Global Journal of Computational Intelligence Research.
ISSN 2249-0000 Volume 1, Number 1 (2011), pp. 31-39

http://www.ripublication.com/gjcir.htm

Quality Clusters with Outliers using Minimum Spanning Tree

T. Karthikeyan[1] and S. John Peter[2]

[1]Department of Computer Science,
PSG College of Arts and Science, Coimbatore, Tamil Nadu, India
E-mail: t.karthikeyan.gasc@gmail.com
[2]Department of Computer Science and Research Center
St. Xavier's College, Palayamkottai, Tamil Nadu, India.
E-mail: jaypeeyes@rediffmail.com

Abstract

Clustering is a process of discovering group of objects such that the objects of the same group are similar, and objects belonging to different groups are dissimilar. A number of clustering algorithms exist that can solve the problem of clustering, but most of them are very sensitive to their input parameters. Minimum Spanning Tree clustering algorithm is capable of detecting clusters with irregular boundaries. Detecting outlier in database (as unusual objects) is a big desire. In data mining detection of anomalous pattern in data is more interesting than detecting inliers. In this paper both partitioning and density-based methods were adopted using Minimum Spanning Tree, resulting in a new algorithm – Clustering Algorithm based on Objects Density using Minimum Spanning Tree (CAODMST). The key feature of the algorithm is, it finds high quality clusters and detect outliers in the data set.

Keywords: Euclidean minimum spanning tree, Clustering, Center, Outliers.

Introduction

One of the best known problems in the field of data mining is clustering. The problem of clustering is to partition a data set into groups (clusters) in such a way that the data elements within a cluster are more similar to each other than data elements in different clusters [1]. Clustering is the subject of active research in several fields such as statistics, pattern recognition, machine learning, and data mining. A wide variety of clustering algorithms have been proposed for different applications [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 [3]. 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 [1] 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. Outliers can often be individual or groups of clients exhibiting behavior outside the range of what is considered normal. Outliers can be removed or considered separately in *regression modeling* to improve accuracy which can be considered as benefit of outliers. Identifying them prior to modeling and analysis is important [4]. In clustering–based methods, outlier is defined as observation that does not fit to the overall clustering pattern [5]

A high quality clustering algorithm will be

- Capable to identify strongest intra-cluster similarity data or objects;
- Capable to identify weakest inter-cluster data or objects;
- Capable to identify hidden patterns;
- Should not have predefined cluster number as input parameter.

The quality of clustering usually depends on the similarity measurement adopted and the implementation of the algorithm. It also depends on whether the algorithm can identify some or all hidden patterns.

K-means algorithm is well established and mature, and is most frequently used clustering method. Many methods and algorithms have been developed that harness the K-means algorithm. Jain [2] explores a density based approach to identify clusters in K-dimensional point sets. The data set is portioned into a number of nonoverlapping cells and histograms are constructed. Cells with relatively high frequency counts of points are the potential cluster centers and boundaries between clusters fall in the "valleys" of the histogram. This method has the capability of identifying clusters of any shape.

Many data-mining algorithms find outliers as a side-product of clustering algorithms. However these techniques define outlier as points, which do not lie in clusters. Thus, the techniques implicitly define outliers as the background noise in which the clusters are embedded. Another class of techniques defines outlier as points, which are neither a part of a cluster nor part of background noise; rather they are specifically points which behave very differently from the norm [6].

Given a connected, undirected 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 subgraph 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 [7], [8], [9]. 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 MSTs has also been extensively researched [10], [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 (E^n), where the length of an edge is the Euclidean distance between a pair of points in the point set.

Clustering algorithms using minimal spanning tree takes the advantage of MST. The MST ignores many possible connections between the data patterns, so the cost of clustering can be decreased. The MST based clustering algorithm is known to be capable of detecting clusters with various shapes and size [13]. Unlike traditional clustering algorithms, the MST clustering algorithm does not assume a spherical shapes structure of the underlying data. The EMST clustering algorithm [14], [7] uses the Euclidean minimum spanning tree of a graph to produce the structure of point clusters in the *n*-dimensional Euclidean space. Clusters are detected to achieve some measures of optimality, such as minimum intra-cluster distance or maximum inter-cluster distance [15]. The EMST algorithm has been widely used in practice.

All existing clustering Algorithm require a number of parameters as their inputs and these parameters can significantly affect the cluster quality. Our algorithm does not require a predefined cluster number. In this paper we want to avoid experimental methods and advocate the idea of need-specific as opposed to case-specific because users always know the needs of their applications. We believe it is a good idea to allow users to define their desired similarity within a cluster and allow them to have some flexibility to adjust the similarity if the adjustment is needed. Our Algorithm produces clusters of *n*-dimensional points with a naturally approximate intra-cluster distance.

In this paper, we propose a new *Clustering Algorithm based on Objects Density using Minimum Spanning Tree (CAODMST)* which fuses the particular strength of both the K-means algorithm and DENCLUE algorithm using Minimum Spanning Tree resulting in performance enhancements.

This paper is structured as follows: In section 2 we review some of the existing works on clustering algorithms and outliers. In Section 3 we propose CAODMST algorithm which produces high quality clusters with outliers. Finally in conclusion we summarize the strength of our methods and possible improvements.

Related Work

The Prototype-based clustering techniques create a one-level partitioning of the data object. There are a number of such techniques, but two of the most prominent are K-means and K-medoids. K-means defines a prototype in terms of a centroid, which is usually the mean of a group of points. K-medoids defines a prototype in terms of a medoid, which is the most representative point for a group of points, and can be applied to a wide range of data since it requires only proximity measure for a pair of objects.

A K-means clustering technique is simple. A key precondition of the K-means algorithm [16] [17] is that the user must determine the number K of the clusters. As the clustering result is very sensitive to the value of the K, and different K values can often result in completely different result, a user determined K value can make the clustering result very unsatisfactory. Thus users need domain knowledge to a good K

value. This reduces the applicability and automation level of K-means. So far, there is no simple and universal applicable solution to the initial point's selection problem. K-means algorithm is also very sensitive to abnormal data. If some maximum value exists, the data distribution may be highly distorted.

DBSCAN [18] operationalizes density propagation according to the principle "accessibility from a core point". Each points whose *ε-neighborhood* contains more points than *MinPts* is called a core point. Each point which lies in an ε- neighborhood of a core points *p* adopts the same cluster label as *p.* The DBSCAN-algorithm propagates this relation through the set *D.* The algorithm terminates if each points is either assigned to a certain cluster or classified as noise.

Neighborhood-based Clustering (NBC) algorithm [19] proposed by Zhou S. G. et al is a good data clustering algorithm and can discover clusters of arbitrary shape and different densities using neighboring relationship among data points. To apply NBC to segment an image fast and efficiently Grayscale k-neighborhood based Density Factor (GNDF) [20] is introduced, which characterizes the local density of a gray's neighborhood in a relative sense.

DENCLUE [21] [22] adopts density-based clustering approach that models the overall density of a set of points as the sum of influence functions associated with each point. The resulting overall density function will have local peaks, ie., local density maxima, and these peaks can be used to define clusters in a natural way. But DENCLUE can be more computationally expensive than other density based clustering technique and it is one of the limitations of DENCLUE.

DENCLUE [23] has a solid theoretical foundation and it can be used to determine the K value and the initial cluster center points based on a density function, without a need for human intervention. In this way, the K-means algorithm will not be influenced by the user determine K value and random initial cluster center points, and user no longer need to have domain expertise. So, combination of both should theoretically improve clustering performance [24].

Zhan [13] proposed Minimum Spanning Tree (MST) based clustering algorithm. The MST based clustering algorithm has been widely used in practice. Xu (Ying), Olman and Xu (Dong) [25] use MST as multidimensional gene expression data. They point out that MST- based clustering algorithm does not assume that data points are grouped around centers or separated by regular geometric curve. Thus the shape of the cluster boundary has little impact on the performance of the algorithm. They described three objective functions and the corresponding cluster algorithm for computing *k*-partition of spanning tree for predefined $k > 0$. The algorithm simply removes *k*-1 longest edges so that the weight of the subtrees is minimized. The second objective function is defined to minimize the total distance between the center and each data point in the cluster. The algorithm removes first *k*-1 edges from the tree, which creates a *k*-partitions.

There is no single universally applicable or generic outlier detection approach [26], [27]. Therefore there is 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* [5]. Here we discuss about density-based and clustering-based approaches for outlier detection..

In *Density-based* methods outlier is defined from local density of observation. These methods used different density estimation strategies. A low local density on the observation is an indication of a possible outlier. Brito et al [28] proposed a *Mutual k-Nearest-Neighbor* (MkNN) graph based approach. MkNN graph is a graph where an edge exits between vertices v_i and v_j if they both belong to each others k-neighborhood.

Clustering-based approaches [26], [29], [30], [31], 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.

Our Clustering Algorithm

Through this MST representation, we can convert a multi-dimensional clustering problem to a tree partitioning problem, i.e. finding particular set of tree edges and then cutting them. Representing a set of multi-dimensional data points as simple tree structure will clearly lose some of the inter data relationship. However many clustering algorithm proved that no essential information is lost for the purpose of clustering. This is achieved through rigorous proof that each cluster corresponds to one subtree, which does not overlap the representing subtree of any other cluster. Clustering problem is equivalent to a problem of identifying these subtrees through solving a tree partitioning problem. The inherent cluster structure of a point set in a metric space is closely related to how objects or concepts are embedded in the point set. In practice, the approximate number of embedded objects can sometimes be acquired with the help of domain experts. Other times this information is hidden and unavailable to the clustering algorithm.

CAODMST Clustering Algorithm

CAODMST is an incremental algorithm. It uses the nearby data points to increase the initial clusters. If a candidate data points meets any principle of any cluster, the point can be clustered to a certain cluster. For every single clustering event, CAODMST can start from any object (point) in the cluster. This does not affect the clustering result. It is important to note that only if an object is a density attractor, the point is a cluster center. Only a cluster center can cluster its nearby points.

Here we describe the key concepts which are required for CAODMST algorithm.

Object Density: Given space $\Omega \in F^d$ consists of a data set of n objects $D = \{x_1, x_2, \ldots\ldots\ldots\ldots x_n\}$ in which density of x_i, density $density(x_i)$, is the value of the influence function of the object in space.

$$density(x_i) = \sum_{j=1}^{n} e^{-d(x_i - x_j)^2/2\sigma^2} \quad (1)$$

where the Gaussian influence function

$$e^{-d(x_i - x_j)^2/2\sigma^2}$$

$f_{guass}(x_i, x_j) = e$ indicates the density influence of each data object to the density of object x_i, and σ is the density adjustment parameter which is analogous to the standard deviation, and governs how quickly the influence objects drops off [24].

Neighborhood Radius: Neighborhood radius *R*: for any object *x* and distance *R* in space, a circular region with center *x* and radius *R* is defined as neighborhood of objects *x*, marked as $\delta = \{x|0<d(x, x_j)\leq R\}$. The radius of the defined neighborhood can be calculated as follows

$$R = mean\ (D)/n^{coefR} \qquad (2)$$

where *mean(D)* is the mean Euclidean distance among all objects and *coefR* is the coefficient of neighborhood radius adjustment.

Object Neighborhood: For any object *x* and distance *R* in space, a circular region with center *x* and radius *R* is called neighborhood of object x, defined as $\delta = \{x|0<d(x, x_j)\leq R\}$, in which $d(x, x_j)$ is the distance between objects *x* and x_j.

Candidate Object: A candidate object is such a point that does not yet belong to the current cluster but needs to be clustered. For every new member *S* of the current cluster *C,* a circle region with a suitable radius *R* is used to examine and find out new candidate object.

Based on the definition of *small clusters* as defined in [26], we define *small cluster* as a cluster with fewer points than half the average number of points in the *K* number of clusters.

The algorithm first read data set from data base and convert into EMST. The weight of the edge in the tree is Euclidean distance between the two end points. Then it finds density attractors to be the cluster centers and produce candidate objects. Then it examine the remaining objects by checking the distance between each object and cluster center is less than or equal to neighborhood radius *R* and if it is does, the object belong to the cluster otherwise it does not. After the object is clustered, the object does not participate in the next clustering process and it is removed from the data set (EMST). After all the clusters have been completed, remaining data objects will be examined to determine outliers in the data set.

Algorithm: CAODMST ( )
Input: *S* the point set, coefR
Output: Clusters with outliers

Let σ be the standard deviation of the edge weights in EMST
Let *i* be the number of clusters

1. Construct an EMST *T* from *S*
2. Compute standard deviation σ of the edges from EMST
3. Compute the densities of each data objects (Points) in EMST *T* using (1)
4. $i = 1$; $O = \Phi$;

5. Repeat
6. Seek the maximum density attractor *ODensityMaxi* in the EMST *T* as cluster center of c_i Assign the objects in the EMST *T* which are with in the radius *R* (inside the circle) to the cluster center c_i of the cluster C_i and at the same time delete the clustered objects from the EMST *T.*
7. i =*i + 1*
8. Until the points (objects) in EMST *T* is empty
9. Mark small clusters which have a few points as outliers; *O = OU{small clusters}*
10. Return *i* number of clusters with outliers

The CAODMST algorithm reads data set from data base and converts into EMST (line 1). Standard deviation is computed from the edges of EMST (line 2). Density for each object (point) is computed for identifying density attractor (line 3). Next we find Maximum density attractor for identifying current cluster center (line 6). Then candidate objects are assigned to cluster center and removed from EMST (line 7). Lines 6-8 in the algorithm are repeated until points (objects) in the EMST become empty. Finally Outliers are detected based on the definition of small clusters in line 10.

Conclusion

Our CAODMST clustering algorithm does not assumes any predefined cluster number. The algorithm gradually finds clusters with center for each cluster. These clusters ensure guaranteed intra-cluster similarity. Our algorithm also detects outliers in the dataset. Our algorithm does not require the users to select and try various parameters combinations in order to get the desired output. All of these look nice from theoretical point of view. However from practical point of view, there is still some room for improvement for running time of the clustering algorithm. This could perhaps be accomplished by using some appropriate data structure. In the future we will explore and test our proposed clustering algorithm in various domains. We will further study the rich properties of EMST-based clustering methods in solving different clustering problems.

References

[1] 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.

[2] Anil K. Jain, Richard C. Dubes “Algorithm for Clustering Data”, Michigan State University, Prentice Hall, Englewood Cliffs, New Jersey 07632.1988.

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

[4] G. Williams, R. Baxter, H. He, S. Hawkins and L. Gu, "A Comparative Study for RNN for Outlier Detection in Data Mining", In Proceedings of the 2nd IEEE International Conference on Data Mining, page 709, Maebashi City, Japan, December 2002.

[5] 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.

[6] 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.

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

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

[9] 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.

[10] 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.

[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):109-122, 1986.

[13] C. Zahn. "Graph-theoretical methods for detecting and describing gestalt clusters". IEEE Transactions on Computers, C-20:68-86, 1971.

[14] F. Preparata and M.Shamos. "Computational Geometry": An Introduction. Springer-Verlag, Newyr, NY,USA, 1985

[15] T. Asano, B. Bhattacharya, M.Keil and F.Yao. "Clustering Algorithms based on minimum and maximum spanning trees". In Proceedings of the 4th Annual Symposium on Computational Geometry,Pages 252-257, 1988.

[16] S.Z. Selim, M.A. Ismail, "K-Means-Type Algorithms: A Generalized Convergence Theorem and characterization of Local Optimality," IEEE Trans Pattern Analysis and Machine Intelligence, pp. 81-87, 1984.

[17] D.T. Pham, S.S. Dimov, C.D. Nguyen, "An Incremental K-means Algorithm", Proceedings of the Institution of Mechanical Engineers, Journal of Mechanical Engineering Science, vol. 218, Issue 7, pp.783-795, 2004.

[18] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD96), 1996.

[19] Zhou, S., Zhao, J. "A Neighborhood-Based Clustering Algorithm". PAKD 2005, LNAI 3518 361-371, 1982.

[20] Jundi Ding, SongCan Chen, RuNing Ma and Bo Wang, "A Fast Directed Tree Based Neighborhood Clustering Algorithm for Image Segmantation", Neural Information Processing, Lecture Notes in Computer Science, Vol 4233,pp 369-378, 2006.

[21] A. Hinneburg and D.A. Keim, "An Efficient Approach to Clustering in Large Multimedia Databases with Noise," In Proc. Of th 4th Intl. Conf. on Knowledge Discovery and Data Mining, pp. 58-65. 1998.

[22] A. Hinneburg and D.A. Keim, "A General Approach to Clustering in Large Databases with Noise," Knowledge and Information Systems (KAIS), vol. 5, no. 4, pp. 387-415, 2003.

[23] P.N. Tan, M. Steinbach, and V. Kumar, "Introduction to Data Mining," Post & Telecom Press of P.R.China (China Edition), pp. 377-379, 2006.

[24] Yu-Chen Song, J.O'Grady,G.M.P.O'Hare, Wei Wang,"A Clustering Algorithm incorporating Density and Direction", IEEE Computer Society, CIMCA 2008.

[25] Y.Xu, V.Olman and D.Xu. "Minimum spanning trees for gene expression data clustering". Genome Informatics,12:24-33, 2001.

[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] 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.

[28] M. R. Brito, E. L. Chavez, A. J. Quiroz, and J. E. Yukich. "Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection". Statistics & Probability Letters, 35(1):33-42, 1997.

[29] Gath and A.Geva, "Fuzzy Clustering for the estimation of the Parameters of the components of Mixtures of Normal distribution", Pattern Recognition letters, 9, pp.77-86, 1989.

[30] 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.

[31] 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.