RESEARCH ARTICLE OPEN ACCESS

A Survey on Novel Graph Based Clustering and Visualization Using Data Mining Algorithm

M. Guruprasath [1], M. M. Elamparithi [2]
Research Scholar [1], Assistant Professor [2]
Department of Computer Science
Sree Saraswathi Thyagaraja College, Pollachi
Tamil Nadu –India

ABSTRACT

As the amount of data stored by various information systems grows very fast, there is an urgent need for a new generation of computational techniques and tools to assist humans in extracting useful information (knowledge) from the rapidly growing volume of data. Data mining (DM) is one of the most useful methods for exploring large data sets. Clustering, as a special area of data mining, is one of the most commonly used methods for discovering the hidden structure of the considered data set. The main goal of clustering is to divide objects into well separated groups in a way that objects lying in the same group are more similar to each other than to objects in other groups. If the data set to be analyzed contains many objects, the computation of the complete weighted graph requires too much execution time and storing space. To reduce the time and space complexity many algorithms work only with sparse matrices, and thereby do not utilize the complete graph. The suggested visualization method is called Topology Representing Network Map. The primary aim of this analysis was to examine the preservation of distances and neighborhood relations of data.

Keywords:- Data Cluster, Graph based Cluster, data Visualization

.I. INTRODUCTION

As the amount of data stored by various information systems grows very fast, there is an urgent need for a new generation of computational techniques and tools to assist humans in extracting useful information (knowledge) from the rapidly growing volume of data. Data mining (DM) is one of the most useful methods for exploring large data sets. Clustering, as a special area of data mining, is one of the most commonly used methods for discovering the hidden structure of the considered data set. The main goal of clustering is to divide objects into well separated groups in a way that objects lying in the same group are more similar to each other than to objects in other groups. Clustering can be used to quantize the available data, to extract a set of cluster prototypes for the compact representation of the data set, to select the relevant features, to segment the data set into homogenous subsets, and to initialize regression and classification models.

II. GRAPH BASED CLUSTERING

Jaromczyk and Toussaint pointed out that graph based methods are the most powerful methods of clustering in difficult problems, which give results having the best agreement with human performance [1]. In accordance with this statement there have been many graph based clustering algorithms developed in recent years [2, 3, 4, 5, 6]. In graph based clustering methods objects are considered as vertices of a graph, while edges between them are treated differently by the various approaches. In the simplest case, the graph is a complete graph, where all vertices are connected to each other, and the edges are labeled according to the degree of the similarity of the objects. Consequently, in this case the graph is a weighted complete graph. If the data set to be analyzed contains many objects, the computation of the complete weighted graph requires too much execution time and storing space. To reduce the time and space complexity many algorithms work only with sparse matrices, and thereby do not utilize the complete graph. The sparse similarity matrices contain information only about a small subset of the edges, mostly those corresponding to higher similarity values. The accentuation of the most similar vertices has the effect that the sparse similarity matrix expresses spatial proximity and thereby only objects placed near each other are connected with edges on the graph. We can say that sparse matrices encode the most relevant similarity values, and graphs based on these matrices visualize these similarities in a graphical way. Another way to reduce the time and space complexity is the application of a vector quantization (VQ) method (e.g. k-means [7], neural gas (NG) [8], Self- Organizing Map (SOM) [9]). The main goal of the vector quantization is to represent the entire set of objects by a set of representatives (codebook vectors), whose cardinality is much lower than the cardinality of the original data set. If a vector quantization method is used to reduce the time and space complexity, and the clustering method is based on graph-theory, vertices of the graph represent the codebook vectors and the edges denote the connectivity between them. Weights assigned to the edges express similarity of pairs of objects. Similarity measures corresponding to the labels of the edges are stored in a similarity matrix, and they can be calculated in different ways. The type of the objects' similarity can be divided into two main categories: (i) structural information and (ii) distance information. Structural information of the edges expresses the degree of the connectivity of the vertices (e.g. number of common neighbors). Weights based on distance information are arising from different distance measures, which reveal the distance of the connected objects (e.g Euclidean distance). This paper suggests using a new distance measure to label the edges of the graph. The suggested measure stores structural information about the similarity of the vertices and it can be used in arbitrary clustering algorithm.

The key idea of the graph based clustering is extremely simple: compute a graph of the original points or its representatives, and then delete any edge in the graph according to some criteria. The result is an unconnected graph and each sub graph represents a cluster. Using graphs for clustering we are interested in finding the edges, whose elimination leads to the best clustering result. Such edges are called inconsistent edges. In the simplest case those edges are removed step by step that have the largest weight value. But the elimination of the edges opens the door to other possibilities, as well.

III. DATA VISUALIZATION

The visualization of the data set plays an important role in the knowledge discovery process. In practical data mining problems usually high-dimensional data is to be analyzed. Data can be thought of as points in a high-dimensional vector space, with each dimension corresponding to an attribute of the observed object. Because humans simply cannot see high-dimensional data, it is necessary to reduce the dimensionality of the data. In most of these cases it is very informative to map and visualize the hidden structure of the complex data set in low-dimensional vector space.

Exploratory Data Analysis (EDA) is an approach for data analysis that employs a variety of techniques (mostly graphical) to maximize insight into a data set, uncover underlying structure, extract important variables, detect outliers and anomalies, and test underlying assumptions. The seminal work in EDA is written by Tukey [10]. Most EDA techniques are graphical in nature with a few quantitative techniques [11]. The role of EDA is to open-mindedly explore the data. Exploratory data analysis should ideally be a non-parametric method which is very often accompanied by heavy computational cost [12]. There are several methods to visualize the high dimensional data in two dimensional vector space (e.g. scatterplot, multidimensional scaling (MDS) [13, 14, 15], Principal Component Analysis (PCA) [16, 17], Sammon mapping (SM) [18], Kohonen's Self-Organizing Maps [19], etc.). The scatterplot (scatter diagram) visualizes a relation between two variables, each one corresponding to a different one of the variables of the objects. The data points are represented in two dimensional spaces, where axes represent the variables. The scatterplot is a useful tool to identify potential associations between two variables, but this representation method does not necessarily reflect the true nature of the structure of the data set. An alternative approach is to project the data set or its representatives into a low-dimensional vector space in such a way that it preserves their structure as fully as possible. The reduction of dimensionality of the feature space also is important because of the curse of dimensionality. In a nutshell, the same number of examples fills more of the available space when the dimensionality is low, and its consequence is that exponential growth with dimensionality in the number of examples is required to accurately estimate a function. Hence, dimensionality reduction is an important task of (visual) data mining. There are many dimensionality reduction methods which preserve as much of the intrinsic structure of the data as possible. The combination of the graph based clustering methods with the low dimensional visualization techniques guarantees a powerful tool for the analyzers. Applying such methods the analyzers can see not only the objects or its representatives in the low-dimension, but the intrinsic connections also show themselves.

IV. MOTIVATION

In the last decades a large number of clustering techniques have been developed. In different clustering algorithms different similarity or distance measures are used. The choice of the distance measure may significantly influence the opportunities and the limitations of the clustering methods. For example clustering algorithms based on the Euclidean distance are able to uncover only the well separated or compact clusters, but they work not so well if the clusters overlap each other or there is a bridge (chain link) between them. Thereby the choice of the applied similarity measure is a key point. While some similarity measures are grounded on the spatial distances of the n-dimensional vector points, other similarity measures utilize the neighborhood relations of the objects. The neighborhood clustering algorithm is one of those methods that are based on the neighborhood relations of the objects. It utilizes the number of the common neighbors of the k-nearest neighbors of the objects to disclose the clusters. The neighborhood algorithm first calculates the k nearest neighbors for each object to be clustered, and then it places the objects into the same cluster if they are contained in each other's k-neighbor list and they have at least l nearest neighbors in common.

V. FUZZY NEIGHBORHOOD SIMILARITY MEASURE

There are several ways to apply the fuzzy neighborhood similarity or distance matrix. For example, hierarchical clustering methods work on similarity or distance matrices. Generally, these matrices are obtained from the Euclidian distances of pairs of objects. Instead of the other similarity/distance matrices, the hierarchical methods can also utilize the fuzzy neighborhood similarity/distance matrix. The dendrogram not only shows the whole iteration process, but it can also be a useful tool to determine the number of the data groups and the threshold of the separation of the clusters. To separate the clusters we suggest to draw the fuzzy neighborhood similarity based dendrogram of the data, where the long nodes denote the proper threshold to separate the clusters.

The visualization of the objects may significantly assist in revealing the clusters. Many visualization techniques are based on the pair wise distance of the data. Because multidimensional scaling methods work on dissimilarity matrices, this method can also be based on the fuzzy neighborhood distance matrix.

Calculation of the transitive fuzzy neighborhood similarity measure Algorithm

Step-1: Given a set of data X, specify the number of the maximum clusters r_{max}, and choose a first-order filter parameter α. Initialize the fuzzy neighborhood similarity matrix as S(0) = 0.

Step-2: Repeat for r = 1, 2, …., r_{max}

Step-3: Calculate the fuzzy neighborhood similarities for each pair of objects as follows:

$$S_{i,j}^{(r)} = \frac{\left|A_i^{(r)} \cap A_j^{(r)}\right|}{\left|A_i^{(r)} \cup A_j^{(r)}\right|}$$

where set $A_i^{(r)}$ denotes the r-order k-nearest neighbors of object x_i ε X, and $A_j^{(r)}$ respectively for x_j ε X.

Step-4: Update the fuzzy neighborhood similarity measures based on the following formula:

$$S_{i,j}^{(r)} = (1\text{-}\alpha)\, S_{i,j}^{(r-1)} + \alpha\, S_{i,j}^{(r)}$$

Finally $S_{i,j}^{(r_{max})}$ yields the fuzzy neighborhood similarities of the objects.

As a result of the whole process a fuzzy neighborhood similarity matrix (S) will be given, which summarizes the pair wise fuzzy neighborhood similarities of the objects. The fuzzy neighborhood distance matrix (D) of the objects is obtained by the formula: D = 1 - S. Naturally, both the fuzzy neighborhood similarity and the fuzzy neighborhood distance matrices are symmetric matrices, that is $S^T = S$ and $D^T = D$.

The computation of the proposed transitive fuzzy neighborhood similarity/distance measure includes the following three parameters: k, r_{max} and α. The choice of the value of these parameters has an affect on the separation of clusters. Lower values of parameter k (e.g k = 3) separate the clusters better. By increasing value k clusters that overlap in some objects become more similar. The higher the value of parameter r_{max} is, the higher the similarity measure of similar objects becomes. The increase of the value r_{max} results in more compact clusters. The lower the value of α, the less the affect of neighbors far away becomes. As the fuzzy neighborhood similarity measure is a special case of the transitive fuzzy neighborhood similarity measure in the following these terms will be used as equivalent.

The Variety data set is a synthetic data set which contains 100 2-dimensional data objects. 99 objects are partitioned in 3 clusters with different sizes (22, 26 and 51 objects), shapes and densities, and it also contains an outlier. Figure-1 shows some results of Neighborhood clustering applied on the normalized data set. The objects belonging to different clusters are marked with different markers. In these cases the value of parameter k was fixed to 8, and the value of parameter l was changed from 2 to 5. (The parameter settings k = 8, l = 2 gives the same result as k = 8 and l = 3.) It can be seen that the Neighborhood clustering algorithm was not able to identify the clusters in any of the cases. The cluster placed in the upper right corner in all cases is split into sub clusters. When parameter l is low (l = 2, 3, 4) the algorithm is not able to detect the outlier. When parameter l is higher, the algorithm detects the outlier, but the other clusters are split into more sub clusters. After multiple runs of the JP algorithm there appeared a clustering result, where all objects were clustered according to expectations. This parameter setting was k = 10 and l = 5. To show the complexity of this data set in Figure-2 the result of the well-known k-means clustering is also presented(the number of the clusters are 4). This algorithm is not able to disclose the outlier; thereby the cluster with small density is split into two sub clusters. Table-1 summarizes the clustering rates of the previously presented algorithms. The clustering rate was calculated as the fraction of the number of well clustered objects and the total number of objects.

Neighborhood Algorithm	Cluster Rate
k=8 and l=3	95 %
k=8 and l=4	98 %
k=8 and l=5	65 %
k=10 and l=5	100 %

Table-1: Clustering rates for different mappings of the Variety data set

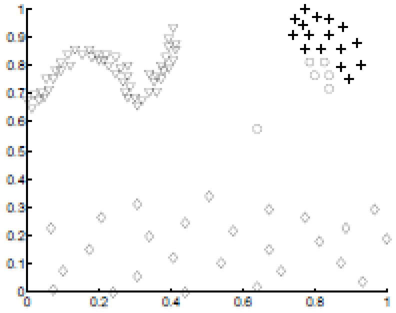

Figure-1(a): k=8 and l=3

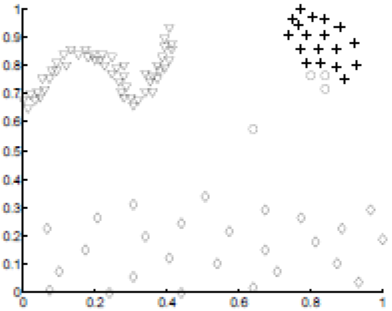

Figure-1(b): k=8 and l=4

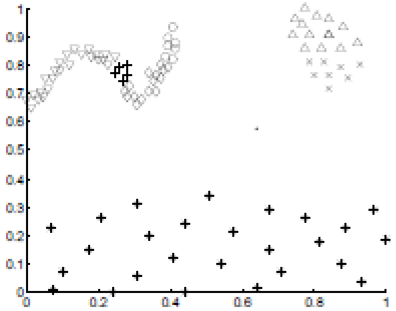

Figure-1(b): k=8 and l=5

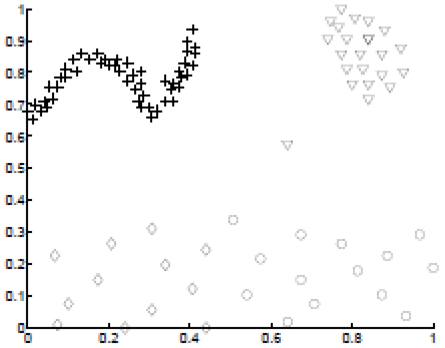

Figure-2: Result of the k-means clustering on the normalized Variety data set

The proposed fuzzy neighborhood similarity measure was calculated with different k, r_{max} and α parameters. Different runs with parameters k = 3, …, 25, and r_{max} = 2, …, 5 and α= 0.1, …, 0:4 have been resulted in good clustering outcomes. If a large value is chosen for parameter k, it is necessary to keep parameter r_{max} on a small value to avoid merging the outlier object with one of the clusters.

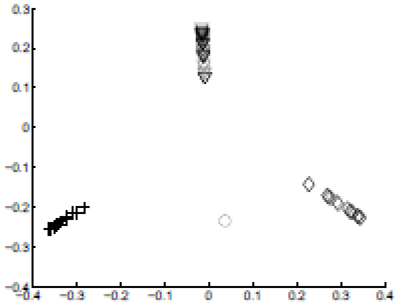

Figure-3(a): MDS based on the fuzzy neighborhood distance matrix

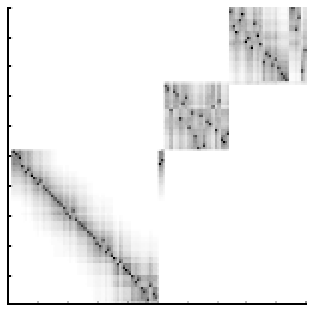

Figure-3(b): VAT based on the single linkage fuzzy neighborhood distances

To show the fuzzy neighborhood distances of the data, the objects are visualized by multidimensional scaling and VAT. Figure-3(a) shows the MDS mapping of the fuzzy neighborhood distances with the parameter settings: $k = 6$, $r_{max} = 3$ and $\alpha = 0.2$. Other parameter settings have also been tried, and they show similar results to Figure-3(a). Figure-3(b) shows the VAT representation of the data set based on the single linkage fuzzy neighborhood distances. The three clusters and the outlier are also easily separable in this figure. To find the proper similarity threshold to separate the clusters and the outlier the dendrogram based on the single linkage connections of the fuzzy neighborhood distances of the objects shown in Figure-4 has also been drawn. The dendrogram shows that the value $d_{i,j} = 0.75$ $(d_{i,j} = 1-s_{i,j})$ is a suitable choice to separate the clusters and the outlier from each other ($k = 6$, $r_{max} = 3$ and $\alpha = 0.2$). Applying a single linkage agglomerative hierarchical algorithm based on the fuzzy neighborhood distances, and halting this algorithm at the threshold $d_{i,j} = 0.75$ the clustering rate is 100%. In other cases ($k = 3,\ldots,25$) $r_{max} = 2,\ldots, 5$ and $\alpha = 0.1,\ldots, 0.4$, and if the value of parameter k was large, the parameter r_{max} was kept on low values) the clusters also were easily separable and the clustering rate obtained was 99 to 100%.

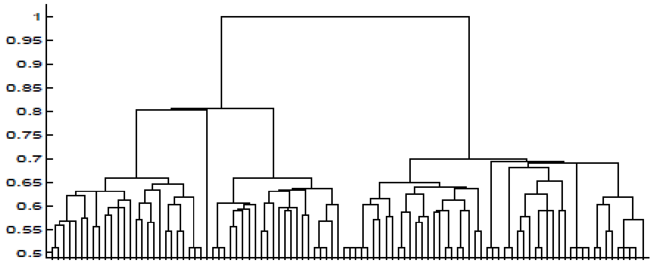

Figure-4: Single linkage dendrogram based on the fuzzy neighborhood distances (Variety data set)

This simple example illustrates that the proposed fuzzy neighborhood similarity measure is able to separate the clusters with different sizes, shapes and densities; furthermore it is able to identify outliers.

VI. CONCLUSION

The fuzzy neighborhood similarity measure extends the similarity measure of the neighborhood algorithm in two ways: (i) it takes into account the far neighbors partway and (ii) it fuzzifies the crisp decision criterion of the neighborhood algorithm. The fuzzy neighborhood similarity measure is based on the common neighbors of the objects, but differently from the neighborhood algorithm it is not restricted to the direct neighbors. While the fuzzy neighborhood similarity measure describes the similarities of the objects, the fuzzy neighborhood distance measure characterizes the dissimilarities of the data. The values of the fuzzy neighborhood distances are easily computable from the fuzzy neighborhood similarities. The application possibilities of the proposed measures are widespread. All methods that work on distance or similarity measures can also be based on the fuzzy neighborhood similarity/distance measures. We have introduced the application possibilities of the fuzzy neighborhood similarity and distance measures in hierarchical clustering and in VAT representation. It was demonstrated through an application example that clustering methods based on the proposed fuzzy neighborhood similarity/distance measure can discover clusters with arbitrary shapes, sizes and densities. Furthermore, the proposed fuzzy neighborhood similarity/distance measure is able to identify outliers, as well.

REFERENCES

[1] J.W. Jaromczyk and G.T. Toussaint. Relative neighborhood graphs and theirrelatives. Proceedings of the IEEE, 80(9):1502–1517, 1992.

[2] N. Chen, A. Chen, L. Zhou, and L. Lu. A graph-based clustering algorithmin large transaction. Intelligent Data Analysis, 5(4):327–338, 2001.

[3] S. Guha, R. Rastogi, and K. Shim. Rock: a robust clustering algorithm for categorical attributes. In Proceedings of the 15th International Conference On Data Engeneering, pages 512–521, 1999.

[4] X. Huang and W. Lai. Clustering graphs for visualization via node similarities.Journal of Visual Languages and Computing, 17:225–253, 2006.

[5] S. Kaski, J. Nikkilä, M. Oja, J. Venna, J. Toronen, and E. Castrén. Trustworthiness and metrics in visualizing similarity of gene expression. BMC Bioinformatics, 4(48), 2003.

[6] M.J. Zaki, M. Peters, I. Assent, and T. Seidl. Clicks: An effective algorithm for mining subspace clusters in categorical datasets. Data and Knowledge Engineering, 60:51–70, 2007.

[7] J. McQueen. Some methods for classification and analysis of multivariate observations. In Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability, pages 281–297, 1967.

[8] T.M. Martinetz and K.J. Schulten. Artificial Neural Networks, chapter A neural-gas network learns topologies, pages 397–402. North-Holland, Amsterdam, 1991.

[9] T. Kohonen. Self-Organizing Maps. Springer, third edition, 2001.

[10] J. Tukey. Exploratory Data Analysis. Addison-Wesley, 1977.

[11] J. Militk and M. Meloun. Some graphical aids for univariate exploratory data analysis. Analytica Chimica Acta, 277(2):215–221, 1993.

[12] I. Borg and P. Groenen. Modern Multidimensional Scaling: Theory and Applications. Springer Series in Statistics. Springer Verlag, New York, 1997.

[13] J. Leeuw and W. Heiser. Handbook of Statistics, volume 2, chapter Theory of multidimensional scaling, pages 285–316. North-Holland, Amsterdam, 1982.

[14] M. Wish and J.D. Carroll. Handbook of Statistics, volume 2, chapter Multidimensional scaling and its applications, pages 317–345. North-Holland, Amsterdam, 1982.

[15] H. Hotelling. Analysis of a complex of statistical variables into principal components. Journal of Education Psychology, 24:417–441, 1933.

[16] T. Jolliffe. Principal Component Analysis. Springer, New York, 1996.

[17] J.W. Sammon. A non-linear mapping for data structure analysis. IEEE Transactions on Computers, 18(5):401–409, 1969.

[18] T. Kohonen. Self-Organizing Maps. Springer, third edition, 2001.