ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2016

Minimum Spanning Tree based Clustering with Cluster Validity in Data Mining

Indu Maurya

Research Scholar, Dept. of CSE, B.I.E.T Jhansi, Uttar Pradesh, India

ABSTRACT: The minimum spanning tree clustering algorithm is used for detecting clusters with irregular boundaries. In this paper, we are considering two clustering algorithm .First we will take a set of points of any given k and produces a k-partition of them. It produces k clusters with center and guaranteed intra-cluster similarity. This process is repeated until k clusters are produced. In the second algorithm maximizing the overall standard deviation reduction, in order to produce a group of clusters by partitions a point set, without a given k value. In this paper we used both the algorithms to find Informative Meta similarity clusters.

KEYWORDS: Minimum spanning trees, k-constrained clustering, and representative point sets, standard deviation reduction, and Euclidean minimum spanning tree.

I. INTRODUCTION

A spanning tree is an acyclic subgraph of a graph *G*, which contains all the vertices from *G*. The minimum spanning tree (MST) of a weighted graph is the minimum weight spanning tree of that graph. With the classical MST algorithms [7,4,5], the cost of constructing a minimum spanning tree is *O* (*m*log *n*), where *m* is the number of edges in the graph, *n* is the number of vertices. More efficient algorithms for constructing MSTs have also been extensively researched [3,1,2]. 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 is mostly unsupervised process, thus evaluating the result of the clustering algorithms is very important. In the clustering process there are no predefined classes therefore it is difficult to find an appropriate metric for measuring whether the cluster configuration found during the process, is acceptable or not. Several clustering approaches have been developed [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 contain 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 [10], [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 **MST**s has also been extensively researched. 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.

The MST clustering algorithm is known to be capable of detecting clusters with irregular boundaries [11]. Unlike traditional clustering algorithms, the MST clustering algorithm does not assume a spherical shaped clustering structure of the underlying data. The EMST clustering algorithm 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 measure of optimality, such as minimum intracluster distance or maximum intercluster distance .The (E) MST clustering algorithm has been widely used in practice. An example application of the algorithm is image color clustering in web image analysis. Web images are usually supplied with shaded or multicolored complex backgrounds, often found in photographs, maps, engineering drawings and commercial advertisements. Analyzing web images is a challenging task due to their low spatial resolution and the large number of colors in the images. The purpose of color clustering in web image analysis is to reduce thousands of colors to a representative few that clearly differentiate objects of interest in an image. In this paper, we propose two EMST based clustering algorithms to address the issues—undesired clustering structures and an unnecessarily large number of clusters, commonly faced by the SEMST and the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2016

ZEMST algorithm respectively. First we will take a set of points of any given k and produces a k-partition of them. It produces k clusters with centre and guaranteed intra-cluster similarity. This process is repeated until k clusters are produced. In the second algorithm maximizing the overall standard deviation reduction, in order to produce a group of clusters by partitions a point set, without a given k value. In this paper we used both the algorithms to find Informative Meta similarity clusters.

II. RELATED WORK

The simplest k-constrained EMST-based algorithm is to remove k - 1 edges from the EMST, resulting in k subtrees. Each cluster is the set of points in each subtree. In the remaining paper, we will refer to this algorithm as SEMST.
Zahn [12] describes a method to remove *inconsistent* edges—edges, whose weights are significantly larger than the average weight of nearby edges—from the EMST. His definition of inconsistent edges relies on the concept of *depth-d neighborhoods*, N_1 and N_2, for each incident point, v_1 and v_2, of an edge e. The neighborhood of v_1 is the set of edges on paths from v_1 having length no greater than d, and excluding the edge e. Let $\bar{w}_{N1}$ be the average weight, and $_{N1}$ be the standard deviation, of neighborhood N_1, with similar definitions for N_2. He provides several alternative inconsistency criteria:

(1) $w > \bar{w}_{N1} + c \times \sigma_{N1}$ or $w > \bar{w}_{N2} + c \times \sigma_{N2}$
(2) $w > \max(\bar{w}_{N1} + c \times \sigma_{N1}; \bar{w}_{N2} + c \times \sigma_{N2})$
(3) $w/\max(c \times \bar{w}_{N1}; c \times \bar{w}_{N2}) > f$

The values d, c and f are user-assigned tuning parameters. Each cluster contains the points within each subtree resulting from the removal of inconsistent edges from the EMST. We will denote this algorithm as ZEMST. Bansal, Blum and Chawla [13] introduced an unconstrained algorithm called correlation clustering. The clustering problem they consider is a complete graph in which each edge is labelled qualitatively as "+" (similar) or "-" (dissimilar). The objective is to find the clustering that minimizes the number of disagreements with the edge labels. They proved NP-hardness of the fundamental problem, but provided approximation algorithms which have been further improved by others [15, 14]. Clustering algorithm proposed by S.C.Johnson [19] uses proximity matrix as input data. The algorithm is an agglomerative scheme that erases rows and columns in the proximity matrix as old clusters are merged into new ones. The algorithm is simplified by assuming no ties in the proximity matrix. Graph based algorithm was proposed by Hubert [17] using single link and complete link methods. He used threshold graph for formation of hierarchical clustering. An algorithm for single-link hierarchical clustering begins with the minimum spanning tree (MST) for G (∞), which is a proximity graph containing *n(n-1)/2* edge was proposed by Gower and Ross [18]. Later Hansen and DeLattre [16] proposed another hierarchical algorithm from graph coloring.
Eldershaw and Hegland [20] re-examine the limitations of many clustering algorithms that assume the underlying clusters of a data set are spherical. They provide a broader definition of a cluster based on the rule of transitivity: if two points p_1 and p_2 are close to the same point p_0, p_1 and p_2 are themembers of the same cluster in which they are indirectly related through p_0, despite the fact that p_1 and p_2 are not constrained by distance measure. They present a clustering algorithm by constructing a graph using Delaunay triangulation, and removing the edges between neighbors that are longer than a cut-off point. Next, they apply a graph partitioning algorithmto find the isolated connected components in the graph, and each discovered component is treated as a cluster.
Similar to Zahn'sMST clustering algorithm, this algorithm divides a point set into a certain number of clusters at once by removing all edges in the graph that are longer than a threshold. Unlike in Zahn's method, they choose a cut-off point which corresponds to the "global" minimum of a function that measures how well the consistent edges and the inconsistent edges in the graph are separated. Lopresti and Zhou [21] suggest an RGB color clustering method by constructing a Euclidean minimum spanning tree. Each distinct color in a given image is considered as a point in the three dimensional RGB color space. Thus each color is a node in the EMST. The weight of an edge is the Euclidean distance between two color nodes in the tree. They compute the average distance of the edges in the EMST once it is built. Subsequently the edges that are "longer" than the average weight by a predetermined amount are removed from the tree, leaving a set of disjoint subtrees. Colors in each subtree are the members of a color cluster. They point out that the EMST based color clustering algorithm may fail when dealing with textures and when there are a large number of colors in an image.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2016

III. PROPOSED CLUSTERING ALGORITHM

By using 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. A tree is a simple structure for representing binary relationship, and any connected components of tree is called *subtree.* Representing a set of multi-dimensional data points as simple tree structure will clearly lose some of the inter data relationship. 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. 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. In this section we preset clustering algorithm which produce *k* clusters, with center, radius, diameter and variance of each cluster. We also present another algorithm to find the hierarchy of *k* clusters and central cluster.

A. Measuring the Cluster Tightness and Compactness

The cluster tightness measure is a within – cluster estimate of clustering effectiveness , however it is possible to devise inter- cluster measure also, to better measure the separation between the various clusters.The Cluster compactness measure is based on the variance of the data points distributed in the subtrees (clusters). The variance of cluster *T* is computed as:

$$\nu(T) = \left(\frac{1}{n} \sum_{i=1}^{n} d^2(x_i, x_0) \right)^{1/2}$$

Where $d(x_i, x_j)$ is distance metric between two points(objects) xi and x_j, where n is the number of objects in the subtree Ti., and x_0 is the mean of the subtree *T*. A smaller the variance value indicates, a higher homogeneity of the objects in the data set, in terms of the distance measure d ( ). Since d ( ) is the Euclidean distance, v (T_i) becomes the statistical variance of data set σ (T_i). The cluster compactness for the output clusters generated by the algorithm is the defined as:

$$Cmp = \frac{1}{k} \sum_{i=1}^{k} \frac{\nu(I_i)}{\nu(S)}$$

Where *k* is the number of clusters generated on the given data set S, v (T_i) is the variance of the clusters T_i and V(S) is the variance of data set S.

IV. CLUSTERING ALGORITHM: RDEMST

This algorithm finds the radius, diameter and variance of subtrees, by using these we can easily find out the tightness and compactness of clusters. Hence we named the new algorithm as Radius and Diameter based Euclidean Minimum Spanning Tree algorithm (**RDEMST).**
Algorithm is defined as follow:

Algorithm: RDEMST (*k*)
Parameters are:

Input: point set is *S*
Output: number of clusters are *k* with C (represents set of center points)
e :edge in the **EMST** constructed from S
W_e :be the weight of e
σ :be the standard deviation of the edge weights
S_T :be the set of disjoint subtrees of the
EMST

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2016

n_c : be the number of clusters
Steps are:
1. First construct an **EMST** from S
2. Calculate the average weight of $\hat{W}$ of all the edges
3. Calculate the standard deviation σ of the edges
4. Calculate the variance of the point set S
5. S_T = ø; n_c = 1; C= ø;
6. **Repeat**
7. **for** each e ∈**EMST**
8. If ($W_e > \hat{W} + \sigma$) or (current longest edge e)
9. Remove e from **EMST**
10. the new disjoint subtree is $S_T = S_T \cup \{T'\}$ // T''
11. $n_c = n_c+1$
12. Using eccentricity of points calculate the center C_i of T_i
13. Using eccentricity of points calculate the diameter of T_i
14. Calculate the variance of T_i
15. $C = U_{Ti}\ S_T \{Ci\}$
16 Until $n_c = k$
17. **Return** k clusters with *C*

The variance for each subtree (cluster) is computed to find the compactness of clusters. A smaller the variance value indicates, a higher homogeneity of the objects in the data set. The cluster compactness measure evaluates how well the subtrees (clusters) of the input is redistributed in the clustering process, compared with the whole input set, in terms of data homogeneity reflected by Euclidean distance metric used by the clustering process. Smaller the cluster compactness value indicates a higher average compactness in the output clusters. Figure 1 illustrate a typical example of cases in which simply remove the *k*-1 longest edges does not necessarily output the desired cluster structure. Our algorithm finds the center, radius, diameter and variance of the each cluster which will be useful in many applications.

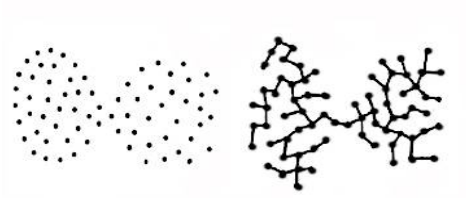

Figure1. Clusters connected through a point

V. CONCLUSION

This algorithm finds the radius, diameter and variance of clusters using eccentricity of points in a cluster. The radius and diameter value gives the information about tightness of clusters. The variance value of the cluster is useful in finding the compactness of cluster. This information will be very useful in many applications. Our **RDEMST** algorithm automatically determines the desired number of clusters. Our **RDEMST** clustering algorithm assumes a given cluster number. The algorithm gradually finds *k* clusters with center for each cluster. These *k* clusters ensures guaranteed intra-cluster similarity. The algorithm finds radius, diameter and variance of clusters using eccentricity of points in a cluster. The radius and diameter value gives the information about tightness of clusters and defined to maximize the overall standard deviation reduction. In the future, we want to explore and test our proposed clustering algorithms in various domains.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 8, August 2016

REFERENCES

[1] 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.
[2] H. Gabow, T. Spencer, and R. Tarjan. Efficient algorithms for finding minimum spanning trees in undirected and directed graphs. *Combinatorica*, 6(2):109–122, 1986.
[3] D. Karger, P. Klein, and R. Tarjan. A randomized lineartime algorithm to find minimum spanning trees. *Journal of the ACM*, 42(2):321–328, 1995.
[4] J. Kruskal. On the shortest spanning subtree and the travelling salesman problem. In *Proceedings of the American Mathematical Society*, pages 48–50, 1956.
[5] J. Nesetril, E. Milkov´a, and H. Nesetrilov´a. Otakar boruvka on minimum spanning tree problem: Translation of both the 1926 papers, comments, history. *DMATH: Discrete Mathematics*, 233, 2001.
[6] M. Halkidi, Y. Batisakis and M. Vazirgiannis: "Cluster validity methods": *part II, SIGMOD Rec.*, Vol. 31, No.3., pp.19- 27,2002
[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*, Pages 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] R.Prim. "Shortest connection networks and some generalization". *Bell systems Technical Journal*,36:1389-1401, 1957.
[11] C.Zahn. "Graph-theoretical methods for detecting and describing gestalt clusters". *IEEE Transactions on Computers*, C-20:68-86, 1971.
[12] C. Zahn. Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on Computers*, C-20:68–86, 1971.
[13] N. Bansal, A. Blum, and S. Chawla. Correlation clustering. In *Proceedings of the 43rd FOCS*, pages 238–247, 2002.
[14] M. Charikar, V. Guruswami, and A. Wirth. Clustering with qualitative information. *Journal of Computer and System Sciences*, 71:360–383, 2005.
[15] E. Demaine and N. Immorlica. Correlation clustering with partial information. In *Proceedings of the 6th RANDOM-APPROX*, pages 1–13, 2003.
[16] P. Hansen and M. Delattre " Complete-link cluster analysis by graph coloring" *Journal of the American Statistical Association* 73, 397-403, 1978.
[17] Hubert L. J " Min and max hierarchical clustering using asymmetric similarity measures " *Psychometrika* 38, 63-72, 1973.
[18] J.C. Gower and G.J.S. Ross "Minimum Spanning trees and single-linkage cluster analysis" *Applied Statistics* 18, 54-64, 1969.
[19] S. C. Johnson "Hierarchical clustering schemes" *Psychometrika* 32, 241-254, 1967.
[20] C. Eldershaw and M. Hegland. Cluster analysis using triangulation. In B. Noye, M. Teubner, and A. Gill, editors, *Computational Techniques and Applications: CTAC97*, pages 201–208. World Scientific, 1997.
[21] D. Lopresti and J. Zhou. Locating and recognizing text in www images. *Information Retrieval*, 2:177–206, 2000.