A Review on Clustering Algorithms

Angeline Christobel. Y
College of Computer Studies
AMA International University, Bahrain

Suresh Subramanian
Ahlia University
Bahrain

Dr. Minerva M. Bunagan
College of Computer Studies
AMA International University, Bahrain

Dr. Siva Prakasam
Department of Computer Science
Sri Vasavi College, Erode, India

Abstract— **In recent years, large amount of information hidden in huge databases has created tremendous interests in the field of data mining. Data mining is a field developed as a means of extracting information and knowledge from databases to discover patterns. Clustering is one of the primary tools in unsupervised learning. Clustering means creating groups of objects based on their features in such a way that the objects belonging to the same groups are similar and those belonging to different groups are dissimilar. In this paper we address the main algorithms used for clustering in different categories. We use the theoretical survey and recommendations for considerations to have in mind while choosing a specific method.**

Key Words: ***Data Mining, Clustering, Supervised Learning, Unsupervised Learning***

I. INTRODUCTION

Data mining has become a powerful information technology tool in today's competitive business world. The interest in data mining is increasing as the sizes and varieties of electronic datasets grow. Data mining is widely used to extract patterns in a large amount of data generated in applications such as location based services, sensor networks, scientific and biological databases. It uses a combination of statistics, probability analysis and database technologies. Data mining is fairly young but clever algorithms developed through database research. A popular data mining task is to segment a data set into groups, with each group acting as one unit for further analysis.

Data mining is divided into two primary sub fields: supervised learning and unsupervised learning. In Supervised learning, a training set of examples with the correct responses (targets) are provided and based on this training set, the algorithm generalizes to respond correctly to all possible inputs. This is also called learning from exemplars. Classification is a supervised data mining technique that is used to classify objects based on their features into a predefined category**.** In unsupervised learning, the algorithm is provided with the data points and no labels. The task is to find a suitable representation of the underlying distribution of the data. One of the primary tools of unsupervised learning is clustering. The main advantage of clustering is that interesting patterns and structures can be found directly from very large data sets with little or none of the background Knowledge. The goal of clustering is the interpretation of results which provides meaningful insights of original unclassified data. Figure1 from [21] shows all stages of clustering process.

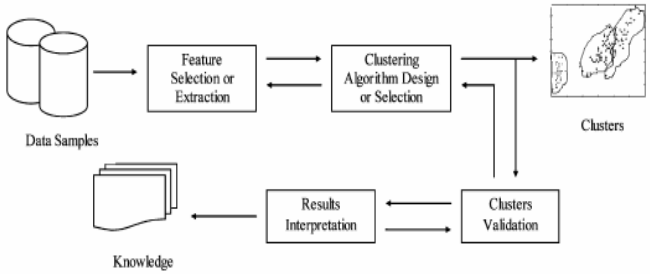

Fig1: The stages of clustering process.

Fig1 shows the procedure of cluster analysis with four basic steps.

1. *Feature Selection* is the process of identifying the most effective subset of the original features to use in clustering. *Feature extraction* is the use of one or more transformations of the input features to produce new significant features.
2. *Clustering algorithm design or selection* is usually combined with the selection of a corresponding proximity measure and the construction of a criterion function.
3. *Cluster Validation*. The goal of clustering is to determine the intrinsic grouping in a set of unlabeled data. But how to decide what constitutes a good clustering? The best criterion is heavily dependent of the final aim of the clustering. Consequently, it is the user which must supply this criterion, in such a way that the result of the clustering will suit their needs.
4. *Results Interpretation*. The goal of clustering is to provide users with meaningful insights from the original data, so that they can effectively solve the problems encountered. Experts in the relevant fields interpret the data partition. Further analyzes, even experiments, may be required to guarantee the reliability of extracted knowledge.

The widely used data clustering algorithms are focused in this paper.

II. CLUSTERING ALGORITHMS

Clustering is a data mining technique used to identify clusters based on the similarity between data objects. Traditionally, clustering is applied to unclassified data objects with the objective to maximize the distance between clusters and minimize the distance inside each cluster. Clustering is used in marketing, insurance, city-planning, biology, and earthquake studies. In marketing clustering helps them discover trends in their customer bases and develop marketing programs. In insurance, clustering helps to identify groups of motor insurance policy holders with a high average claim cost as well as to identify frauds. In city planning it helps identify groups of houses according to type, value, and location. Clustering helps biologists in the classification of plants and animals. Clustering algorithms deal with a set of objects whose positions are accurately known [3].

The requirements for a Good Cluster Analysis [7] is given below

- Scalability
- Ability to deal with different types of attributes
- Ability to find clusters of arbitrary shape
- Ability to deal with noise and outliers
- Incremental clustering and insensitivity to the order of input records.
- Ability to deal with high dimensionality

In [7], the clustering techniques are organized into the following categories:

1. Partitional clustering
2. Hierarchical clustering
3. Density-based methods
4. Grid based methods
5. Model based methods
6. Methods for high dimensional data
7. Constraint based clustering

A. Partitional Clustering

A partitioning method first creates initial *k* partitions, where parameter *k* is the number of partitions to construct. It then uses an iterative relocation technique that attempts to improve the partitioning by moving objects from one group to another. The partitional clustering algorithms such as K-means and Fuzzy C means are discussed below.

K-Mean Clustering algorithm

One of the best known and most popular clustering algorithms is the k-means algorithm. K-means clustering involves search and optimization.

K-means is a partition based clustering algorithm. K-means' goal is to partition data D into K parts, where there is little similarity across groups, but great similarity within a group. More specifically, K-means aims to minimize the mean square error of each point in a cluster, with respect to its cluster centroid.

Formula for Square Error:

$$\text{Square Error (SE)} = \sum_{i=1}^{k} \sum_{j=1}^{|ci|} \left(x_j - M_{ci} \right),$$

where k is the number of clusters, |ci| is the number of elements in cluster ci, and M_{ci} is the mean for cluster ci.

Steps of K-Means Algorithm

The k Means algorithm is explained in the following steps. The algorithm normally converges in short iterations. But will take considerably long time for iteration if the number of data points and the dimension of each data are high.

Step 1: Choose k random points as the cluster centroids.

Step 2: For every point p in the data, assign it to the closest centroid. That is compute d(p, M_{ci}) for all clusters, and assign p to cluster C* where distance

(d(P, M_{c*}) <= d(P, M_{ci}))

Step 3: Recompute the center point of each cluster based on all points assigned to the said cluster.

Step 4: Repeat steps 2 & 3 until there is convergence. (Note: Convergence can mean repeating for a fixed number of times, or until SE_{new} - SE_{old} <= ε, where ε is some small constant, the meaning being that we stop the clustering if the new SE objective is sufficiently close to the old SE.)

Fuzzy C-Means Clustering Algorithm (FCM)

In Fuzzy-C-Means, the data point can belong to more than one cluster. Also fuzzy clustering gives the degree to which each point belongs to the other clusters. The Fuzzy C-means clustering algorithm is the most widely used fuzzy clustering algorithm. This method was first developed by Dunn in 1973 and improved by Bezdek in 1981.

Algorithmic steps for Fuzzy c-means clustering

Let X = {x_1, x_2, x_3 ..., x_n} be the set of data points and V = {v_1, v_2, v_3 ..., v_c} be the set of centers.

1) Randomly select *'c'* cluster centers.

2) Calculate the fuzzy membership $'\mu_{ij}'$ using:

$$\mu_{ij} = 1 / \sum_{k=1}^{c} (d_{ij} / d_{ik})^{(2/m-1)}$$

3) Compute the fuzzy centers $'v_j'$ using:

$$v_j = (\sum_{i=1}^{n} (\mu_{ij})^m x_i) / (\sum_{i=1}^{n} (\mu_{ij})^m), \forall j = 1, 2, ...c$$

4) Repeat step 2) and 3) until the minimum $'J'$ value is achieved or $||U^{(k+1)} - U^{(k)}|| < \beta$.

where,

'k' is the iteration step.

'β' is the termination criterion between [0, 1].

'$U = (\mu_{ij})_{n*c}$' is the fuzzy membership matrix.

'J' is the objective function.

B. Hierarchical Clustering

This method builds the hierarchy by progressively merging the clusters. Hierarchical clustering works in two ways it builds or breaks up a hierarchy of clusters. The method of building up is called agglomerative and the method of breaking up is called divisive. Agglomerative begins at the top where divisive begins at the bottom.

Agglomerative clustering algorithm

1. Start by assigning each item to its own cluster, so that if you have N items, you now have *N* clusters, each containing just one item

2. Find the closest distance (most similar) pair of clusters and merge them into a single cluster, so that now you have one less cluster

3. Compute pair wise distances between the new cluster and each of the old clusters

4. Repeat steps 2 and 3 until all items are clustered into a single cluster of size *N*

5. Draw the dendrogram, and with the complete hierarchical tree, if you want *K* clusters you just have to cut the *K*-1 top links

Computing distances between clusters implemented in different ways:

Single-linkage clustering

The distance between one cluster and another cluster is computed as the *shortest* distance from any member of one cluster to any member of the other cluster.

Complete-linkage clustering

The distance between one cluster and another cluster is computed as the *greatest* distance from any member of one cluster to any member of the other cluster.

Centroid clustering

The distance between one cluster and another cluster is computed as the distance from one cluster *centroid* to the other cluster *centroid.*

C. Density Based Methods

Density based clustering methods cluster data based on a local cluster criterion such as density connected points. Typically, density based algorithms can discover clusters of arbitrary shapes and are relatively noise tolerant. The density based methods, DBSCAN and OPTICS are discussed below.

DBSCAN

Density based spatial clustering of applications with noise rely on a density-based notion of clusters, which is designed to discover clusters of arbitrary shape and also have ability to handle noise.

DBSCAN requires two parameters

- Eps: Maximum radius of the neighborhood
- MinPts: Minimum number of points in an Eps-neighborhood .

The clustering process is based on the classification of the points in the dataset as core points, border points and noise points and on the use of density relations between points directly density reachable, density reachable, density connected[Ester 1996] to form the clusters.

Core points:

The points that are at the interior of a cluster are called core points. A point is an interior point if there are enough points in its neighborhood.

Border points:

Points on the border of a cluster are called border points.

NEps(p): {q belongs to D | dist(p,q) <= Eps}

Noise points:

A noise point is any point that not a core point or a border point.

Directly Density-Reachable:

A point *p* is directly density-reachable from a point *q* with respect to *Eps*, *MinPts* if *p* belongs to *NEps(q)* *|NEps (q)| >= MinPts*

Density-Reachable:

A point *p* is density-reachable from a point *q* with respect to *Eps*, *MinPts* if there is a chain of points *p1*, …, *pn*, *p1 = q*, *pn = p* such that *pi+1* is directly density-reachable from *pi*

Density-Connected:

A point *p* is density-connected to a point *q* with respect to *Eps*, *MinPts* if there is a point *o* such that both, *p* and *q* are density-reachable from *o* with respect to *Eps* and *MinPts*.

Algorithm: The algorithm of DBSCAN is as follows (M. Ester, H. P. Kriegel, J. Sander, 1996)

- Arbitrary select a point *p*
- Retrieve all points density-reachable from *p* with respect to *Eps* and *MinPts*.
- If *p* is a core point, a cluster is formed.
- If *p* is a border point, no points are density-reachable from *p* and DBSCAN visits the next point of the database.

- Continue the process until all of the points have been processed.

OPTICS

This algorithm was presented by Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel and Jörg Sander. It is used to find density based clusters in spatial data. Although DBSCAN can cluster objects given input parameters such as Eps and MinPts, it still leaves the user with the responsibility of selecting parameter values that will lead to the discovery of acceptable clusters. Such parameter settings are usually empirically set and difficult to determine, especially for real world, high dimensional data sets. To overcome this difficulty, a cluster analysis method called OPTICS was proposed. Rather than produce a data set clustering explicitly, OPTICS computes an augmented cluster ordering for automatic and interactive cluster analysis. This ordering represents the density based clustering structure of the data. It contains information that is equivalent to density based clustering obtained from a wide range of parameter settings. The cluster ordering can be used to extract basic clustering information (such as cluster centers or arbitrary-shaped clusters) as well as provide the intrinsic clustering structure.

D. Grid Based Methods

A grid based method first quantizes the object space into a finite number of cells that form a grid structure and then performs clustering on the grid structure. The grid based methods, STING and WAVECLUSTER are discussed below.

STING

Wang et al. (1997) proposed a STatistical INformation Grid-based clustering method (STING) to cluster spatial databases. The algorithm can be used to facilitate several kinds of spatial queries. The spatial area is divided into rectangle cells, which are represented by a hierarchical structure. Let the root of the hierarchy be at level 1, its children at level 2, etc. The number of layers could be obtained by changing the number of cells that form a higher-level cell. A cell in level i corresponds to the union of the areas of its children in level i + 1. In the algorithm STING, each cell has 4 children and each child corresponds to one quadrant of the parent cell. Only two-dimensional spatial space is considered in this algorithm.

The STING algorithm [15] is

1. Construct the grid hierarchical structure according to the database and generate the parameters of each cell;
2. Determine a layer to begin with;
3. For each cell in this layer, compute the confidence interval of the probability that this cell is relevant to the query;
4. If this layer is not the bottom layer then
5. Go to the next level in the hierarchy structure and go to step 3 for the relevant cells of the higher-level layer;
6. Else if the specification of the query is met then
7. Find the regions of relevant cells and return those regions that meet the requirements of the query;
8. Else
9. Reprocess the data in the relevant cells and return the results that meet the requirements of the query;
10. End if

The algorithm cannot be scaled to high-dimensional databases. In high-dimensional data, if each cell has four children, then the number of cells in the second layer will be 2^d , where d is the number of dimensions of the database.

WAVE CLUTSER

Wave Cluster (Sheikholeslami et al., 2000) is an algorithm for clustering spatial data based on wavelet transforms. Wave Cluster is insensitive to noise, capable of detecting clusters of arbitrary shape at different degrees of detail, and efficient for large databases. The key idea of Wave Cluster is to apply wavelet transforms on the feature space, instead of the objects themselves, to find the dense regions in the feature space [15].

The algorithm [15] first partitions the original data space into non overlapping hyper rectangles, i.e., cells. The *j th* dimension is segmented into m_j of intervals. Each cell c_i is the intersection of one interval from each dimension and has the form $(c_{i1}, c_{i2}, \ldots, c_{id})$, where $c_{ij} = l_{ij}, h_{ij}$ is the right open interval in the partitioning of the *j th* dimension and *d* is the number of dimensions.

A point $x = (x_1, x_2, \ldots, x_d)$ is said to be contained in a cell x_i if $l_{ij} \le x_j < h_{ij}$ for $j = 1, 2, \ldots, d$. Let $c_i \cdot$ count denote the number of points contained in the cell *ci* . Then the algorithm applies a wavelet transform to c_i .count values. Then the transformed space is defined as the set of cells after the wavelet transformation on the count values of the cells in the quantized space.

In this algorithm, a cluster is defined to be a set of significant cells $\{c_1, c_2, \ldots, c_m\}$ that are k-connected in the transformed space. A cell is called a significant cell if its count in the transformed space is above a certain threshold τ . A cell c_1 is an ϵ neighbor of a cell c_2 if both are either significant cells (in transformed space) or nonempty cells (in quantized space) and $D(c_1, c_2) \le \epsilon$, where $D(c_1, c_2)$ is an appropriate distance metric and $\epsilon > 0$. A cell c1 is a k–ϵ-neighbor of a cell c_2 if both are significant cells or both are nonempty cells and if c1 is one of the *k* prespecified ϵ-neighbors of c_2. Two cells c_1 and c_2 are said to be k-connected if there is a sequence of cells $c_1 = p_1, p_2, \ldots, p_j = c_2$ such that p_{i+1} is a $k - \epsilon$-neighbor of p_i for $i = 1, 2, \ldots, j$.

WaveCluster is capable of finding arbitrarily shaped clusters and the number of clusters is not required in advance.

E. MODEL BASED METHODS

A model based method hypothesizes a model for each of the clusters and finds the best fit of the data to that model. The model based algorithms; EM and SOM are discussed below.

EM

Clustering by means of a mixture of Gaussians is a model-based approach. This type of clustering consists of using a model for the clusters and optimizing the fit between the data and the model. The algorithm which is used most to find the mixture of Gaussians is called EM (Expectation-Maximization).

First initialize the parameters, then the calculations are divided into two steps the E-step or expectation step and the M-step or maximization step.

In the expectation step plug in all the data into the algorithm and in the maximization step solve for the unknowns.

The main advantages of model-based clustering are that we obtain density estimation for each cluster, as well as having some flexibility when it comes to choosing the component distribution.

SOM

SOM's are one of the most popular neural network methods for cluster analysis. The Self Organizing Map (SOM) is developed by Professor Teuvo Kohonen in the early 1980's. It is a computational method for the visualization and analysis of high dimensional data.

A self organizing map consists of components called nodes. The nodes of the network are connected to each other, so that it becomes possible to determine the neighborhood of a node. Each node receives all elements of the training set, one at a time, in vector format. For each element, Euclidean distance is calculated to determine the fit between that element and the weight of the node. The weight is a vector of the same dimension as the input vectors. This allows to determine the "winning node", that is the node that represents the best training element. Once the winning node is found, the neighbors of the winning node are identified. The winning node and these neighbors are then updated to reflect the new training element.

It appears to be customary that both the neighborhood function and the learning rate are a decreasing function of time. This means that as more training elements are learned, the neighborhood is smaller and the nodes are less affected by the new elements.

We express this change as the following function: for a node x, the update is equal to

$x(t+1) = x(t) + N(x,t)\alpha(t)(\xi(t) - x(t))$

Where

$x(t+1)$ is the next value of the weight vector

$x(t)$ is the current value of the weight vector

$N(x,t)$ is the neighborhood function, which decreases the size of the neighbourhood as a function of time

$\alpha(t)$ is the learning rate, which decreases as a function of time

$\xi(t)$ is the vector representing the input document

Based on this information, the algorithm is given below.

Algorithm

1. Initialize the weights of the nodes, either to random or pre computed values
2. For all input elements:
    - Take the input, get its vector
    - For each node in the map: Compare the node with the input's vector
    - The node with the vector closest to the input vector is the winning node.
    - For the winning node and its neighbors, update them according to the formula above.

F. METHODS FOR HIGH DIMENSIONAL DATA

Clustering high dimensional data is of crucial importance, because in many advanced applications, data objects such as text documents and microarray data are high dimensional in nature. There are three typical methods to handle high dimensional data sets: dimension-growth subspace clustering represented by CLIQUE, dimension-reduction projected clustering, represented by PROCLUS and frequent pattern-based clustering represented by pCluster. CLIQUE and PROCLUS are discussed below.

CLIQUE(Clustering in QUEst)

This was the first algorithm proposed for dimension growth subspace clustering in high-dimensional space. In dimension growth subspace clustering, the clustering process starts at single-dimensional subspaces and grows upward to higher dimensional ones. Because CLIQUE partitions each dimension like a grid structure and determines whether a cell is dense based on the number of points it contains, it can also be viewed as an integration of density based and grid based clustering methods.

CLIQUE [7] performs multidimensional clustering in two steps:

In the first step, CLIQUE partitions the d-dimensional data space into non overlapping rectangular units, identifying the dense units among these. This is done for each dimension. The subspace representing these dense units are interested to form a candidate search space in which dense units of higher dimensionality may exist.

In the second step, CLIQUE automatically generates a minimal description for each cluster as follows. For each cluster, it determines the maximal region that covers the cluster of connected dense units. It then determines a minimal cover (logic description) for each cluster.

CLIQUE automatically finds subspace of the highest dimensionality such that high-density clusters exist in those subspaces. It is insensitive to the order of input objects and does not presume any canonical data distribution. It scales linearly with the size of input and has good scalability as the number of dimensions in the data is increased. However, obtaining meaningful clustering results is dependent on proper tuning of the grid size and the density threshold.

PROCLUS

PROCLUS (PROjected CLUStering) is a dimension reduction subspace clustering. That is, instead of starting from single

dimensional spaces, it starts by finding an initial approximation of the clusters in the high dimensional attribute space. Each dimension is then assigned a weight for each cluster, and the updated weights are used in the next iteration to regenerate the clusters. This leads to the exploration of dense regions in all subspaces of some desired dimensionality and avoids the generation of a large number of overlapped clusters in projected dimensions of lower dimensionality.

This algorithm consists of three phases:

1. In the initialization phase, it uses a greedy algorithm to select a set of initial medoids that are far apart from each other so as to ensure that each cluster is represented by atleast one object in the selected set.
2. The iteration phase selects a random set of *k* medoids from this reduced set (of medoids), and replaces "bad" medoids with randomly chosen new medoids if the clustering is improved. For each medoid, a set of dimensions is chosen whose average distances are small compared to statistical expectation. The total number of dimensions associated to medoids must be *k* x *l*, where l is an input parameter that selects the average dimensionality of cluster spaces.
3. The refinement phase computes new dimensions for each medoid based on the clusters found, reassigns points to medoids and removes outliers.

This method is efficient and scalable at finding high dimensional clusters.

G. Constraint based clustering

Constrained based clustering finds clusters that satisfy user specified preferences or constraints. Depending on the nature of the constraints, constrained based clustering may adopt different approaches. Here are a few categories of constraints.

1. Constraints on individual objects
2. Constraints on the selection of clustering parameters
3. Constraints on distance or similarity functions.
4. User-specified constraints on the properties of individual clusters.
5. Semi-supervised clustering based on "partial" supervision

Comparison of Clustering Methods

Table 1 shows the comparison of different clustering methods based on outlier handling, complexity and capability of handling high dimensional data.

Table 1: Comparison of Clustering Algorithms

Methods	**Algorithm**	**Outlier handling**	**Complexity**	**Capability of handling high dimensional data**
Partitioning	K-Means	No	O(lkn)	No
	Fuzzy C Means	Yes	$O(ndc^2i)$	No
Hierarchical	Hierarchical Clustering	No	$O(n^2)$	No
Density Based	DBSCAN	Yes	O (nlog n)	No
	OPTICS	Yes	O (k)	No
Grid Based	STING	Yes	O (n)+O(g) g:query number	No
	WAVE CLUSTER	Yes	O (n)	No
Model Based	EM	No	O (n)	No
High dimensional data	CLIQUE	Yes	O (n)	Yes

III. CONCLUSION

The cluster evaluation is quite subjective because the results can be interpreted in different ways. The user's need is an important factor while evaluating the clustering technique. The best technique provides the results that are useful for the user's purposes. When choosing the clustering technique, the factors that should be considered are the type of clustering techniques, characteristics of clusters, characteristics of the data set and attributes, noise and outliers, the number of data objects, the number of attributes, cluster description and algorithm consideration.

IV. REFERENCES

[1] R.A. Jarvis, E.A. Patrick. "Clustering Using a Similarity Measure Based on Shared Near Neighbors," IEEE Transactions on Computers. C22: 1025-1034, 1973.

[2] G.N. Lance, W.T. Williams. "A General Theory of Classificatory Sorting Strategies 1. Hierarchical Systems," Computer Journal, (9): 373-380 (1966).

[3] Ben Kao Sau, Dan Lee, David W. Cheung, Wai-Shing Ho, K. F. Chan, "Clustering Uncertain Data using Voronoi Diagrams", Eighth IEEE International Conference on Data Mining2008

[4] H.P.Kriegel and M.Pfeifle, "Density based clustering of uncertain data:, ACM KDD Conference(2005)

[5] THOMOPOULOS S., BOUGOULIAS D., and WANN, C-D. 1995. Dignet: "An unsupervised learning clustering algorithm for clustering and data fusion", IEEE Trans. on Aerospace.

[6] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2001). "8.5 The EM algorithm". The Elements of Statistical Learning. New York: Springer. pp. 236–243. ISBN 0-387-95284-5.

[7] J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2000.

[8] M. S. Chen, J. Han, and P. S. Yu. Data mining: An overview from a database perspective. IEEE Trans. Knowledge and Data Engineering, 8:866-883, 1996.

[9] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, 1996.

[10] W. J. Frawley, G. Piatetsky-Shapiro and C. J. Matheus, Knowledge Discovery in Databases: An Overview. In G. Piatetsky-Shapiro et al. (eds.), Knowledge Discovery in Databases. AAAI/MIT Press, 1991.

[11] T. Imielinski and H. Mannila. "A database perspective on knowledge discovery", Communications of ACM, 39:58-64, 1996.

[12] G. Piatetsky-Shapiro, U. M. Fayyad, and P. Smyth. From data mining to knowledge discovery: An overview. In U.M. Fayyad, et al. (eds.), Advances in Knowledge Discovery and Data Mining, 1-35. AAAI/MIT Press, 1996.

[13] G. Piatetsky-Shapiro and W. J. Frawley,Knowledge Discovery in Databases. AAAI/MIT Press, 1991.

[14] P. Lin Nancy, I. Chang Chung,Yi. Jan Nien, Jen. Chen Hung and Hua. Hao Wei, "A Deflected Grid-based Algorithm for Clustering Analysis", International Journalof Mathematical Models and Methods In Applied Sciences, Vol. 1,No. 1,2007.

[15] Guojun Gan, Chaqun Ma and Jianhong wu, " Data Clustering Theory, Algorithms and Applications" ASA-SIAM series

[16] Gholamhosein Sheikholeslami, Surojit Chatterjee, Aidong Zhang, "WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases".

[17] Bill Andreopoulos, Aijun An, Xiaogang Wang and Michael Schroeder, "A road map of clustering algorithms: finding a match for a biomedical application",Advance Access publication, Vol. 10, No. 3, 2009.

[18] Harley ER. "Comparison of Clique-Listing Algorithms",In Proceedings of the International Conference on Modeling, Simulation and Visualization Methods (MSV'04), Las Vegas, Nevada,USA, June 21-24, 2004, pages 433-438. CSREA Press.

[19] S. Guha, R. Rastogi, and K. Shim. CURE: "An efficient clustering algorithm for large databases", ACMSIGMOD International Conference on Management of Data, 1998.

[20] A.K. JAIN, M.N. MURTY, and P.J. FLYNN, "Data Clustering: A Review", ACM Computing Surveys, Vol. 31, 1999.

[21] R. Xu, and D. Wunsch, "Survey of clustering algorithms", IEEE Transactions on Neural Networks, Vol. 16, pp. 645-678, 2005.

[22] Zhang Y. , Mao J. and Xiong Z.: An efficient Clustering algorithm, In Proceedings of Second International Conference on Machine Learning and Cyber netics, November 2003.

[23] Zulkifli, A.H. and Meeran, S. Decomposition of interacting features using a Kohonon self-organizing feature map neural network. Engineering Applications of Artificial Intelligence, 1999, 12, 59-78.

[24] Sun Jigui, Liu Jie, Zhao Lianyu, "Clustering algorithms Research",Journal of Software ,Vol 19,No 1, pp.48-61,January 2008.

[25] A. Topchy, A. Jain, W. Punch, "A mixture model for clustering ensembles", In Proceedings of the SIAM International Conference on Data Mining, pp. 331–338, 2004.

[26] E. R. Hruschka, R. J. G. B Campello, L. N. de Castro, "Evolutionary Search for Optimal Fuzzy C-Means Clustering", In Proc. Int. Conference on Fuzzy Systems, pp. 685-690, 2004.

[27] A. A. Freitas, "A Review of Evolutionary Algorithms for Data Mining", In: Soft Computing for Knowledge Discovery and Data Mining, pp. 61-93, O. Maimon; L. Rokach (Editors), Springer, 2007.

[28] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. "Automatic subspace clustering of high dimensional data for data mining applications", In Proc. ACM SIGMOD, pages 94–105, 1998.

[29] K. Sequeira and M. J. Zaki. "Schism: A new approach forinteresting subspace mining", In Proc. IEEE ICDM, pages186–193, 2004.

[30] Nikhil R. Pal, Kuhu Pal, James M. Keller, and James C. Bezdek," A Possibilistic Fuzzy c-Means Clustering Algorithm", IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 13, NO. 4, AUGUST 2005

AUTHORS PROFILE

Ms. Angeline Christobel is working as an Asst. Professor in AMA International University, Bahrain. She is currently pursuing her research in Karpagam University, Coimbatore, India. Her research interest is in Data mining, Web mining and Neural Networks.

Mr.Suresh Subramanian, IS Analyst, Ahlia University Bahrain is currently pursuing his research in Karpagam University, Coimbatore, India. His research interest is in Web mining and Web mining.

Dr Minerva Bunagan is presently associated with AMA International University, Bahrain as the Dean of College of Computer Studies. Her research interests include Software Engineering and Web mining

Dr. Sivaprakasam is working as a Professor in Sri Vasavi College, Erode, India. His research interests include Data mining, Internet Technology, Web & Caching Technology, Communication Networks and Protocols, Content Distributing Networks.