Evaluating Optimal Clustering Techniques for Efficient Storage Retrieval Methods in Large Database Using Soft Computing Techniques

Dr. Ashutosh Gaur
Associate Professor
Bharati Vidyapeeth Institute of Management & Research,
New Delhi, India
E-ID: ashutosh_gaur@yahoo.com

Dr. Manu Pratap Singh
Associate Professor
Dept. of Computer Science
Dr. B. R. Ambedkar University
Agra, India

Abstract— **Data storage is one of the universal acts in computer transactions in the world. With the advancement of technology, now computers are the affordable device to almost all the human beings especially in the urban areas. At the same time the data is now becoming the problem. So efficient storage retrieval methods are quite important search for patterns in data is a human endeavor that is as old as it is ubiquitous, and has witnessed a dramatic transformation in strategy throughout the years. Whether we refer to hunters seeking to understand the animals' migration patterns, or farmers attempting to model harvest evolution, or turn to more current concerns, like sales trend analysis, assisted medical diagnosis, or building models of the surrounding world from scientific data, we reach the same conclusion: hidden within raw data we could find important new pieces of information and knowledge. This paper is an attempt to evaluate the Optimal Clustering Techniques for efficient storage retrieval methods in large databases using soft computing techniques.**

Index Terms— Clustering, Data Storage, KDD, K-Means.

I. Introduction

A formal definition of *data mining* (DM), also known – historically – as *data fishing, data dredging* (1960-), *knowledge discovery in databases* (1990-), or – depending on the domain, as *business intelligence, information discovery, information harvesting* or *data pattern processing* – is [Fay96]

Definition: *Knowledge Discovery in Databases (KDD) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.*

By *data* the definition refers to a set of facts (e.g. records in a database), whereas *pattern* represents an expression which describes a subset of the data, i.e. any structured representation or higher level description of a subset of the data. The term *process* designates a complex activity, comprised of several steps, while *non-trivial* implies that some search or inference is necessary, the straightforward derivation of the patterns is not possible. The resulting models or patterns should be *valid* on new data, with a certain level of confidence. Also, we wish that the patterns be *novel* – at least for the system and, ideally, for the analyst – and *potentially useful*, i.e. bring some kind of benefit to the analyst or the task. Ultimately, they need to be interpretable, even if this requires some kind of result transformation.

An important concept is that of *interestingness,* which normally quantifies the added value of a pattern, combining validity, novelty, utility and simplicity. This can be expressed either explicitly, or implicitly, through the ranking performed by the DM system on the returned patterns. A short note should be made on the fact that, even if initially DM represented a component in the KDD process, responsible with finding the patterns in data, currently the two terms are used interchangeably, both being employed to refer to the overall discovery process, which is comprised of several steps, as presented in the next section. This entire process, as originally envisioned by Fayyad, Piatetsky- Shapiro and Smyth (1996), is shown in Figure 1 the first three steps in Figure 1 involve preparing the data for mining. The relevant data must be selected from a potentially large and diverse set of data, any necessary preprocessing must then be performed, and finally the data must be transformed into a representation suitable for the data mining algorithm that is applied in the data mining step. As an example, the preprocessing step might involve computing the day of week from a date field, assuming that the domain experts thought that having the day of week information would be useful. An example of data transformation is provided by Cortes and Pregibon (1998). If each data record describes one *phone call* but the goal is to predict whether a *phone number* belongs to a business or residential customer based on its calling patterns, then all records associated with each phone number must be *aggregated* , which will entail creating attributes corresponding to the average number of calls per day, average call duration, etc.

While data preparation does not get much attention in the research community or the data mining community in general, it is critical to the success of any data mining project because without high quality data it is often impossible to learn much from the data. Furthermore, although most research on data mining pertains to the data mining algorithms, it is commonly acknowledged that the choice of a specific data mining algorithms is generally less important than doing a good job in data preparation. In practice it is common for the data preparations steps to take more time and effort than the actual data mining step. Thus, anyone undertaking a data mining project should ensure that sufficient time and effort is allocated to the data preparation steps. For those interested in this topic, there is a book (Pyle 1999) that focuses exclusively on data preparation for data mining.

The fourth step in the data mining process is the data mining step. This step involves applying specialized computer algorithms to identify patterns in the data.

A lot of data can be gathered from different fields but this data is useless without proper analysis to obtain useful information. In this paper, we focus on one of the important techniques in data mining: Clustering

An illustration example of clustering is shown in Fig 1.1. Data clustering is considered as an unsupervised learning technique in which objects are grouped in unknown predefined clusters. On the contrary, classification is a supervised learning in which objects are assigned to predefined classes (clusters).

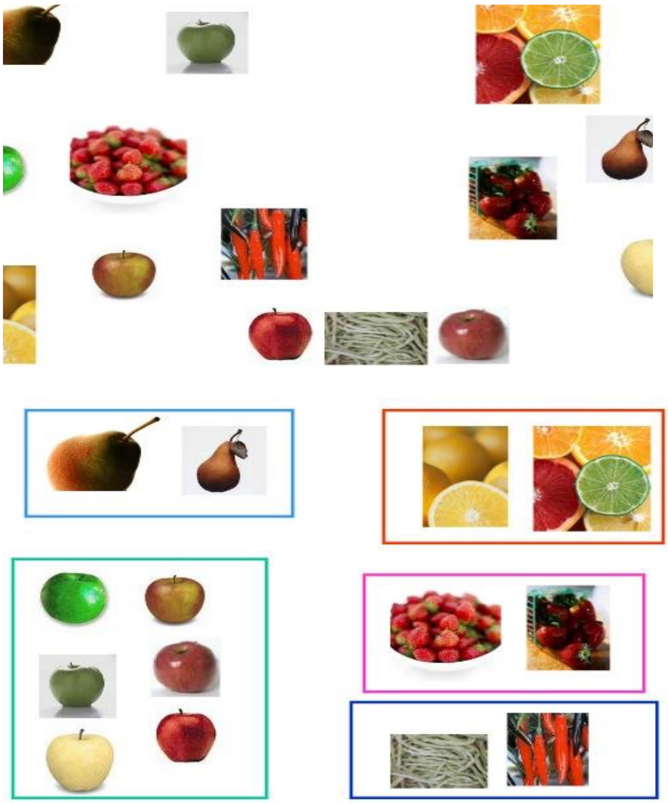

Figure 1.1: Clustering example (a) dataset contains 14 objects. (b) Objects are grouped into 5 clusters. (Adapted from: http://www.slideshare.net/pierluca.lanzi/machine-learning-and-data-mining-08-clustering-hierarchical)

II. Basic Concept Of Clustering

The problem of data clustering can be formulated as follows: given a dataset D that contains n objects x1,x2,…,xn (data points, records, instances, patterns, observations, items) and each data point is in a d-dimensional space, i.e. each data point has d dimensions (attributes, features, variables, components). This can be expressed in a matrix format as:

$$D=\left\{\begin{matrix} X11 & X12 & X13 & ---- & X1d \\ X21 & X22 & X23 & ---- & X2 \\ -- & & -- & & -- \\ Xn1 & Xn2 & Xn3 & ---- & Xnd \end{matrix}\right\}$$

Data clustering is based on the similarity or dissimilarity (distance) Measures between data points. Hence, these measures make the cluster analysis meaningful [28]. The high quality of clustering is to obtain high intra-cluster similarity and low inter-cluster similarity as shown in Fig. 1.2. In addition, when we use the dissimilarity (distance) concept, the latter sentence becomes: the high quality of clustering is to obtain low intra-cluster dissimilarity and high inter-cluster dissimilarity.

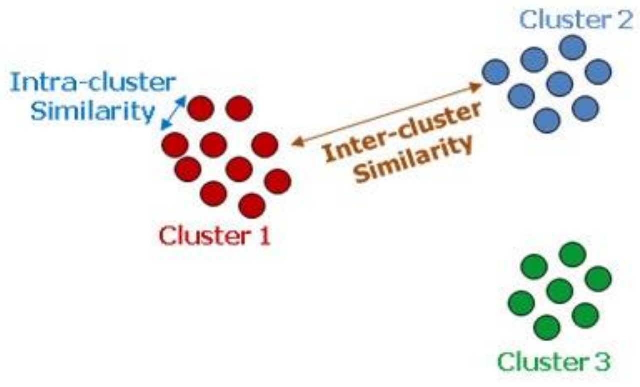

Figure 1.2: Inter-cluster and Intra-cluster similarities of clusters.

Importance of Clustering

Data clustering is one of the main tasks of data mining [1] and pattern Recognition [2]. Moreover, it can be used in many applications such as:

1. Data compression [3].
2. Image analysis [5].
3. Bioinformatics [6].

4. Academics [9].
5. Search engines [79].
6. Wireless sensor networks [80].
7. Intrusion detection [81].
8. Business planning [82].

Motivations

The K means algorithm is considered as one of the top ten algorithms in data Mining [35]. A lot of researches and studies have been proposed due to its Simplicity and efficiency [55]. These efforts have focused on finding possible Solutions to one or more of the limitations that have been identified in page 10. Kmeans with random initialization conditions need to be rerun many times each with different conditions to find more suitable results [21]. Many Algorithms have been considered to provide better seeds so the Kmeans Algorithm is likely to converge to the global optimum like Minmax[43], Kmeans++ [44] and [45]. Other solutions to the initial prototypes sensitivity
can be found in [46] where they defined new criterion functions for Kmeans and they proposed three algorithms: weighted Kmeans, inverse weighted Kmeans [52] and inverse exponential Kmeans [53]. Other improvements of Kmeans focus on its efficiency where the complexity of Kmeans involves the data set size, number of dimensions, number of clusters and the number of iteration to be converged. There are many works to reduce the computational load and make it faster such as [4], [47-49]. Asgharbeygi and Maleki [39] proposed a new distance metric which is the geodesic distance to ensure resistance to outliers. Several works have been introduced to extend the use of means for numerical variables, thus Kmeans can deal with categorical variables such as [50], [51]. JJ Sheu et. al. [61] proposed a new algorithm and they named it Intelligent Kmeans (IKM) for deciding the proper number of clusters, choosing a better initial prototypes and reducing the effect of outliers on the clustering result. IKM divided the range of data points for each d dimensions into M regions where M is a constant input number. One drawbacks of this method, is the
choice of grid size. If it is small, it will produce a large number of clusters and Vice versa.

Many researchers have been involved in developing solutions to the Kmeans and other clustering algorithms such as using neighborhood model [23], ant colony [24], the principle of gravity [26], genetic algorithms [25], and clustering method with constraints [27]. The problem in clustering is that we do not have prior information knowledge about the given dataset. Moreover, the choice of input parameters such as the number of clusters, number of nearest neighbors and other factors in these algorithms make the clustering more challengeable topic. Thus any incorrect choice of these parameters yields bad clustering results. Furthermore, these algorithms suffer from unsatisfactory accuracy when the dataset contains clusters with different complex shapes, densities, sizes, noise and outliers.

Uses of Clustering

There are many reasons to cluster data. The main reason is that it allows us to build simpler, more understandable models of the world, which can be acted upon more easily. People naturally cluster objects for this reason all the time. For example, we are able to identify objects as a “chair” even if they look quite different and this allows us to ignore the specific characteristics of a chair if they are irrelevant. Clustering algorithms automate this process and allow us to exploit the power of computer technology. A secondary use for clustering is for dimensionality reduction or data compression. For example, one could identify ten attributes for a data set, cluster the examples using these attributes, and then replace the ten attributes with one new attribute that specifies the cluster number. Reducing the number of dimensions (i.e., attributes) can simplify the data mining process. Clustering can also aid with data compression by replacing complex objects with an index into a table of the object closest to the center of that objects cluster

There are many specific applications of clustering and we list only a few here. Clustering can be used to automatically segment customers into meaningful groups (e.g., students, retirees, etc.), so that more effective, customized, marketing plans can be developed for each group. In document retrieval tasks the returned documents may be clustered and presented to the users grouped by these clusters (Zamir and Etzioni 1998) in order to present the documents to the user in a more organized and meaningful way. For example, clustering can be employed by a search engine so that the documents retrieved from the search term “jaguar” cluster the documents related to the jaguar animal separately from those related to the Jaguar automobile (the ask.com search engine currently provides this capability). The clustering algorithm can work effectively in this case because one set of returned documents will repeatedly have the term “car”, “automobile” or “S-type” in it while the other set may have the terms “jungle” or “animal” appear repeatedly.

Categories of Clustering Algorithms

Clustering algorithms can be organized by the basic approach that they employ. These approaches are also related to the type of clustering that the algorithm produces. The two main types of clusterings are hierarchical and non-hierarchical. A hierarchical clustering has multiple levels while a non-hierarchical clustering has only a single level. An example of a hierarchical clustering is the taxonomy used by biologists to classify living organisms (although that hierarchy was not formed using data mining algorithms).

The non-hierarchical clustering algorithms will take the presented objects and place each into one of *k* clusters, where each cluster must have at least one object. Most of these algorithms require the user to specify the value of *k*, which is often a liability, since the user will generally not know ahead of time the optimal number of meaningful clusters. The framework used by many of these algorithms is to form an initial random clustering of the objects and then repeatedly move objects between clusters to improve the overall quality of the clusters. One of the oldest and most notable of these methods is the K-means clustering algorithm (Jain and Dubes 1988). This algorithm randomly assigns each object to one of the *k* clusters and then computes the mean (i.e., center or centroids) of the points in the cluster. Then each object is reassigned to the cluster based on which centroid it is closest to and then the centroids of each cluster are recomputed. This cycle continues until no changes are made. This very simple method sometimes works well. Another way to generate non-hierarchical clustering is via density-based clustering methods, such as DBSCAN (Ester et al. 1996), which find regions of high density that are separated from regions of low density. One advantage of DBSCAN is that because it is sensitive to the density differences it can form clusters with arbitrary shapes.

Hierarchical clustering algorithms are the next type of clustering algorithms. These algorithms can be divided into agglomerative and divisive algorithms. The agglomerative algorithms start with each object as an individual cluster and then at each iteration merge the most similar pair of clusters. The divisive algorithms take the opposite approach and start with all objects in a single partition and then iteratively split one cluster into two. The agglomerative techniques are by far the more popular method. These methods are appropriate when the user prefers a hierarchical clustering of the objects.

Soft computing:-

The term soft computing was also introduced by Prof Zadeh in 1992[2].It is a collection of some biological inspired methodologies such as Fuzzy logic(FL),Neural Network(NN),Genetic Algorithm(GA) and their different combined forms namely GA-FL,GA-NN,NN-FL,GA-FL-NN, in which precision is traded for tractability, robustness, ease of implementation and a low cost solution

Features of soft computing:

1. Soft computing is an emerging field which has the following features:-
2. It does not require an extensive mathematical formulation of the problem
3. It may not be able to yield so much precise solution as that obtained by the hard technique
4. Different members of this family are able to perform various type of tasks. For example fuzzy logic(FL) is a power fool tool for dealing with imprecision and uncertainity,Neural network is a potential tool for learning and adaptation and genetic algorithm(GA) is an important tool for search and optimization .Each of these tools has its inherent merits and demerits .In combined techniques(such as GA-FL,GA-NN,NN-FL,GA-FL-NN),either two or three constituent tools are coupled to get the advantage from both of them and remove their inherent limitations .thus in soft computing, the functions of the constituent members are complementary in nature and there is no competition among themselves.
5. Algorithm developed based on soft computing is generally found to be adaptive in nature. Thus it can accommodate to the changes of dynamic environment.

III. RELATED LITERATURE REVIEWS

The clustering problems can be categorized into two main types: fuzzy clustering and hard clustering. In fuzzy clustering, data points can belong to more than one cluster with probabilities between 0 and 1 [10], [11] which indicate the strength of the relationships between the data points and a particular cluster. One of the most popular fuzzy clustering algorithms is fuzzy c-mean algorithm [12], [13], [14]. In hard clustering, data points are divided into distinct clusters, where each data point can belong to one and only one cluster. The hard clustering is divided into hierarchical and partitional algorithms.

Hierarchical algorithms create nested relationships of clusters which can be represented as a tree structure called dendrogram [28]. Hierarchical algorithms can be divided into agglomerative and divisive hierarchical algorithms. The agglomerative hierarchical clustering starts with each data point in a single cluster. Then it repeats merging the similar pairs of clusters until all of the data points are in one cluster, such as complete linkage clustering [29] and single linkage clustering [30]. CURE [15], ROCK [16], BIRCH [17] and Chameleon [18] are examples of this hierarchical algorithm. The divisive hierarchical algorithm reverses the operations of agglomerative clustering, it starts with all data points in one cluster and it repeats splitting large clusters into smaller ones until each data point belongs to a single cluster such as DIANA clustering algorithm [31].In the contrary, Partitional clustering algorithm divides the dataset into a set of disjoint clusters such as Kmeans [32], [42] PAM [31] and CLARA [31].Moreover, the partitional algorithms have been considered more appropriate for applications with large dataset, in which the construction of the dendrogram is computationally expensive [1], [37]. One of the problems in applying partitional methods is the choice of the number of clusters within the given datasets where the determination of the number of clusters is one of the most problematic issues in data clustering [7]. The partitional algorithms often use a certain objective function and produce the desired clusters by optimizing this objective function [36].

The clustering algorithms that are based on estimating the densities of data points are known as density-based methods. One of the basic density based clustering algorithm is DBSCAN [40]. It defines the density by counting the number of data points in a region specified by a predefined radius known asepsilon _ around the data point. If a data point has a number greater than or equal to predefined minimum points known as *MinPts*, then this point is treated as a core point. Non-core data points that do not have a core data point within the predefined radius are treated as noise. Then the clusters are formed around the core data points and are defined as a set of density-connected data points that is maximal with respect to density reach ability. DBSCAN may behave poorly due its weak definition of data points' densities and its globally predefined parameters of ε and *MinPts*. There are many works that try to improve the well known DBSCAN such as [41], [56-60].

Similarity Graphs

Another type of clustering algorithms is based on the construction of similarity graphs in which a given set of data points is transformed into vertices and edges. The constructed graph can be used to obtain a single highly connected graph that is then partitioned by edge cutting to obtain sub graphs [72], [74],[68]. Basically, the kinds of graphs are neighborhood, k-nearest neighbor and fully connected graph [2], [70], [54].The neighborhood graph connects all data points whose pair wise distances are smaller than a predefined threshold .In the k-nearest neighbor graph the data point *vi* (vertex) is connected with another data point in the dataset if it is in the k-nearest neighbors of *vi* where k is a predefined parameter. This method lets the k-nearest neighbor produces a directed graph. The undirected graph can be obtained from the k-nearest neighbor by simply ignoring the directions of edges or by having a mutual k nearest neighbor graph in which two vertices are connected by an edge if and only if these two vertices are among the k-nearest neighbors of each other. The fully connected graph connects all data points that have a positive similarity measurement with each other. The similarity measure can be produced by using the Gaussian similarity function *Sij=exp(*-d___*/2_2)* where *dij* is the Euclidean distance between two data points ***xi*** and ***xj*** and the parameter is also a user defined one that controls the width of neighborhoods.

K means Algorithm

One of the most well-known unsupervised learning algorithms for clustering datasets is K means algorithm [31], [37]. The K means clustering is the most widely used due to its simplicity and efficiency in various fields [33], [38]. It is also considered as the top ten algorithms in data mining [35]. The K means algorithm works as follows:

1. Select a set of initial *k* prototypes or means throughout a dataset, where k is a user-defined parameter that represents the number of clusters in the dataset.
2. Assign each data point in a dataset to its nearest prototype ***m***.
3. Update each prototype according to the average of data points assigned to it.
4. Repeat step 2 and 3 until convergence.

The K means algorithm depends on minimizing the sum of squared error function which is very simple and can be easily implemented. Where dataset *D* contains *n* data points ***x1*,*x2***,*…*,***xn*** such that each data point is *d* dimensional vector in *Rd*, and ***mi*** is the prototype of cluster *Ci*, and *k* is the given number of clusters. However, it has several drawbacks: the number of clusters k in a given dataset should be known in advance, the result strongly depends on the initial prototypes, the sensitivity to noise and outliers, the problem of dead prototypes or empty clusters and the converge to local optima [34]. The K means works for globular shaped, similar size and density clusters.

CURE and Chameleon Algorithms

CURE [15] uses a constant number of well scattered representative data points from all data points in the dataset to represent a cluster instead of selecting one single centroid to represent a cluster in Kmeans. These are shrunk towards the centroid of the cluster according to a user predefined shrinking factor. Then a consecutive merging of the closest pair of the cluster's representative points are occurred until the predefined number of clusters is obtained. The selection of the shrinking factor and the merging process make CURE ineffective with complex datasets and they can cause false outliers [22].

Chameleon [18] uses a graph construction based on k-nearest neighbors, and then it splits the graph into a set of small clusters using hMetis algorithm [19]. After that it merges these small clusters based on their similarity measure. It has been used to find non-convex shaped clusters, however, it cannot handle noise and outliers and needs to set parameters correctly in order to obtain good results [22], [20].

Affinity Propagation Algorithm

Another type of clustering algorithms is called Affinity Propagation [67] that passes messages between data points to identify a set of exemplars (cluster centers) and their corresponding clusters. In contrary of selecting an initial set of cluster centers randomly and iteratively refines them such that the sum of squared error is minimized as in K means; the Affinity Propagation provides a different approach that simultaneously considers all data points as candidate exemplars. Then two types of messages are exchanged between data points. The Responsibility messages are sent from data points to candidate exemplars and indicate how strongly each data point is biased to the candidate exemplar over other candidate exemplars. The Availability messages are sent from candidate exemplars to data points and reflect evidence that each candidate exemplar is available to be a cluster center of the data points. The Affinity Propagation uses the median of similarities between data points as preferences rather than the predetermined number of clusters.

Spectral Clustering Algorithm

Recently, the spectral clustering [70] has become one of the most popular clustering algorithms which outperform the traditional algorithms such as K means. Furthermore, they are designed to handle non-convex shaped clusters. However, spectral clustering suffers from heavily computations. The similarity measure and graph cutting are also used in spectral clustering algorithms. The core of the spectral clustering algorithms is to use the properties of

Eigen vectors of Laplacian matrix for performing graph partitioning [69-76].The Laplacian matrix is constructed by building an affinity graph matrix with a similarity measure. The common similarity measure is to use the Gaussian function *Sij* as stated previously for its simplicity. Hence, the Laplacian matrix L is calculated as *L=D-S* where *D* is the diagonal matrix whose elements are the sum of all row elements of *S*. Then, the spectral Clustering computes a column matrix of the first k eigenvectors of *L* where k is a predefined number of clusters. Thus it finds the clusters of mapped data Points that corresponding to the column matrix of eigenvectors by performing Kmeans algorithm

Spectral Clustering using Nystrom Method

W.-Y. Chen et. al. [76] proposed sparsification and Nystrom approaches to address the computational difficulties and to improve the results. We compare our algorithm with spectral clustering using Nystrom method because it needs less computation and does not need the prespecified number of nearest neighbors as in sparsification method. Nystrom method is a technique for finding an approximate eigen decomposition. The spectral clustering using Nystrom method uses randomly sample data points from the dataset to approximate the similarity matrix of all data points in the dataset. Then it finds the first k eigenvectors of the normalized Laplacian matrix of the Nystrom method and performs K means to cluster dataset.

Topology Preserving Mapping

A topographic mapping is a transformation of high dimensional data. Furthermore, it preserves some structure in the data such as the points which are mapped close to each other share some common properties while in contrast the points which are mapped far from each other do not share a common feature or property.

The Self-organizing map (SOM) [84] and the Generative topographic mapping (GTM) [85] have been considered as very popular topology preserving mapping techniques for data visualization and dimensionality reduction. The GTM can be considered as a statistical alternative to the SOM Overcoming many of its limitations such as the absence of a cost function and the lack of proof convergence [86].

Self-organizing Map (SOM)

The Self-organizing Map (SOM) [84] is a type of artificial neural network that is trained using unsupervised learning. SOM reduces dimensions of the given datasets by producing a map of usually one or two dimensions. Furthermore, SOM uses a neighborhood function to preserve the topological properties of the input space. The SOM consists of components called nodes or neurons in which they are Usually arranged in a hexagonal or rectangular grid. It first initializes the weights associated with each neuron by assigning them small random values. Then the SOM proceeds to three essential processes: competition, cooperation, and adaptation [28].

IV. Objectives Of The Study

The objectives of my study are as follows:

- To review existing clustering techniques for retrieval methods used in large database
- To compare and analyze the performance of existing retrieval methods used in clustering such as K means, Hierarchical, Partitioning, Clara algorithm etc.
- To develop efficient retrieval method for large database using soft computing techniques like Genetic Algorithm, Self organizing Map, Back Propagation algorithm etc and hybrid evolutionary algorithms using Neural Network
- To evaluate the performance of existing retrieval techniques and the developed soft computing techniques of data retrieval

V. Research Methodology

- Construction implementation of real world problem data base
- Apply data cleaning methods
- Implementation of clusters (Static & Dynamic) using K-means algorithm and analyze the performance of storage retrieval (on the basis of time, accuracy, space, response time etc.)
- It is proposed to implement clusters using SOM with dynamic competitive learning Artificial Neural Network (ANN) and analysis the performance of storage retrieval (on the basis of time, accuracy, space, response time etc.)

- It is proposed to analysis the performance of storage retrieval on a database using various pattern classification methods.
- It is proposed to analysis the performance of storage retrieval on a database using various Hybrid Evolutionary Algorithm
- It is proposed to analyze the combined performances of storage retrieval methods.

VI. Conclusion

Data clustering is a method of grouping similar objects together. Thus the similar objects are clustered in the same group and dissimilar objects are clustered in different ones.

Traditional approaches for deriving knowledge from data rely strongly on manual analysis and interpretation. For any domain – scientific, marketing, finance, health, business, etc. – the success of a traditional analysis depends on the capabilities of one/more specialists to read into the data: scientists go through remote images of planets and asteroids to mark interest objects, such as impact craters; bank analysts go through credit applications to determine which are prone to end in defaults. Such an approach is slow, expensive and with limited results, relying strongly on experience, state of mind and specialist know-how.

Moreover, the volume of generated data is increasing dramatically, which makes traditional approaches impractical in most domains. Within the large volumes of data lay hidden strategic pieces of information for fields such as science, health or business. Besides the possibility to collect and store large volumes of data, the information era has also provided us with an increased computational power. The natural attitude is to employ this power to automate the process of discovering interesting models and patterns in the raw data. Thus, the purpose of the knowledge discovery methods is to provide solutions to one of the problems triggered by the information era: "data overload".

References

[1] A. Jain, M. Murty, and P. Flynn, "Data Clustering: A review," ACM Computing Surveys (CSUR), Vol. 31, issue (3), pp. 264-323, Sept. 1999.

[2] R. Duda, P. Hart, and D. Stork, "Pattern Classification," John Wiley & Sons, second edition, 2001.

[3] A. Gersho and R. Gray, "Vector Quantization and Signal Compression," Kulwer Academic, Boston, 1992.

[4] M. Al- Zoubi, A. Hudaib, A. Huneiti and B. Hammo, "New Efficient Strategy to Accelerate k-Means Clustering Algorithm," American Journal of Applied Science, Vol. 5, No. 9, pp. 1247-1250, 2008.

[5] M. Celebi, "Effective Initialization of K-means for Color Quantization," Proceeding of the IEEE International Conference on Image Processing, pp. 1649-1652, 2009.

[6] M. Borodovsky and J. McIninch, "Recognition of genes in DNA sequence with ambiguities, " Biosystems, Vol. 30, issues 1-3, pp. 161-171, 1993

[7] X. Wang, W. Qiu and R. H. Zamar "CLUES: A Non-parametric Clustering Method Based on Local Shrinking," Journal Computational Statistics & Data Analysis, Vol.52, Issue 1, pp. 286-298, 2007.

[8] M. Khalilian, N. Mustapha, N. Suliman, and A. Mamat, "A Novel Kmeans Based Clustering Algorithm for High Dimensional Datasets," Proceedings of the International Multi Conference on Engineers and Computer Scientists (IMECS 2010),Vol. I, Hong Kong, March 2010.

[9] O. Oyelade, O. Oladipupo and I. Obagbuwa, "Application of Kmeans Clustering Algorithm for Prediction of Students' Academic Performance," International Journal of Computer Science and Information Security, vol. 7, no. 1, pp. 292-295, 2010.

[10] J. Bezdek and N. Pal, "Fuzzy Models for Pattern Recognition," IEEE press, New York,NY, USA, 1992.

[11] D. Karaboga and C. Ozturk, "Fuzzy Clustering with Artificial Bee Colony Algorithm," Scientific Research and Essays, Vol. 5(14), pp. 1899-1902, 2010.

[12] J. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms," Plenum Press, New York, NY, USA, 1981.

[13] J. Bezdek, J. Keller, R. Krishnapuram, and N. Pal, "Fuzzy Models and Algorithms for Pattern and Image Processing," Kluwer, Dordrecht, Netherland, 1999.

[14] F. Hoppner, F. Klawonn, R. Kruse, and T. Runkler, "Fuzzy Cluster Analysis," J.Wiley & Sons, Chichester, England, 1999.

[15] S. Guha, R. Rastogi, and K. Shim, "CURE: An Efficient Clustering Algorithm for Large Databases," Proceedings of ACM International Conference on Management of Data, pp. 73-84, 1998.

[16] S. Guha, R. Rastogi, and K. Shim, "ROCK: A Robust Clustering Algorithm for Categorical Attributes," Information Systems, Vol. 25, No. 5, pp.345-366, 2000.

[17] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An Efficient Clustering Method for Very Large Databases," Proceedings of ACM SIGMOD Workshop Research Issues on Data Mining and Knowledge Discovery, pp. 103-114, 1996.

[18] G. Karypis, E.-H. Han, and V. Kumar, "Chameleon: Hierarchical Clustering Using Dynamic Modeling," IEEE Computer, Vol. 32, No. 8, pp. 68-75, 1999.

[19] G. Karypis and V. Kumar, "Multilevel Algorithms for Multi-Constraint Graph Partitioning," Technical Report, University of Minnesota, May 1998.

[20] CAO Chang-hu, and LI Ya-fei, "Improved Chameleon Algorithm for Cluster Analysis," Journal of Science Technology and Engineering, issues 33, 2010.

[21] P. Bhattacharya and M. L. Gavrilova, "CRYSTAL: A New Density Based Fast and Efficient Clustering Algorithm," Proceeding of the 3rd International Symposium on Voronoi Diagrams in Science and Engineering, pp. 102-111, 2006.

[22] A. Foss and O. Zaiane, "A Parameterless Method for Efficiently Discovering Clusters of Arbitrary Shape in Large Datasets," In IEEE International Conference on Data Mining, pp. 179-186, 2002.

[23] F. Cao, J. Liang, and G. Jiang, "An Initialization Method for Kmeans Algorithm Using Neighborhood Model," Journal Computers & Mathematics with Applications, Vol. 58, Issue 3, pp. 474-483, Aug. 2009.

[24] X. Liu and H. Fu, "An Effective Clustering Algorithm with Ant Colony," Journal of Computers, Vol. 5, No. 4, pp. 598-605, 2010.

[25] H.-J. Lin, F.-W. Yang, and Y.-T. Kao, "An Efficient GA-based Clustering Technique,"Tamkang Journal of Science and Engineering, Vol. 8, No 2, pp. 113-122, 2005.

[26] T. Zhang, and H. Qu, "An Improved Clustering Algorithm," Proceedings of the Third International Symposium on Computer Science and Computational Technology (ISCSCT '10) Jiaozuo, China, pp. 112-115, Aug. 2010.

[27] V.-V. Vu, N. Labroche, and B. Bouchon-Meunier, "An Efficient Active Constraint Selection Algorithm for Clustering," 20th International Conference on Pattern Recognition, pp.2969-2972, 2010

[28] G. Gan, Ch. Ma, and J. Wu, "Data Clustering: Theory, Algorithms, and Applications," ASA-SIAM series on Statistics and Applied Probability, SIAM, 2007.

[29] D. Defays, "An Efficient Algorithm for A Complete Link Method," The Computer Journal, Vol. 20, pp. 364-366, 1977.

[30] R. Sibson, "SLINK: an Optimally Efficient Algorithm for the Single Link Cluster Method," The Computer Journal, Vol. 16, No. 1, pp. 30-34, 1973.

[31] L. Kaufman, and P. Rousseeuw, "Finding Groups in Data: An Introduction to Cluster Analysis," John Wiley & Sons, 1990.

[32] J. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations," 5th Berkeley Symp. Math. Statist. Prob., Vol. 1, pp. 281-297, 1967.

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

[34] R. Xu and D. Wunsch II, "Computational Intelligence in Clustering Algorithms, With Applications," In A. Iske and J. Levesley, Eds., Algorithms for Approximation, Proceedings of 5th International Conference, Chester, England, UK, Springer, Heidelberg, pp. 31-50, 2007.

[35] XindongWu and et. Al., "Top 10 Algorithms in Data Mining," Journal of Knowledge and Information Systems, Vol. 14, Issues 1-37, 2008.

[36] P. Hansen and B. Jaumard, "Cluster analysis and mathematical programming," Mathematical Programming, pp. 191-215, 1997.

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

[38] B.Bahmani Firouzi, T. Niknam, and M. Nayeripour, "A New Evolutionary Algorithm for Cluster Analysis," Proceeding of world Academy of Science, Engineering and Technology, Vol. 36, Dec. 2008.

[39] N. Asgharbeygi, and A. Maleki, "Geodesic Kmeans Clustering," 19th International Conference on Pattern Recognition, 2008.

[40] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, " A Density-based Algorithm for Discovering Clusters in Large Spatial Data sets with Noise," 2nd International Conference on Knowledge Discovery and Data Mining, pp. 226-231, 1996.

[41] M. Ankerst, M. Breunig, H. P. Kriegel, and J. Sander, "OPTICS: Ordering Objects to Identify the Clustering Structure," Proceedings of ACM SIGMOD in International Conference on Management of Data, pp. 49–60, 1999.

[42] E. Forgy, "Cluster Analysis of Multivariate Data: Efficiency vs. Interpretability of Classification," Biometrics, Vol. 21, 1965.

[43] D. Hochbaum, and D. Shmoys, "A Best Possible Heuristic for the K-Center Problem," Math. Oper. Res., Vol. 10, No. 2, pp. 180-184, 1985.

[44] D. Arthur, and S. Vassilvitskii, "Kmeans++: The Advantages of Careful Seeding,"Proceeding of SODA'07, pp. 1027-1035, 2007.

[45] M. Al-Daoud, "A New Algorithm for Clustering Initialization," Proceeding World Academy of Science, Engineering, and Technology, Vol. 4, 2005.

[46] W. Barbakh, and C. Fyfe, "Local vs. Global Interactions in Clustering Algorithms:Advances over Kmeans," International Journal of knowledge-based and Intelligent Engineering Systems, Vol. 12, 2008.

[47] Jim Z.C. Lai, and T. J. Huang, "Fast Global Kmeans Clustering Using Cluster Membership and Inequality," Pattern Recognition, Vol. 43, pp. 1954-1963, 2010.

[48] G. Frahling, and Ch. Sohler, "A Fast Kmeans Implementation Using Coresets," International Journal of Computational Geometry and Applications, Vol. 18, Issue 6, pp. 605-625, 2008.

[49] L. Taoying, and Y. Chen, "An Improved Kmeans for Clustering Using Entropy Weighting measures," 7th World Congress on Intelligent Control and Automation, 2008.

[50] S. Gupara, K. Rao, and V. Bhatnagar, "Kmeans Clustering Algorithm for Categorical Attributes," Proceeding 1st International Conf. on Data Warehousing and Knowledge Discovery, pp. 203-208, Italy, 1999.

[51] Z. Huang, "Extensions to The Kmeans Algorithms for Clustering Large Data Sets with Categorical Values," Data Mining & Knowledge Discovery, Vol. 2, pp. 283-304,1998.

[52] W. Barbakh and C. Fyfe. "Inverse Weighted Clustering Algorithm," Computing and Information Systems, 11(2), pp. 10-18, May 2007. ISSN 1352-9404.

[53] W. Barbakh. "The Family of Inverse Exponential Kmeans Algorithms," Computing and Information Systems, 11(1), pp. 1-10, February 2007. ISSN 1352-9404.

[54] W. Barbakh, and C. Fyfe, "Clustering and Visualization with Alternative SimilarityFunctions." The 7th WSEAS international conference on artificial intelligence,knowledge, engineering and data bases, AIKED'08, pp. 238–244. University of Cambridge, UK. 2008.

[55] A. K. Jain, "Data clustering: 50 years beyond K-means." Pattern Recognition Letters,Vol. 31, No. 8, pp. 651-666, 2010.

[56] B. Borah, and D.K. Bhattacharyya, "DDSC: A Density Differentiated Spatial

[57] Clustering Technique." Journal of Computers, Vol. 3, No. 2, pp. 72-79, 2008.

[58] A. Ram, S. Jalal, A. S. Jalal, and M. Kumar, "A Density Based Algorithm for Discovering Density Varied Clusters in Large Spatial Databases." International Journal of Computer Applications, Vol. 3, No. 6, June 2010.

[59] K. Mumtaz and K. Duraiswamy "A Novel Density Based Improved Kmeans Clustering algorithm – Dbkmeans," International Journal on Computer Science and Engineering (IJCSE), Vol. 2, No. 2, pp. 213-218, 2010.

[60] A. Fahim, A. Salem, F. Torkey, and M. Ramadan, "Density Clustering Based on Radius of Data (DCBRD)," International Journal of Mathematical and Computer Sciences, Vol. 3 No. 2, pp. 80-86, 2007.

[61] S. Kisilevich, F. Mansmann, and D. Keim, "P-DBSCAN: A Density Based Clustering Algorithm for Exploration and Analysis of Attractive Areas of Geo-tagged Photos,"Proceedings of the 1st International Conference and Exhibition on Computing forGeospatial Research & Application, Washington, D.C., 2010

[62] J.J Sheu, W.M Chen, W.B Tsai and K.T Chu, "An Intelligent Initialization Method for the Kmeans Clustering Algorithm." International Journal of Innovative Computing,Information and Control, Vol. 6, No. 6, pp. 2551-2566, June 2010. Convex Hull, "http://en.wikipedia.org/wiki/Convex_hull," last visit: June, 2011.Delaunay triangulation, "http://en.wikipedia.org/wiki/Delaunay_triangulation," last visit: June, 2011.

[63] G. Nalbantov, P. Groenen, J. Bioch, "Nearest Convex Hull Classification," Erasmus University Rotterdam, Econometric Institute in its series Econometric Institute report Dec 2006(http://repub.eur.nl/publications/index/728919268/NCH10.pdf)

[64] X. Zhou, and Y. Shi, "Nearest Neighbor Convex Hull Classification Method for Face Recognition," Proceedings of the 9th International Conference on Computational Science, pp. 570-577, 2009.[66] J. Hershberger, N. Shrivastava, and S. Suri, "Summarizing Spatial Data Streams Using ClusterHulls," Journal of Experimental Algorithmics (JEA), Vol. 13, Feb. 2009.

[65] B. J. Frey, and D. Dueck, "Clustering by Passing Messages Between Data Points,"Science, Vol. 315, pp. 972-949, 2007.

[66] E. Hartuv and R. Shamir, "A Clustering Algorithm Based on Graph Connectivity,"Information Processing Letters, Vol. 76, Nos. 4-6, pp. 175-181, 2000.

[67] M. Fielder, "A Property of Eigenvectors of Nonnegative Symmetric Matrices and Its Application to Graph Theory," Czechoslovak Mathematical Journal, Vol. 25, No. 100,pp.619-633, 1975.

[68] U. Luxburg, "A Tutorial on Spectral Clustering," Statistics and Computing, Vol. 17,No. 4, pp. 395-416, 2007.

[69] X. Zhang, J. Li, and H. Yu, "Local Density Adaptive Similarity Measurement for Spectral Clustering," Pattern Recognition Letters, 32, pp. 352-358, 2011.

[70] J. Santos, J. Marques de Sa, and L. Alexandre, "LEGClust: A Clustering Algorithm Based on Layered Entropic Subgraphs," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, No. 1, pp. 62-75, 2008.

[71] A. Ny, M. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an Algorithm," Advances in Neural Information Processing Systems, Vol. 14, 2001.

[72] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, pp. 888-905, 2000.

[73] D. Verma and Meila, "A Comparison of Spectral Clustering Algorithms," Technical Report UW-CSE-03-05-01, Washington University, 2003.

[74] W.-Y. Chen, Y. Song, H. Bai, C.-J. Lin, E. Chang, "Parallel Spectral Clustering in Distributed Systems," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 3, pp. 568–586, 2011.

[75] M. Wu and B. Scholkopf, "A Local Learning Approach for Clustering," Proceedingsof NIPS, pp. 1529–1536, 2007.

[76] C. Papadimitriou and K. Steiglitz, "Combinatorial Optimization: Algorithms and Complexity," Dover, New York, 1998.

[77] T. Liu, C. Rosenberg and H. Rowley, "Clustering Billions of Images with Large Scale Nearest Neighbor Search," Proceedings of the Eighth IEEE Workshop on Applications of Computer Vision (WACV), 2007.

[78] K. Akkaya, F. Senel and B. McLaughlan, "Clustering of Wireless Sensor and Actor Networks Based on Sensor Distribution and Connectivity," Journal of Parallel and Distributed Computing, Vol. 69, No. 6, pp. 573-587, 2009.

[79] L. Portnoy, E. Eskin and S. Stolfo, "Intrusion Detection with Unlabeled Data using Clustering," In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA), pp. 5-8, 2001.

[80] L. van der Maaten, E. Postma, and H. van den Herik, "Dimensionality reduction: A Comparative Review," Technical Report, MICC, Maastricht University, The Netherlands, 2007.

[81] T. Khonen, "Self-Organizing Maps," Springer, 1995.

[82] C. Bishop, M. Svensen, and C. Williams, "GTM: The Generative Topographic Mapping," Neural Computation, Vol. 10, No. 1, pp. 215–234, 1998.

[83] C. Bishop, M. Svensen, and C. Williams, "Developments of the Generative Topographic Mapping," Neurocomputing, vol. 21, No. 1, pp. 203-224, 1998.

[84] J.Y. Choi, J. Qiu, M.E. Pierce, and G. Fox, "Generative Topographic Mapping by Deterministic Annealing," In Proceedings of International Conference on Membrane Computing, pp.47-56, 2010.

[85] UCI Machine Learning Repository, Available from (http://archive.ics.uci.edu/ml).Last visit: Sep., 2011.

[86] C. Zhang and S. Xia, "K-means Clustering Algorithm with Improved Initial Center,"Second International Workshop on Knowledge Discovery and Data Mining, pp. 790-792, 2009.

[87] D. Pham, S. Dimov, and C. Nguyen, "Selection of K in K-means Clustering,"Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, Vol. 219, No. 1, pp. 103-119, 2005.

[88] G. Yu, H. Peng, J. Wei, and Q. Ma, "Enhanced Locality Preserving Projections using Robust Path Based Similarity," Neurocomputing, Vol. 74, No. 4, pp. 598-605, 2011.

[89] R. Harrison, and K. Pasupa, "Sparse Multinomial Kernel Discriminant Analysis

[90] L. Zhang, L. Qiao, and S. Chen, "Graph-optimized Locality Preserving Projections,"Pattern Recognition, Vol. 43, No. 6, pp. 1993-2002, 2010.

[91] H. Liu, J. Sun, L. Liu, and H. Zhang, "Feature Selection with Dynamic Mutual Information," Pattern Recognition, Vol. 42, No. 7, pp. 1330-1339, 2009.