually used to get trap in the densely connected areas in a network. The node-to-node distance is computed for choosing the closest communities. The construction of distance is done by the addition of the differences for all nodes, with a proper degree. Initially there is only one partition. In each iteration of this algorithm two communities are chosen based on the distance between them and a then a new partition is created.

Leading Eigenvector algorithm (LEV): LEV algorithm was proposed in 2006 by M.E.J Newman [22] which is a matrix based approach. The maximization of the modularity function is achieved using the modularity matrix. Detection of communities faces problem when nodes are clustered with a higher than average density of edges connecting them. The maximization of the modularity, for the total possible number of divisions in the whole network can be the solution to the problem. The process of community detection can be aided by the maximization process in the terms of the eigen spectrum of a matrix i.e. the modularity matrix. Thus the approach is useful for detecting the community structure of a network.

Fast Greedy algorithm (FG): FG algorithm was proposed in 2003 by M.E.J Newman [23] which is applicable to hierarchical agglomerative approach and relies on a greedy optimization method. In this algorithm initially number of communities is equivalent to the number of nodes. Then the communities will be merged gradually until it gathers all the nodes in a single community. The criterion for merging the nodes is based upon the largest increase (or smallest decrease) in modularity and in each step of merging greedy principle is applied. Since the nature of the FG algorithm is hierarchical so it results in producing a hierarchy of community structures similar to divisive approaches. The modularity values are compared for choosing the best one, while merging the communities [24].

4.5 Effects of deleting the outliers from the formed network

The networks formed from the collected data sets, have been tested using the previously mentioned functions. The outliers or the noisy nodes formed after the application of DBSCAN algorithm have been deleted. So, the resultant networks i.e. after the deletion of outliers, have been compared with the networks before the deletion of the outliers i.e. before the application of DBNG algorithm. So, the values from the tables shown can be analyzed, because the positive effect of the DBNG algorithm has been proved using the functions in the formed networks from the collected data sets. The graphs formed from the data sets, have been simplified before the application of the functions. The values we got after clearing the outliers are more than before clearing the outliers except in few cases. For the Data set 1, which is shown in table 4, the value of the average closeness centrality is less after the clearance of outliers and the number of communities is same. For the Data set 2, which is shown in table 5, the value of the three centrality measures are less, after the clearance of outliers and the number of communities are same. For the Data set 3, which is shown in table 6, the number of communities is same. For the Data set 4, which is shown in table 7, the number of communities is more after clearing the outliers. The communities formed from the data sets i.e. before and after deleting the outliers has been shown in the Figs. 6 - 13.

Table 4. The effect in DS1, after deleting the outliers

Functions	Before clearing outliers	After clearing outliers
Graph density	0.09447415	0.09469697
Modularity	0.3798505	0.409
No. of communities	6	6
Average betweenness centrality	0.3465062	0.3656507
Average degree centrality	0.2691622	0.280303
Average closeness centrality	0.3045474	0.2929738

Table 5. The effect in DS2, after deleting the outliers

Functions	Before clearing outliers	After clearing outliers
Graph density	0.1034483	0.1034483
Modularity	0.2204938	0.2293084
No. of communities	4	4
Average betweenness centrality	0.6748527	0.6693968
Average degree centrality	0.5517241	0.5394089
Average closeness centrality	0.5518728	0.5412741

Table 6. The effect in DS3, after deleting the outliers

Functions	Before clearing outliers	After clearing outliers
Graph density	0.1082621	0.1086957
Modularity	0.3822715	0.4072222
No. of communities	4	4
Average betweenness centrality	0.6367735	0.7237326
Average degree centrality	0.5071225	0.5434783
Average closeness centrality	0.4755881	0.4633228

Table 7. The effect in DS4, after deleting the outliers

Functions	Before clearing outliers	After clearing outliers
Graph density	0.1190476	0.1253561
Modularity	0.2041975	0.2678202
No. of communities	4	5
Average betweenness centrality	0.4206623	0.4300851
Average degree centrality	0.4365079	0.451567
Average closeness centrality	0.1327846	0.134729

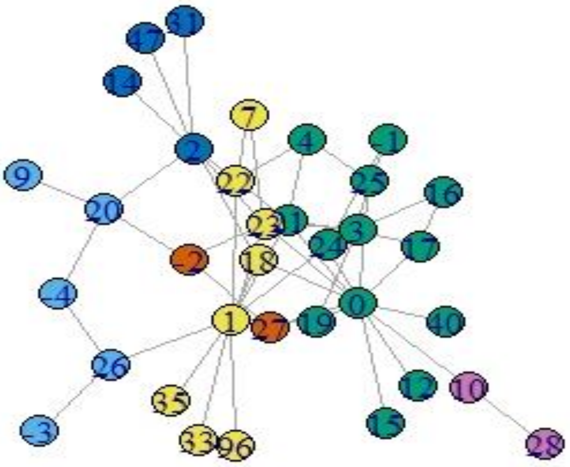

Figure.6 Communities formed before clearing outliers for DS1

Received: March 15, 2018

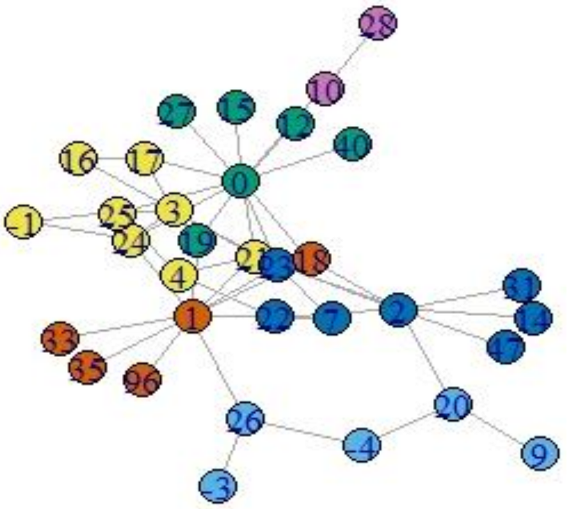

Figure.7 Communities formed after clearing outliers for DS1

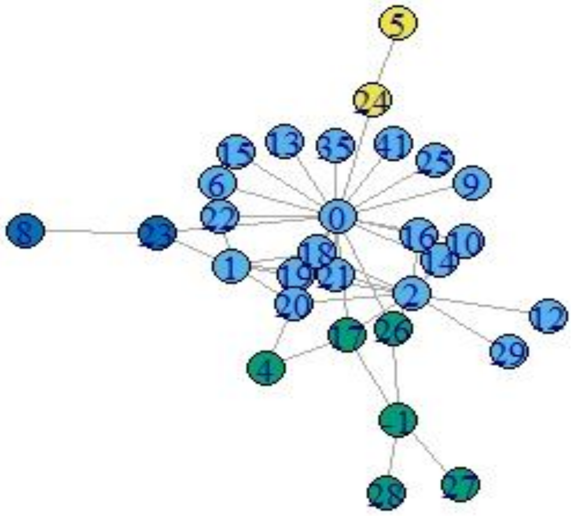

Figure.8 Communities formed before clearing outliers for DS2

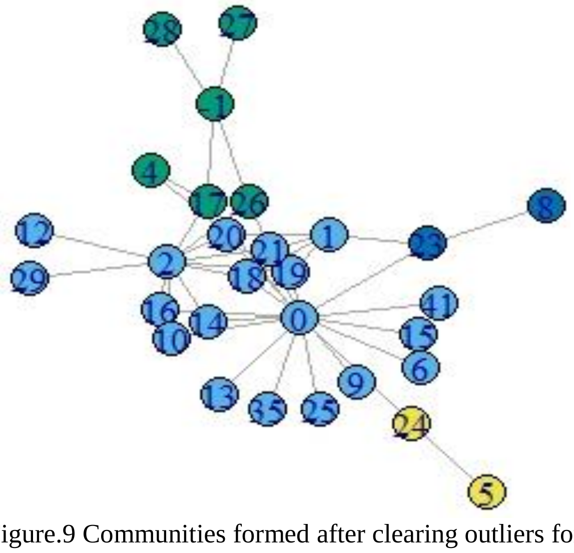

Figure.9 Communities formed after clearing outliers for DS2

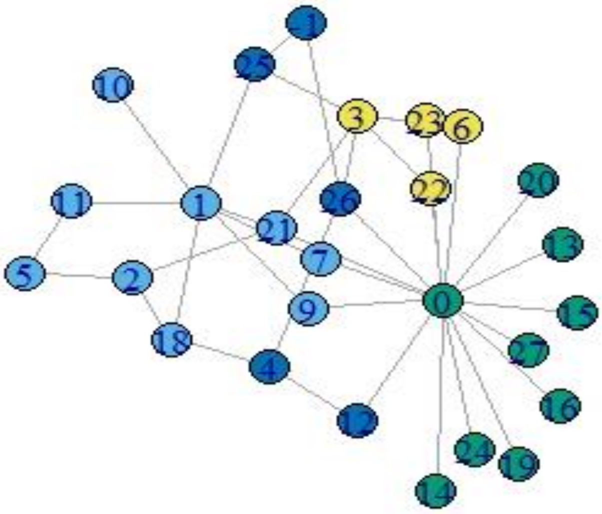

Figure.10 Communities formed before clearing outliers for DS3

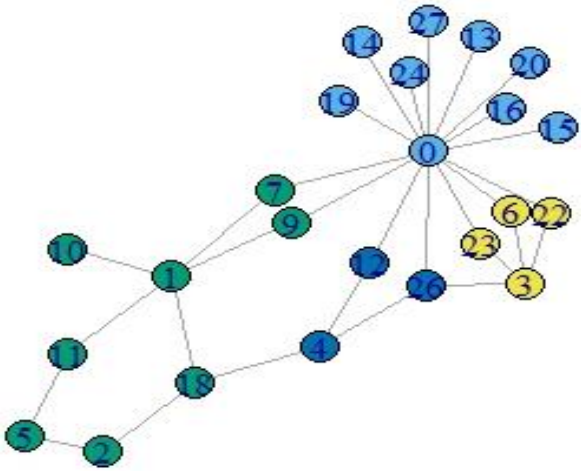

Figure.11 Communities formed after clearing outliers for DS3

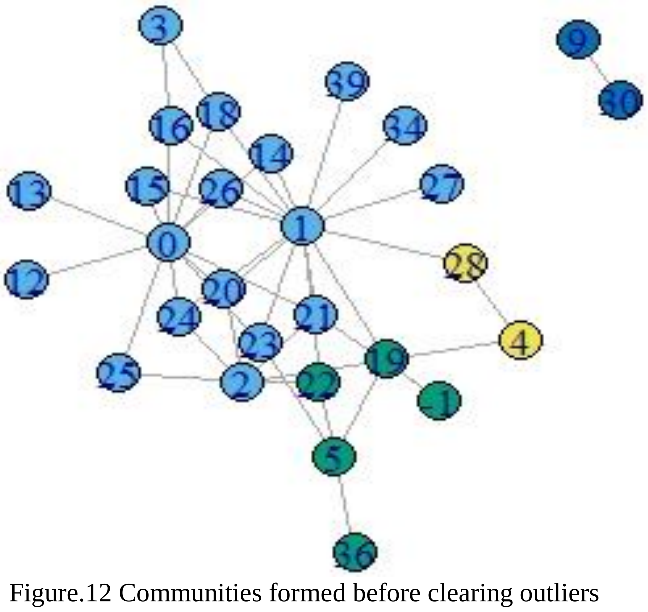

Figure.12 Communities formed before clearing outliers for DS4

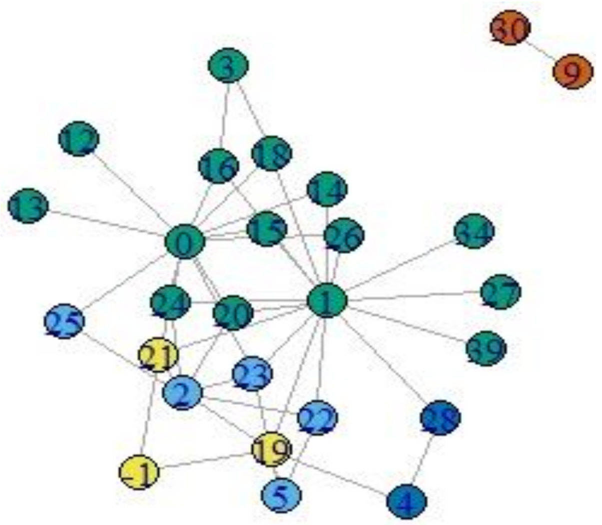

Figure.13 Communities formed after clearing outliers for DS4

4.6 Comparison with the other community detection algorithms

In the DBSCAN & Newman Girvan (DBNG) algorithm the groups formed from the input graphs after applying the DBSCAN algorithm were detected along with noise, then it deletes those noisy nodes i.e. the outliers, and then the final communities were formed using the Newman Girvan algorithm. The resultant graphs from DBNG algorithm were then compared with the graphs formed from four other community detection algorithms i.e. with Louvain [20], Walktrap [21], Leading Eigenvector [22], and Fast Greedy [23] algorithms, for the same considered data sets. The proposed approach is better than the other community detection algorithms because the communities are formed finally without the noisy nodes and removal of which makes the formed communities stronger than the communities formed from the other community detection algorithms. The communities that are formed finally after the application of the DBNG algorithm consists of the connections between the people with similar sentiments i.e. who more or less have given similar type of thoughts regarding the events, movies and newly launched products. These types of formed communities can be used to estimate the high, low and medium supporter for any products, any government policy, any event etc.

The strength of the formed communities have been determined by calculating the difference between the modularity and the conductance metrics i.e. the high modularity and low conductance values shows the strongest community structure [18]. The graph drawn in Fig. 14, in which blue line is of the DBNG algorithm results i.e. the difference between the modularity and the conductance, shows the strongest communities formed from the DBNG algorithm when compared with the other considered community detection algorithms. In the Fig. 14, horizontal x-axis has been labelled for the collected data sets and the vertical y-axis has been labelled for the strength of the communities. The modularity value which determines the quality of the formed communities from the input graphs, used to get increase after the deletion of the noisy nodes i.e. when with the graphs before the deletion of the noise which has been discussed in the section 4.5.

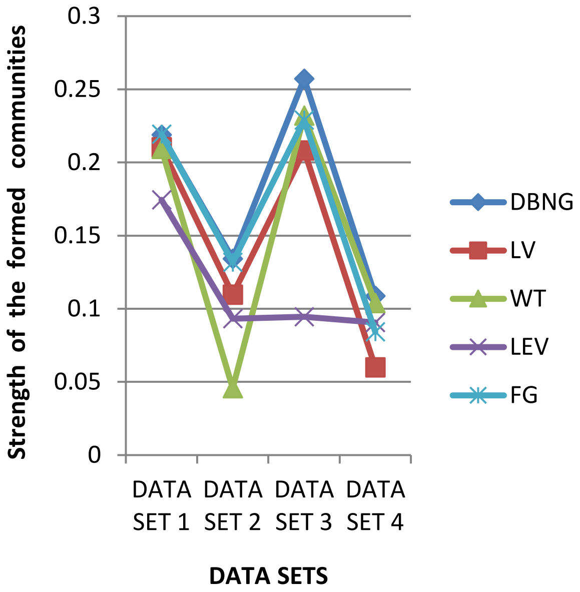

Figure. 14 Graph formed after calculating the difference between the modularity and the conductance for determining the strength of the community structure

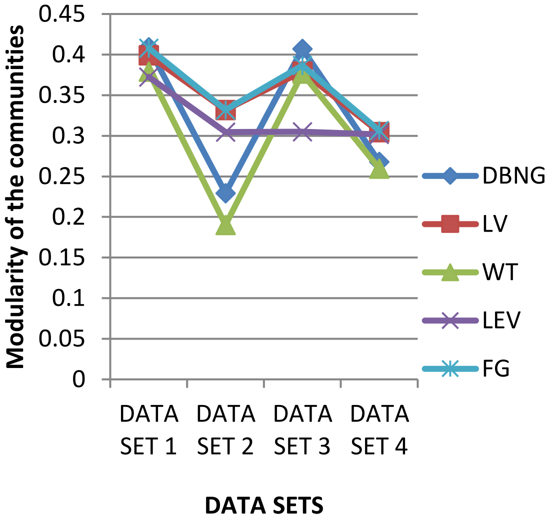

Figure. 15 Graph formed after comparing the modularity values with the other community detection algorithms

The modularity value when compared with the other community detection algorithms results higher except in few cases of DS2 and DS4 as shown in Fig. 15. In the Fig. 15, horizontal x-axis has been labelled for the collected data sets and the vertical y-axis has been labelled for the modularity of the communities.

5. Conclusion

The proposed algorithm which is the combination of DBSCAN and Newman Girvan algorithm i.e. the DBNG algorithm works well on the data sets for the purpose of detecting communities. The communities formed from the full graph of each data set, shows the connection between the similar types of tweets with similar sentiment scores and each community formed according the high and low sentiment scores. The detected communities can be used to determine the quantity high, low or medium supporters of the events, products, movies regarding which data sets were collected from Twitter. The deletion of the outliers from the graph after the DBSCAN algorithm makes the community structure more strong. The graphs before and after the deletion of outliers were also compared. The metric functions mentioned in the paper, have been applied properly on the detected communities, and the final results i.e. the formed communities have been compared with the four other communication detection algorithms i.e. Louvain, Walktrap, Leading Eigenvector and with Fast greedy algorithm. The graphs for the strength of the community structure and for the modularity values have been shown. The benchmark results shown through tables and graphs have shown positive results except in few cases, which have been discussed in the previous sections. The networks formed from the collected real time data sets follow the property of complex network, which was proven by two data sets above.

Thus the proposed SA algorithm and the DBNG algorithm have been implemented properly on the four data sets. In the future, the proposed methodology can be used for the data sets of other social networking websites i.e. other than twitter and with larger number of data sets of various types. The proposed methodology can be improved further, which can detect communities in a very less time.

References

[1] M. Vasudevan and N. Deo, "Efficient community identification in complex networks", *Social Network Analysis and Mining*, Vol.2, No.4, pp.345–359, 2012.

[2] S. Fortunato, "Community detection in graphs", *Physics Reports*, Vol.486, No.3-5, pp.75-174, 2010.

[3] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness,* W. H. Freeman and Company, ISBN: 0716710455, New York, 1979.

[4] V. Singh and S. Pathak, "Robust Outlier Detection Technique in Data Mining: A Univariate Approach", *arXiv:1406.5074v1*, 2014.

[5] S. Jiang and Q. An, "Clustering-Based Outlier Detection Method", In: *Proc. of the fifth International Conference on Fuzzy Systems and Knowledge Discovery*, pp.429-433, 2008.

[6] T. M. Thang and J. Kim, "The Anomaly Detection by Using DBSCAN Clustering with Multiple Parameters", In: *Proc. of the International Conference on Information Science and Applications*, pp.1–5, 2011.

[7] A. Loureiro, L. Torgo, and C. Soares, "Outlier Detection Using Clustering Methods: a Data Cleaning Application", In: *Proc. of the data mining for business workshop*, 2004.

[8] Y.M. ElBarawy, R.F. Mohamed, and N. I. Ghali, " Improving Social Network Community Detection Using DBSCAN Algorithm", In: *Proc. of Computer Applications & Research, 2014 World Symposium*, DOI : 10.1109/WSCAR.2014.6916792 ,2014.

[9] A. L. Mary and K. R. S. Kumar, "A Density Based Dynamic Data Clustering Algorithm based on Incremental Dataset", *Journal of Computer Science*, Vol.8, No.5, pp.656-664, 2012.

[10] S. Chakraborty, N.K. Nagwani, and L. Dey, "Performance Comparison of Incremental K-means and Incremental DBSCAN Algorithms", *International Journal of Computer Applications*, Vol.27, No.11, pp. 14-18, 2011.

[11] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey", *Ain Shams Engineering Journal*, Vol.5, No.4, pp.1093–1113, 2014.

[12] F. M. Kundi, A. Khan, S. Ahmad, and M. Z. Asghar, "Lexicon-Based Sentiment Analysis in the Social Web", *Journal of Basic and Applied Scientific Research*, Vol.4, No.6, pp.238-248, 2014.

[13] M. Hahsler, M. Piekenbrock, and D. Doran, "Dbscan: Fast Density-based Clustering with R", *https://cran.r-project.org/web/packages/dbscan/vignettes/dbscan.pdf.*

[14] M. E. J. Newman and M. Girvan, “Finding and evaluating community structure in networks”, *Physical Review E,* Vol.69, No.2, Article id: 026113, 2004 ,

[15] R. G. Clegg, C. D. C. Gilfedder, and S. Zhou, “A critical look at power law modeling of the Internet”, *Computer Communications*, Vol.33, No.3, pp. 259–268, 2010.

[16] M. Girvan and M. E. J. Newman, “Community structure in social and biological networks”, In: *Proc. of the National Academic of Sciences*, Vol.99, No.12, pp.7821–7826, 2002.

[17] J. Yang and J. Leskovec, “Defining and Evaluating Network Communities based on Ground-truth”, In: *Proc. of the 2012 IEEE International Conference on Data Mining*, pp.745-754, 2012.

[18] D. J. Klein, “Centrality measure in graphs”, *Journal of Mathematical Chemistry*, Vol.47, No.4, pp.1209–1223, 2010.

[19] L. C. Freeman, “Centrality in Social Networks Conceptual Clarification”, *Social Networks*, Vol.1, No.3, pp. 239, 1979.

[20] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks”, *Journal of Statistical Mechanics, DOI:10.1088/1742-5468/2008/10/P10008*, 2008.

[21] P. Pons and M. Latapy, “Computing communities in large networks using random walks”, In: *Proc. of Computer and Information Sciences - ISCIS 2005*, pp. 284-293, 2005.

[22] M. E. J. Newman, “Finding community structure in networks using the eigenvectors of matrices”, *Physical Review E*, Vol.74, No.3, 2006.

[23] M. Newman, “Fast algorithm for detecting community structure in networks”, *Physical Review E,* Vol. 69, No.6, pp. 066133, 2004.

[24] G. K. Orman, V. Labaut, and H. Cherifi, “On accuracy of community structure discovery algorithms”, *Journal of Convergence Information Technology,* Vol.6, No.11, pp.283-292, 2011.

[25] S. Bharathi, A. Geetha, and R.Sathiynarayanan, “Sentiment Analysis of Twitter and RSS News Feeds and Its Impact on Stock Market Prediction”, *International Journal of Intelligent Engineering and Systems,* Vol.10, No.6, pp. 68-77, 2017.

[26] H. Akkineni, V. S. L. Papineni, and V. B. Burra, “Hybrid Method for Framing Abstractive Summaries of Tweets”, *International Journal of Intelligent Engineering and Systems*, Vol.10, No.3, pp.418-425, 2017.