Volume 2, No. 3, May-June 2011

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Detection of Outliers and Hubs Using Minimum Spanning Tree based on Node Connectivity

T. Karthikeyan
Department of Computer Science,
PSG College of Arts and Science,
Coimbatore,
Tamil Nadu, India
t.karthikeyan.gasc@gmail.com

S. John Peter*
Department of Computer Science
St. Xavier's College,
Palayamkottai,
Tamil Nadu,India.
jaypeeyes@rediffmail.com

Abstract-In data mining detection of anomalous pattern in data is more interesting than detecting inliers. Minimum Spanning Tree based Clustering is an important task for the discovery of underlying structures in graph. Many algorithms find clusters by maximizing the number of intra-cluster edges. While such algorithms find useful and interesting structures, they tend to fail to identify and isolate two kinds of vertices that play special roles – vertices that bridge clusters (hubs) and vertices that are marginally connected to clusters (outliers). In this paper, we model hubs as high-degree nodes having high Node Connectivity Factor (**NCF**) value. Identifying hubs is useful for applications such as viral marketing and epidemiology since hubs are responsible for spreading ideas and disease. In contrast, outliers have little or no influence, and many may be isolated as noise in the data. In this paper we propose an algorithm called Minimum Spanning Tree based Algorithm for Outliers and Hubs based on Node Connectivity (**MSTOHNC**), which detect outliers and hubs in graph.

Keywords: Euclidean minimum spanning tree; Degree; Outliers; Hub, Node Connectivity Factor;

I. INTRODUCTION

An outlier is an observation of data that deviates from other observations so much that it arouses suspicious that was generated by a different mechanism from the most part of data. Outlier may be erroneous or real in the following sense. Real outliers are observations whose actual values are very different than those observed for rest of the data and violate plausible relationship among variables. Erroneous outliers are observations that are distorted due to misreporting errors in the data collection process. Outliers of either type may exert undue influence on the result of data analysis. So they should be identified using reliable detection methods prior to performing data analysis [14].

The importance of outlier detection is due to the fact that outliers in the data translate to significant (and often critical) information in a wide variety of application domains. For example, an anomalous traffic pattern in a computer network could mean that a hacked computer is sending out sensitive data to an unauthorized destination. In public health data, outlier detection techniques are widely used to detect anomalous pattern in patient medical records which could be symptoms of new diseases. Similarly, outliers in credit card transaction data could indicate credit card theft or misuse. Outliers can also translate to critical entities such as in military surveillance, whereas the presence of unusual region in a satellite image of enemy are could indicate enemy troop movement. Or anomalous readings from space craft would signify a fault in some of the craft. Outlier detection has been found to be directly applicable in large number of domains.

Much data of current interest to the scientific community can be modeled as graphs. A graph is sets of vertices, representing objects, connected together by edges, representing the relationship between objects. For example World Wide Web can be modeled as a graph, where web pages are represented as vertices that are connected by an edge when one page contains a hyper link to another. [2, 20].

Given a connected, undirected, weighted graph, $G = (V, E)$, where V is the set of nodes, E is the set of edges between pairs of nodes, and a weight $w(u, v)$ specifying weight of the edge (u, v) for each edge $(u, v) \in E$. A spanning tree is an acyclic sub graph of a graph G, which contains all vertices from G. The Minimum Spanning Tree (**MST**) of a weighted graph is minimum weight spanning tree of that graph. Several well established **MST** algorithms exist to solve Minimum Spanning Tree problem [33, 25, 29]. The cost of constructing a Minimum Spanning Tree is $O(m \log n)$, where m is the number of edges in the graph and n is the number of vertices. More efficient algorithm for constructing **MST**s have also been extensively researched [19, 11, 12]. These algorithms promise close to linear time complexity under different assumptions. A Euclidean Minimum Spanning Tree (**EMST**) is a spanning tree of a set of n points in a metric space ($\mathbf{E}^n$), where the length of an edge is the Euclidean distance between a pair of points in the point set.

MST based clustering is a fundamental approach for detecting hidden structures in graph. Various methods have been developed. These methods tend to cluster graphs such there are a dense set of edges with in every cluster and few edges between clusters. However, they do not distinguish the roles of vertices in the graphs. Some vertices are members of cluster; some vertices are hubs that bridge many clusters but do not belong to any clusters, and some vertices are outliers that have only a weak association with particular cluster is shown in Fig 2. An outlier is an observation of data that deviates from other observations so much that it arouses suspicious that was generated by a different mechanism from the most part of data [14]. Inlier, on the other hand, is defined as observation that is explained by underlying probability density function. In clustering, outliers are considered as noise observations that should be removed in order to make more reasonable clustering [15]. An algorithm using Minimal Spanning Tree takes the advantage of **MST**. The **MST** ignores many possible connections between the data patterns, so the computation complexity can be decreased.

In this paper, we consider outliers as points or vertices having *degree number* 1 or 0, which are far from the most of other data. Here a vertex is defined as an outlier if it participates in at most *T* neighborhoods in **MST** graph, where threshold *T* is control parameter. We classify a vertex as outlier on basis of its *degree number* in the **MST** graph. We consider hub as an isolated vertex $v \in V$ has neighbors belonging to two or more different clusters. It bridges different clusters. Our approach will detect outliers and hub from data set with less computational effort.

Here, we consider hubs as *high-degree* nodes with less Node Connectivity Factor (***NCF)*** value with which most of other nodes interact. Hubs have been previously modeled as vertex covers of the graph model *G (V, E),* or by just blindly picking all nodes with degree above certain threshold. Modeling hubs by vertex covers forces all linkage to be covered hide threshold-based approach may miss some hub vertices that are below the degree threshold. We model hub as *high-degree* node having less Node Connectivity Factor (***NCF)*** association with other nodes with distance greater than certain threshold value.

The existing methods such as modularity-based algorithm [7] will partition the network into different number of clusters (communities). The identification and isolation of hubs is essential for many applications. As an example, the identification of hubs in the **WWW** improves the search for relevant authoritative web pages [21]. Furthermore, hubs are believed to play crucial role in viral marketing [10] and epidemiology [37].

In this paper we propose a new method for finding outliers and hubs called **MSTOHNC** *(Minimum Spanning Tree Algorithm for Outliers and Hubs based on Node Connectivity*). Connectivity of a source node is determined in terms of weighted edges associated with the adjacent nodes of the source node. The new objective function *Node Connectivity Factor* (***NCF***) is proposed to detect outliers and hubs in large graph.

The paper is organized as follows. We review the related works on outlier detection and hubs in section 2. We propose the **MSTOHNC** algorithm for detecting outliers and hubs in section 3. Finally in conclusion we summarize the strength of our method and possible improvements.

II. RELATED WORK

There is no single universally applicable or generic outlier detection approach [26, 31]. Therefore there are many approaches have been proposed to deduct outliers. These approaches are classified into four major categories as distribution-based, distance-based, density-based and clustering-based [40].

Distribution-based approaches [16, 5] develop statistical models from the given data then apply a statistical test to determine if an object belongs to this model or not. Objects that have low probability to belong to the statistical model are declared as outliers. However, Distribution-based approaches can not be applied in multidimensional dataset because of the univariate in nature. In addition, prior knowledge of the data distribution is required. These limitations have restricted the ability to apply these types of methods to large real-world databases which typically have many different fields and have no easy way of characterizing the multivariate distribution.

In the distance-based approach [22, 23, 24, 34], outliers are detected using a given distance measure on feature space, a point *q* in a data set is an outlier with respect to the parameters *M* and *d*, if there are less than *M* points within the distance *d* from *q,* where the values of *M* and *d* are determined by the user. The problem in distance–based approach is that it is difficult to determine the *M* and *d* values. Angiulli [4] propose a new definition of outliers. In this definition, for each point, *P*, the sum of the distances from its *k* nearest neighbor's considered. This sum is called the weight of *P*, $W_k (P)$ and is used to rank the points of data set. Outliers are those points having the largest values of W_k. The method proposed by Angiulli [4] needs expected number of outlier n and application dependent k as input parameter. It is difficult to predict correct values for n and k. The problem with distance based approach is its high computational complexity.

Density-based approaches [6, 32] compute the density of the regions in the data and declare the objects in low dense regions as outliers. Clustering-based approaches [26, 17, 18], consider clusters of small sizes as outliers. In these approaches, small clusters (clusters containing significantly less points than other clusters) are considered as outliers. The advantage of clustering based approaches is that they do not have to be supervised. Moreover, clustering –based technique are capable of being used in an incremental mode. Custom and Gath [8] presents a method on fuzzy clustering. In order to test the absence or presence of outliers, two hypotheses are used. However, the hypotheses do not account for the possibility of multiple clusters of outliers.

Jiang et. al.[18] proposed a two-phase method to detect outliers. In the first phase, clusters are produced using modified K-means algorithm, and then in the second phase, an Outlier-Finding Process **(OFP)** is proposed. The small clusters are selected and regarded as outliers. Small cluster is defined as a cluster with fewer points than half the average number of points in the *k* number of clusters. Loureio [26] proposed a method for detecting outlier. Hierarchical clustering technique is used for detecting outliers. The key idea is to use the size of the resulting clusters as indicators of the presence of outliers. Almedia [3] is also used similar approach for detecting outliers. Using the K-means clustering algorithm Yoon [36] proposed a method to detect outliers. The K-means algorithm is sensitive to outliers, and hence it may not give accurate result.

Moh'd Belal Al-Zoubi [27] proposed a method based on clustering approaches for outlier detection using Partitioning Around Medoid (**PAM**). **PAM** attempts to determine *k* partition for *n* objects. The algorithm uses most centrally located object in a cluster (called medoid) instead of cluster mean. **PAM** is more robust than the k-means algorithm in the presence of noise and outlier. This **PAM** based approach suffer from proper cluster Structure. Cluster with irregular boundaries can not be detected using both k-means and **PAM** algorithms.

In [26] clustering methods have been applied to detect outliers. The key idea is to use the size of the resulting clusters as indicators of the presence of outliers. The authors use a hierarchical clustering technique. A similar approach was reported in [3].

In [36], the authors proposed a clustering-based approach to detect outliers. Xiaowei et al [38] proposed a method to find clusters, outliers and hubs based on undirected and un-weighted graph (Networks). Sriganesh Srihari et al [35] proposed a method for finding hubs and communities in scale-free networks.

III. MSTOHNC Algorithm for Outlier and Hubs

In this paper, we focus on simple, undirected and weighted graph. Let $G = \{V, E\}$ be a graph, where V is a set of vertices; and E is a set of pairs of distinct vertices, called edges. $W(u, v)$ is the weight of the edge (u, v). The hierarchical method starts by constructing a Minimum Spanning Tree (**MST).** The weight of the edge in the tree is Euclidean distance between the two end points (vertices). So we named this **MST** as **EMST.** Xiaowei et al [38] proposed a method to find clusters, outliers and hubs based on undirected and un-weighted graph (Networks). Clustering algorithms using Minimal Spanning Tree takes the advantage of **MST**. The **MST** ignores many possible connections between the data patterns, so the computation time can be decreased. We propose a new definition for outliers and hubs based on the **EMST**. We propose a new algorithm, *Minimum Spanning Tree Algorithm for Outliers and Hubs based on Node Connectivity* (***MSTOHNC***). Here the number of edges between the vertices (objects) is considerably reduced. So the performance of our approach has been improved over the method proposed in [38].

To detect the outliers and isolate hubs using **EMST**, we use the *degree number* of points (objects) and Node Connectivity Factor (NCF) of points (objects) in the **EMST.** For any undirected graph G the *degree* of a vertex *v*, written as *deg (v),* is equal to the number of edges in G which contains *v*, that is, which are incident on *v* [13].

We define Node Connectivity Factor (**NCF**) of each node $u\in V$ in **EMST** is the ratio between summation of weights of all adjacent node edges *v* of node *u* in **EMST,** ***SW (v)*** and degree number of the node *u*.

Node Connectivity Factor (u) = Summation of weights of all adjacent node edges of node *u* in **EMST** / Degree number of the node *u, deg (u).*

$$SW(u) = \sum_{v\in adj(u)} W(v) \qquad (1)$$

$$NCF(u)=SW(u)/deg(u) \qquad (2)$$

We propose the following definition for outliers and hubs based on **EMST,**

Definition 1: Given an **EMST** for a data set *S*, outlier is a vertex or node *v,* whose *degree* is equal to 1 or 0, with $NCF > T$.
where *T* is a threshold value used as control parameter. From the input graph *G* (shown in Fig 1) **EMST** is constructed. Then vertices *v,* which have *degree* 1 or 0 are identified. Then the ***NCF*** of the vertex *v* is compared with threshold *T*. If the value of *NCF* is greater than the threshold value *T* then the corresponding vertex is identified as an outlier is shown in the Fig 2.

Definition 2: Given an **EMST** for a data set *S*, hub is a vertex or node *v*, whose *degree* is greater than 2, with $NCF > T$.

where *T* is a threshold value used as control parameter. The vertices *v,* which have *degree* 2 or more are identified. Then the *NCF* of the vertex *v* is compared with threshold *T*. If the value of *NCF* is grater than the threshold value *T* then the corresponding vertex is identified as hub is shown in the Fig 2.

When scanning the **EMST,** the edges are ordered from larger to smaller lengths. Then we define the threshold as:

$$T=max\ (L_i - L_{i-1})*t \qquad (3)$$

Where L_i is largest in the order and $t \in [0, 5]$ is a user defined parameter.

The structure of a vertex in **EMST** can be described by its neighborhood. Let $v \in V$, the structure of *v* is defined by its neighborhood denoted by $\Gamma(v)$

$$\Gamma(v) = \{w\in V|(v, w)\in E\}\cup\{v\} \qquad (4)$$

In Fig 1 vertex 6 is a hub sharing neighbors, with two clusters. If we only use the number of shared neighbors, vertex 6 will be clustered in to either of the clusters or cause two clusters to merge. Therefore, we normalize the number of common neighbors by computing distance of two neighbors using the function *dist (v, w)* the values of structural similarity will be computed. When a member of a cluster shares a similar structure with one of its neighbors, their computed structural similarity will be large. We apply a threshold ε to the computed structural similarity when assigning cluster membership, formulized in the following ε-neighborhood definition.

$$N_{\varepsilon}(v)=\{w\in\Gamma(v)|dist(v, w)\geq\varepsilon\} \qquad (5)$$

When a vertex shares structural similarity with enough neighbors, it becomes a nucleus or seed for cluster. Such a vertex is called a core vertex. Core vertices are special classes of vertices that have a minimum of μ neighbors with a structural similarity that exceeds the threshold ε. From core vertices we grow the clusters. In this way the parameters μ and ε determine the clustering of graph. For a given ε, the minimal size of a cluster is determined by μ. Let $\varepsilon\in\Re$ and $\mu\in\aleph$, a vertex $v \in V$ is called a core w.r.t. ε and μ, if its ε-neighborhood contains at least μ vertices.

$$CORE_{\varepsilon,\mu}(v)\Leftrightarrow/N_{\varepsilon}(v)/\geq\mu \qquad (6)$$

```
Algorithm: MSTOHNC ( )

Input     :  $G = (V, E), \mu, \varepsilon, T$
Output    :  Set of outliers $O$ and Hubs H

Let $O$ be set for outliers
Let $H$ be set for Hubs
Let $\varepsilon, \mu, T$ be input parameters

1. Let $O=\Phi; H = \Phi$

2. Construct an EMST from $G$

3. Compute $NCF$ for each node in EMST

4. For each vertex $v\in V$ in EMST do

5.    If $CORE_{\varepsilon,\mu}(v)$ then

6.       If deg $(v) \geq 2$ and $NCF(v) > T$ then
               $H = H \cup \{v\}$

7.    Else

8.          If deg $(v) < 2$ and $NCF(v) > T$ then
               $O = O \cup \{v\}$

9. Return set of $O$ and $H$
```

Consider vertices in the Fig 1. The Vertices 0 and 5 are connected by edges. Their neighbors are the vertices {0, 1, 4, 5, 6} and {0, 1, 2, 3, 4, 5} respectively. They share many neighbors and they are grouped together in the same cluster. In contrast, consider the neighborhoods of vertex 13 and vertex 9. These two vertices are connected, but share only two common neighbors ie., {9, 13}. Therefore, it is doubtful that they should be grouped in to the same cluster. The situation of vertex 6 is a little different. It has many neighbors, but they are sparsely interconnected. Our **MSTOHNC** algorithm isolates vertex 13 as an outlier and vertex 6 as hub.

Here we use Fig 2 to illustrate the **MSTOHNC** algorithm for detecting outliers and hubs. The **MSTOHNC** algorithm first identifies hubs and then it locates and isolates outliers. The **MSTOHNC** algorithm first constructs **EMST** from the graph G (line 2). Then Node Connectivity Factor ***(NCF)*** for each vertex in **EMST** is computed at line 3. If a vertex *v* is a core, **MSTOHNC** starts with an arbitrary core *v* and search for all hubs (line 6). If the vertex is not a core, then the algorithm starts search for outliers (lines 8). The lines 5-8 in the algorithm are repeated for each of the vertices in the **EMST.**

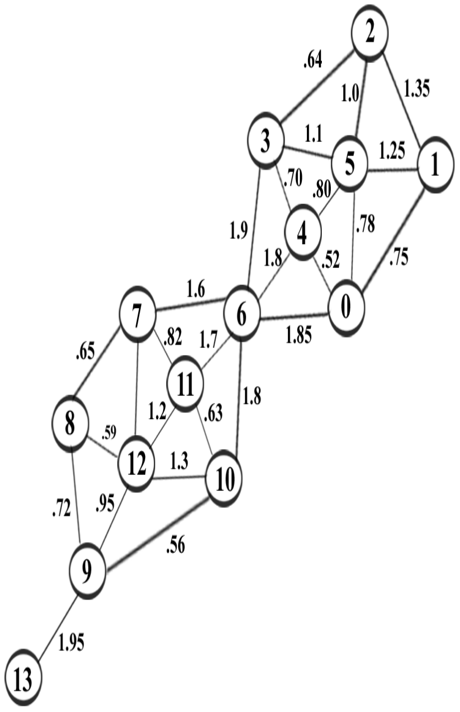
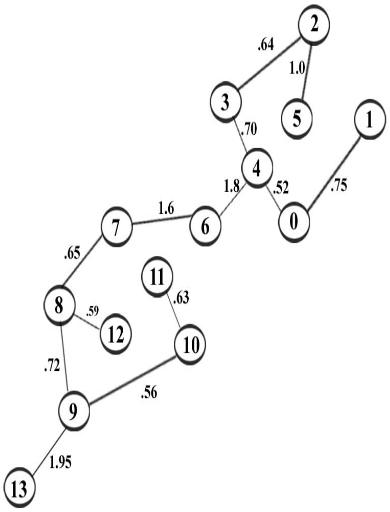


Fig 1: Given input graph *G= (V, E)*

IV. CONCLUSION

In this paper, we proposed **MSTOHNC** algorithm to detect outliers and hubs in graphs. In a graph there are vertices (objects) that play a special role, vertices that bridge clusters (hubs). Identifying hubs is essential for application such as viral marketing and epidemiology. As vertices that bridge clusters, hubs are responsible for spreading ideas or disease. In contrast, outliers have little or no influence, and may be isolated as noise in the data. Many algorithms [16, 18, 22, 32] have been proposed to detect outliers. A few algorithms [35, 38] have been proposed to detect hubs. Our **MSTOHNC** algorithm detects both outliers as well as hubs with less computational time. In Future we will try to apply our algorithm in various domains including bioinformatics, medical image processing and geographical information system.



Vertex v	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Degree(v)	2	1	2	2	3	1	2	2	3	3	2	1	1	1
SW(V)	1.27	0.75	1.64	1.34	3.02	1.0	3.4	2.25	1.96	3.23	1.19	0.63	0.59	1.95
NCF(V)	0.63	0.75	0.82	0.67	1.0	1.0	1.7	1.12	0.65	1.07	0.59	0.63	0.59	1.95

Fig 2: EMST from G with Outlier vertex 13 and Hub vertex 6.

V. REFERENCES

[1] C. Aggarwal and P. Yu, "Outlier Detection for High Dimensional Data" In Proceedings of the ACM SIGMOD International Conference on Management of Data, Volume 30, Issue 2,pages 37 – 46, May 2001.

[2] R. Albert, H. Jeong, and A.-L. Barabási, "Diameter of the world-wide web." Nature 401, 130–131 1999.

[3] J.Almeida, L.Barbosa, A.Pais and S.Formosinho, "Improving Hierarchical Cluster Analysis: A new method with Outlier Detection and Automatic Clustering", Chemometrics and Intelligent Laboratory Systems 87:208-217, 2007.

[4] F.Angiulli, and C.Pizzuti, "Outlier Mining in Large High-Dimensional Data sets",IEEE Transactions on Kaoeledge and Data Engineering, 17(2): 203-215, 2005.

[5] V.Barnett and T.Lewis, "Outliers in Statistical Data"., John Wiley, 1994.

[6] M. Breunig, H.Kriegel, R.Ng and J.Sander, "Lof: Identifying density-based local outliers", In Proceedings of 2000 ACM SIGMOD International Conference on Management of Data. ACM Press, pp 93-104, 2000.

[7] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community in very large networks", Physical Review E 70,066111 2004.

[8] B. Custem and I. Gath, "Detection of Outliers and Robust Estimation using Fuzzy clustering", Computational Statistics & data Analyses 15,pp 47-61, 1993.

[9] C. Ding, X. He, H. Zha, M. Gu, and H. Simon, "A min-max cut algorithm for graph partitioning and data clustering", Proc. of ICDM 2001.

[10] P. Domingos and M. Richardson, "Mining the Network Value of Customers", Proc. 7th ACM SIGKDD, pp. 57 – 66, 2001.

[11] M. Fredman and D. Willard. "Trans-dichotomous algorithms for minimum spanning

trees and shortest paths". In Proceedings of the 31st Annual IEEE Symposium on Foundations of Computer Science,pages 719-725, 1990.

[12] H.Gabow, T.Spencer and R.Rarjan. "Efficient algorithms for finding minimum spanning trees in undirected and directed graphs", Combinatorica, 6(2):pp 109-122, 1986.

[13] Gary Chartrand and Ping Zhang "Introduction to Graph Theory", Tata MgrawwHill, Paperback-2008.

[14] B. Ghosh-Dastidar and J.L. Schafer, "Outlier Detection and Editing Procedures for Continuous Multivariate Data". ORP Working Papers, September 2003.

[15] S. Guha, R. Rastogi, and K. Shim. "CURE an efficient clustering algorithm for large databases". In Proceeding of the 1998 ACM SIGMOD Int. Conf. on Management of Data , pp 73-84, Seattle, Washington, 1998.

[16] D.Hawkins, "Identifications of Outliers", Chapman and Hall, London, ,1980.

[17] Z. He, X. Xu and S. Deng, "Discovering cluster-based Local Outliers", Pattern Recognition Letters, Volume 24, Issue 9-10, pp 1641 – 1650, June 2003.

[18] M. Jaing, S. Tseng and C. Su, "Two-phase Clustering Process for Outlier Detection", Pattern Recognition Letters, Volume 22, Issue 6 – 7, pp 691 – 700, May 2001

[19] D. Karger, P. Klein and R. Tarjan, "A randomized linear-time algorithm to find minimum spanning trees", Journal of the ACM, 42(2):321-328, 1995.

[20] J. M. Kleinberg, S. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "The Web as a graph: Measurements,models and methods". In Proceedings of the International Conference on Combinatorics and Computing, number 1627 in Lecture Notes in Computer Science, pp. 1–18, Springer, Berlin 1999.

[21] J. Kleinberg. "Authoritative sources in a hyperlinked environment." Proc. 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.

[22] E. Knorr and R. Ng, "A Unified Notion of Outliers: Properties and Computation", In Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, pages 219 – 222, August 1997.

[23] E..Knorr and R.Ng, "Algorithms for Mining Distance-based Outliers in Large Data sets", Proc.the 24th International Conference on Very Large Databases(VLDB),pp.392-403, 1998.

[24] E.Knorr, R.Ng and V.Tucakov, "Distance- Based Outliers: Algorithms and Applications", VLDB Journal, 8(3-4):237-253, 2000.

[25] J. Kruskal, "On the shortest spanning subtree and the travelling salesman problem", In Proceedings of the American Mathematical Society, pp 48-50, 1956.

[26] A. Loureiro, L. Torgo and C. Soares, "Outlier detection using Clustering methods: A data cleaning Application", in Proceedings of KDNet Symposium on Knowledge-based systems for the Public Sector. Bonn, Germany, 2004.

[27] Moh'd Belal Al-Zoubi, "An Effective Clustering-Based Approach for Outlier Detection", European Journal of Scientific Research, Vol.28 No.2 ,pp.310-316, 2009

[28] Moh'd Belal Al-Zoubi, Al-Dahoud Ali, Abdelfatah A. Yahya, "Fuzzy Clustering-Based Approach for Outlier Detection", Proceedings of the 9th WSEAS Int. Conference on APPLICATIONS of COMPUTER ENGINEERING, 2008.

[29] J. Nesetril, E.Milkova and H.Nesetrilova. Otakar boruvka "On Minimum spanning tree problem: Translation of both the 1926 papers, comments, history. DMATH:" Discrete Mathematics, 233, 2001.

[30] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks", Phys. Rev. E 69, 026113 2004.

[31] K.Niu, C.Huang, S.Zhang and J.Chen, "ODDC: Outlier Detection Using Distance Distribution Clustering", T.Washio et al. (Eds.): PAKDD 2007 Workshops, Lecture Notes in Artificial Intelligence (LNAI) 4819,pp.332-343,Springer-Verlag, 2007.

[32] S.Papadimitriou, H.Kitawaga, P.Gibbons and C. Faloutsos, "LOCI: Fast outlier detection using the local correlation integral", Proc. Of the International Conference on Data Engineering ,pp.315-326, 2003.

[33] R. Prim. "Shortest connection networks and some generalization". Bell systems Technical Journal,36:1389-1401, 1957.

[34] S. Ramaswamy, R. Rastogi and K. Shim, "Efficient Algorithms for Mining Outliers from Large Data Sets", In Proceedings of the 2000 ACM SIGMOD InternationalConference on Management of Data, Volume 29, Issue 2, pages 427 – 438, May2000.

[35] Srganesh Srihari, Hoong Kee Ng, Kang Ning and Hon Wai Leong " Detecting Hubs and Quasi Cliques in Scale free Network", IEEE 2008.

[36] K. Yoon, O. Kwon and D. Bae, "An approach to outlier Detection of Software Measurement Data using the K-means Clustering Method", First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007), Madrid.pp.443-445, 2007.

[37] Y. Wang, D. Chakrabarti, C. Wang and C. Faloutsos, "Epidemic Spreading in Real Networks: An Eigenvalue Viewpoint", SRDS 2003 (pages 25-34), Florence, Italy.

[38] Xiaowei Xu, Nurcan Yuruk Zhidan Feng and Thomas A.J. Schweiger, "SCAN: A Structural Clustering Algorithm for Networks", SIGKDD, San Jose, CA, US, 2007.

[39] J.Zhang and N.Wang, "Detecting outlying subspaces for high-dimensional data: the new task", Algorithms and Performance, Knowledge and Information Systems, 10(3):333-555, 2006.