International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991
Vol. 2, No. 2
May 2013

www.ijerst.com

Email: editorijerst@gmail.com or editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 2, No. 2, May 2013

Research Paper

INITIALIZATION K-MEANS USING ANT COLONY OPTIMIZATION

S Gnanapriya[1] and P Shiva Ranjani[1]

Corresponding Author:* ***P Shiva Ranjani,* ✉*shiva230482suresh@gmail.com*

Clustering is a machine learning technique that places data elements into related groups. Clustering can be defined as a process of organizing objects into groups whose members are similar in some way. The primary goal of clustering is to determine the intrinsic grouping in a set of unlabeled data. K-Means (KM) is one of the widely used algorithms in clustering techniques. KM is the simplest unsupervised learning algorithms that can solve the well-known clustering problem. Ant Colony Optimization (ACO) is one of the most popular evolutionary algorithms inspired from nature and utilized in the field of clustering. Thus ACO can be defined as the adaptive heuristic search algorithm premised on the evolutionary ideas of natural behavior of ants. ACO is used to initialize the KM clustering algorithm. This will help in better clustering result with lesser error rates. From the experimental results, it is observed that the usage of ACO results in better clustering result when compared to existing techniques (KM).

Keywords: K-Means, Ant Colony Optimization, Clustering, Genetic Algorithm

INTRODUCTION

The fundamental data clustering problem is defined as the process of discovering groups in data or grouping similar objects together. Each of these groups is known as a cluster. A cluster is a region in which density of objects is locally higher than in other regions.

Figure 1 represents the clustering technique, in which there are four clusters into which the data can be divided; the similarity criterion is distance: two or more objects belong to the same cluster if they are close according to a given distance. This is called as distance-based clustering.

Basic K-Means (KM) Algorithm

K-mean is formally described by algorithm. The operation of KM is illustrated in Figure 2, which shows how, starting from three centroids the final cluster is found in four assignments updates steps. In these and other figure displaying KM clustering each subfigure shows (1) the centroids at the start of the iteration; and (2) the assignment

[1] Department of Computer Applications, Dr. SNS Rajalakshmi College of Arts and Science, Coimbatore-46.

Figure 1: Clustering Based on Distances

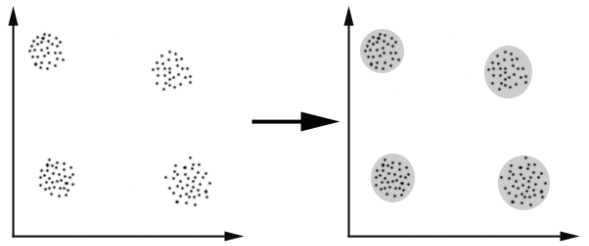
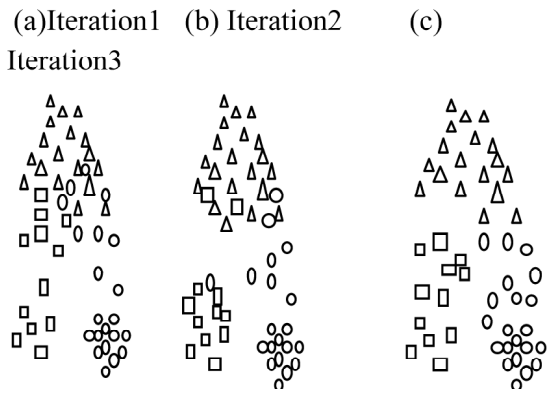


Figure 2: K-Means Clustering



of the point to those centroids. The centroids are indicated by the "+" symbol; all points belonging to the same cluster have the same marker shape.

Algorithm

1. Select K point as initial centroids.
2. Repeat
3. From K cluster by assigning each point to its closest centroid
4. Until Centroids do not change.

In the first step show in Figure 2 point are assignments to the initial centroids which are all in large group of points. For this example, we use the mean as the centroids. After points are assigned to a centroid is the updated. Then the figure for each step shows the centroid at the beginning of the step and the assignment of points to those centroids. In the second step, points are assigned to upload centroids and centroids are updates again. In step 2, 3, and 4 which are shown in Figure 2(b), 2(c) respectively two of the centroids move to the two small groups of points at the bottom of figure.

Ant Colony Optimization

Ant Colony Optimization help in improving the performance of clustering. Swarm intelligence is a relatively new approach to problem solving that takes inspiration from the social behaviors of insects and of other animals. In particular, ants have inspired a number of methods and techniques among which the most studied and the most successful is the general purpose optimization technique known as ant colony optimization. Ant colony optimization exploits a similar mechanism for solving optimization problems. From the early nineties, when the first ant colony optimization algorithm was proposed, ACO attracted the attention of increasing numbers of researchers and many successful applications are now available.

LITERATURE SURVEY

Clustering plays a vital role in the field of data mining. The clustering method implemented should be very fast, efficient, and robust. Various clustering techniques are available but the most popular one is the KM clustering approach.

Pena *et al.* (1999) compares empirically four initialization methods for the KM algorithm: random, Forgy, MacQueen and Kaufman. Although this algorithm is known for its robustness, it is widely reported in the literature that its performance depends upon two key points: initial clustering and instance order. The author conduct a series of experiments to draw up (in terms of mean, maximum, minimum and

standard deviation) the probability distribution of the square-error values of the final clusters returned by the KM algorithm independently on any initial clustering and on any instance order when each of the four initialization methods is used. The results of our experiments illustrate that the random and the Kaufman initialization methods outperform the rest of the compared methods as they make the KM more effective and more independent on initial clustering and on instance order. In addition, the author compares the convergence speed of the KM algorithm when using each of the four initialization methods. The results showed that the Kaufman initialization method induces to the KM algorithm a more desirable behavior with respect to the convergence speed than the random initialization method.

Gabriela Trazzi Perim in (2008) suggests the use of methods for finding initial solutions to the KM algorithm in order to initialize Simulated Annealing and search solutions near the global optima. Clustering (Guha *et al*., 2000; Ramze Rezaee *et al*., 1998; Shehata *et al*., 2006) has been effectively applied in a variety of engineering and scientific disciplines such as psychology, biology, medicine, computer vision, communications, and remote sensing. Cluster analysis organizes data (a set of patterns, each pattern could be a vector measurements) by abstracting underlying structure. The grouping is done such that patterns within a group (cluster) are more similar to each other than patterns belonging to different groups. Thus, organization of data using cluster analysis employs some dissimilarity measure among the set of patterns. The dissimilarity measure is defined based on the data under analysis and the purpose of the analysis. Various types of clustering algorithms have been proposed to suit different requirements. Clustering algorithms can be widely categorized into hierarchical and partitional algorithms based on the structure of abstraction. Hierarchical clustering algorithms construct a hierarchy of partitions, represented as a dendrogram in which each partition is nested within the partition at the next level in the hierarchy. Partitional clustering algorithms generate a single partition, with a specified or estimated number of non-overlapping clusters, of the data in an attempt to recover natural groups present in the data.

Krishna and Murty (1999) proposed a novel hybrid Genetic Algorithm (GA) that finds a globally optimal partition of a given data into a specified number of clusters. GA's used earlier in clustering employ either an expensive crossover operator to generate valid child chromosomes from parent chromosomes or a costly fitness function or both. To circumvent these expensive operations, the author hybridize GA with a classical gradient descent algorithm used in clustering viz., KM algorithm. The KM operator is defined, one-step of KM algorithm, and use it in GKA as a search operator instead of crossover. The authors also define a biased mutation operator in particular to clustering called distance-based-mutation. Using finite Markov chain theory, the author proves that the GKA converges to the global optimum. It is observed in the simulations that GKA converges to the best known optimum corresponding to the given data in concurrence with the convergence result. It is also observed that GKA searches faster than some of the other evolutionary algorithms used for clustering.

Shih-Sian Cheng *et al*. (2004) presented a GA for KM clustering. Instead of the widely applied string of group numbers encoding, the author encode the prototypes of the clusters into the chromosomes. The crossover operator is

intended to exchange prototypes between two chromosomes. The one-step KM algorithm is used as the mutation operator. Therefore, the proposed GA is called the prototypes-embedded genetic KM algorithm (PGKA). With the inherent evolution procedure of evolutionary algorithms, PGKA has superior performance than the classical KM algorithm (Ramze Rezaee *et al*., 1998), while comparing to other GA-based approaches, PGKA is more efficient and suitable for large scale data sets.

Krishna and Murty proposed a new clustering method called genetic KM algorithm (GKA) (Krishna and Murty, 1999), which hybridizes a genetic algorithm with the KM algorithm. This hybrid approach combines the robust nature of the genetic algorithm with the high performance of the KM algorithm. As a result, GKA will always converge to the global optimum faster than other genetic algorithms.

Ke-Zong Tang *et al*. (2010), proposed an Improved Genetic Algorithm (IGA) based on a novel selection strategy is presented to handle nonlinear programming problems. Each individual in selection procedure is represented as a three-dimensional feature vector composed of objective function value, the degree of constraints violations and the number of constraints violations.

Huawang Shi and Yong Deng (2009) proposed an IGA incorporates simulated annealing into a basic genetic algorithm with momentum that enables the algorithm to perform genetic search over the subspace of local optima. The computational results suggest that the IGA algorithm have good ability of solving the problem and the performance of IGA is very promising because it is able to find an optimal or near-optimal solution for the test problems and indicated that IGA was successful in evolving ANNs.

An effective technique in locating a source based on intersections of hyperbolic curves defined by the time differences of arrival of a signal received at a number of sensors is proposed by Lichun Li *et al*. in (2005). By making use of the knowledge of the cell's ID, the approach uses coverage shrinkable IGA to search the position coordinates. It is an approximation of the maximum-likelihood estimator and is shown to attain the Cramer-Rao lower bound. Comparisons of performance with the fixed coverage genetic algorithm and the Chan's method are made. The proposed method has higher accuracy than the fixed coverage algorithm and follows closely to the Cramer-Rao bound even at high noise level.

METHODOLOGY

KM is considered one of the major algorithms widely used in clustering. However, it still has some problems, and one of them is in its initialization step where it is normally done randomly. Another problem for KM is that it converges to local minima.

KM Clustering Algorithm

KM is one of the algorithms that solve the well known clustering problem. The algorithm classifies objects to a pre-defined number of clusters, which is given by the user (assume k clusters). The idea is to choose random cluster centers, one for each cluster. These centers are preferred to be as far as possible from each other. Starting points affect the clustering process and results. After that, each point will be taken into consideration to calculate similarity with all cluster centers through a distance measure, and it will be assigned to the most similar cluster, the nearest cluster center. When this assignment process is over, a new center will be calculated for each cluster using the points in it. For each

cluster, the mean value will be calculated for the coordinates of all the points in that cluster and set as the coordinates of the new center. Once we have these k new centroids or center points, the assignment process must start over. As a result of this loop we may notice that the k centroids change their locations step by step until no more changes are made. When the centroids do not move any more or no more errors exist in the clusters, we call the clustering has reached a minima. Finally, this algorithm aims at minimizing an objective function, which is in this case a squared error function. The algorithm is expressed as follows:

1. Choose random k points and set as cluster centers.
2. Assign each object to the closest centroid's cluster.
3. When all objects have been assigned, recalculate the positions of the centroids.
4. Go back to Steps 2 unless the centroids are not changing.

Proposed Methodology: Ant-Based Clustering

Clustering is concerned with the division of data into homogenous subgroups. Informally, the objective of this division is two-fold: data items within one cluster are required to be similar to each other, while those within different clusters should be dissimilar. Problems of this type arise in a variety of disciplines ranging from sociology and psychology, to commerce, biology and computer science, and algorithms for tackling them continue to be the subject of active research.

The ant algorithm is mainly based on the version. A number of slight modifications have been introduced that improve the quality of the clustering and, in particular, the spatial separation between clusters on the grid. The basic ant algorithm starts with an initialization phase, where

i. All data items are randomly scattered on the grid
ii. Each agent randomly picks up one data item and
iii. Each agent is placed at a random position on the grid.

Following the initialization phase, the sorting phase starts. The sorting phase is a simple loop, where

i. One agent is randomly selected;
ii. The agent performs a step of a given stepsize (in a randomly determined direction) on the grid and
iii. The agent (probabilistically) decides whether to drop its data item.

In the case of a 'drop'-decision, the agent drops the data item at its current grid position (if this grid cell is not occupied by another data item), or in the immediate neighbourhood of it (it locates a nearby free grid cell by means of a random search). It then immediately searches for a new data item to pick up. This is done using an index that stores the positions of all 'free' data items on the grid. The process is as follows:

The agent randomly selects one data item out of the index, proceeds to its position on the grid, evaluates the neighborhood function $f^*(i)$, and (probabilistically) decides whether to pick up the data item. It continues this search until a successful picking operation occurs. Only then the loop is repeated with another agent.

For the picking and dropping decisions the threshold formulae for a given grid position and a particular data item '*i*' is calculated using Equations (1) and (2):

$$p_{pick}(i) = \left(\frac{k*}{K* + f(i)} \right)^2 \quad ...(1)$$

$$p_{drop}(i) = \begin{cases} 2\ f(i) & if\ f(i) < k^- \\ 1 & otherwise \end{cases} \quad ...(2)$$

where k+ and k- are constants and $f(i)$ is a neighborhood function as given in Equation

$$f(i) = \max\left(0, \frac{1}{\sigma^2} \sum_{j \in L} \left(1 - \frac{d(i,j)}{\alpha}\right)\right) \quad ...(3)$$

where $d(i, j) \in [0, 1]$ is a measure of the dissimilarity between data points i and j, $a \in [0, 1]$ is a data-dependent scaling parameter, and σ^2 is the size of the local neighborhood L. An extension of this algorithm is algorithm is presented where the parameter a is adaptively updated during the execution of the algorithm. This algorithm is given in Figure 3.

Figure 3: Ant Based Algorithm

```
(1)  Procedure Lumer and Faieta
(2)     randomly scatter data items on the toroidal grid
(3)     randomly place agents on the toroidal grid
(4)     for t = 1 to max_iterations
(5)        j = random agent
(6)        move agent j randomly by stepsize grid cells
(7)        l = does agent j carry a data item?
(8)        e = is agent j's grid position occupied by a data
           item?
(9)        if (l = TRUE) and (e = FALSE) then
(10)          i = data item carried by agent j
(11)          drop = (random() ≤ Pdrop(i))   // see
              equations (2) and (3)
(12)          if drop = TRUE then
(13)             Let agent j drop data item i at its current
                 position
(14)          end if
(15)       end if
(16)       if (l = FALSE) and (e = TRUE) then
(17)          i = data item at agent j's grid position
(18)          pick = (random() ≤Ppick(i)) // see equations
              (1) and (3)
(19)          if pick = TRUE then
(20)             let agent j pick up data item i
(21)          end if
(22)       end if
(23)    end for
(24) end procedure
```

After discovering the URLs, each URL group is assigned an unique number. After this, in the preprocessed log data, each user's requested URL is substituted with its corresponding number. The result of this step is a file that contains records, where each record represents the navigational sequence of the user in numbers. During pattern discovery, each user's navigational sequence is defined as an array. The size of the array is equal to the number of groups identified in the previous step. The element *'i'* is 1 if the related user has seen on the pages in group *'i'*, otherwise it is set to 0. Dissimilarity of two sequences s1 at point i and s2 at point j in the grid is computed through the following Equation (4).

$$d(i, j) = \frac{\sqrt{\sum_{k=1}^{N} (s1_k - S2_k)^2}}{N} \quad ...(4)$$

where N is the number of groups and $d(i, j)$ becomes 1 when two sequences do not have any similar elements, and becomes 0 when they are exactly the same.

The output of this program is a grid that contains numbers: >–1 and =–1 indicates if there is/is not a data item, respectively. So, clusters should be extracted according to these numbers and size of the local neighborhood, s^2.

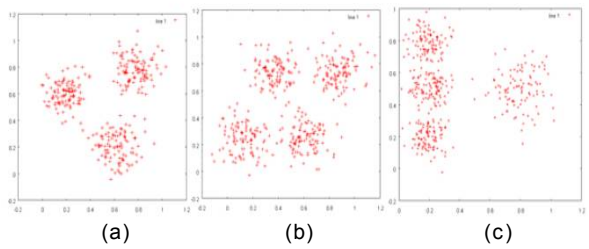
With the generated centriod points, clustering is performed. This will result in better clustering result with reduction in error rate and improvement in time and accuracy.

EXPERIMENTAL RESULTS

Data Sets

The data sets used in these experiments are iris and lymphoma.The experiments are performed over a mathematically generated 2D datasets. Figure 4 shows the dataset distribution. Dataset 1 is made based on a mathematical model to form their clusters with small amount of points interleaving.

Figure 4: Data Set Distribution
(a) Wine (b) Dataset 1 (c) Lymphoma



Dataset 3 Consists of 100 points scattered around 4 specific points with a radius of 0.2. Points are (0.125, 0.25), (0.625, 0.25), (0.375, 0.75), (0.875, 0.75). The first two points' clusters have horizontal interleaving on the boundary. In addition, points two and three have the same thing in common. This dataset is to be clustered into 4 clusters.

Wine data set the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. Dataset contains 178 instances of 3 classes.

The Lymphoma data set is a data set about the three most prevalent adult lymphoid malignancies. It contains 62 samples consisting of 4,026 genes spanning three classes, which include 42 Diffuse Large B-Cell Lymphoma (DLBCL) samples, nine Follicular Lymphoma (FL) samples, and 11 B-cell Chronic Lymphocytic Leukemia (B-CLL) samples. This dataset is to be clustered into 3 clusters. The 62 samples are randomly split into 50 training samples and 12 testing samples at each trial and the average error and time has been obtained for k-mean, GAIK, KIMGA.

Performance Evaluation

The average error and average time taken by the KM with ACO approach is quite high. Finally, running ACO as an initializer to KM definitely guides to the best solution among the group, this appears clearly from the results obtained.

Figure 5 shows the performance of KM, GAIK, and KM with ACO.

From the Figure 6, it is clear that the proposed approach shows very less error rate compared to other methods.

Similarly Figure 6 shows the bar chart representation of KM, GAIK, and KM with ACO comparing their average time in different data sets.

Figure 5: Resulted Error Rate

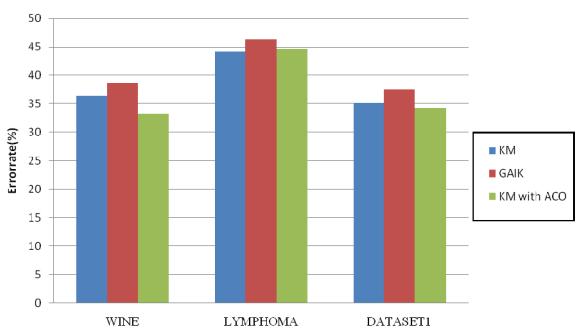
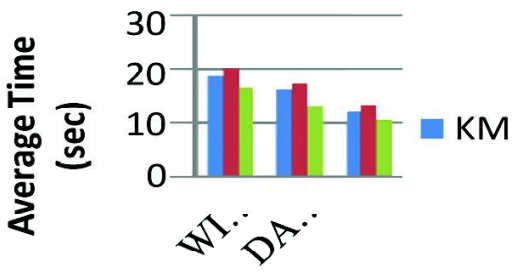


Figure 6: Average Time for Clustering



CONCLUSION

Clustering is one of the most important tools in the field of data mining, statistical data analysis,

data compression, and vector quantization. KM is one of the widely used algorithms in clustering techniques. KM is the simplest unsupervised learning algorithms that can solve the well-known clustering problem.

To reduce the error rate, this paper focused on using ACO. The intention of using ACO is its capability of convergence and it follows the behavior of ant. The experimental evaluation scheme was used to provide a common base of performance assessment and comparison with other methods.

Finally, when comparing the experimental results of KM, GAIK and KM with ACO it is observed clearly that KM with ACO is better than the simple genetic algorithm. As shown by the results on all datasets KM with ACO is ready to achieve high clustering accuracy if compared to other algorithms.

REFERENCES

1. Gabriela Trazzi Perim, Estefhan Dazzi Wandekokem and Flavio Miguel Varejao (2008), "K-Means Initialization Methods for Improving Clustering by Simulated Annealing", *Advances in Artificial Intelligence – IBERAMIA*, Vol. 5290/2008, pp. 133-142.
2. Guha S, Rastogi R and Shim K (2000), "ROCK: A Robust Clustering Algorithm for Categorical Attributes", *Information System*, Vol. 25, No. 5, pp. 345-366.
3. Huawang Shi and Yong Deng (2009), "Application of Improved Genetic Algorithms in Artificial Neural Networks", Proceedings of the 2009 International Symposium on Information Processing (ISIP'09) Huangshan, P R China, August 21-23, 2009, pp. 263-266.
4. Ke-Zong Tang, Ting-Kai Sun and Jing-Yu Yang (2010), "An Improved Genetic Algorithm Based on a Novel Selection Strategy for Nonlinear Programming Problems", *Original Research Article Computers & Chemical Engineering*, in Press, Corrected Proof, Available online on July 8, 2010,
5. Krishna K and Murty M (1999), "Genetic K-Means Algorithm", *IEEE Transactions on Systems, Man, and Cybernetics,* Vol. 29, NO. 3, pp. 433-439.
6. Krishna K and Narasimha Murty M (1999), "Genetic K-Means Algorithm", *IEEE Transactions On Systems, Man, And Cybernetics—Part B: Cybernetics,* Vol. 29, No. 3.
7. Lichun Li and Feng Wei, "Position Estimation by Improved Genetic Algorithm for Hyperbolic Location".
8. Pena J M, Lozanoa J A and Larranagaa P (1999), "An Empirical Comparison of Four Initialization Methods for the K-Means Algorithm", *Pattern Recognition Letters*, Vol. 20, No. 10, pp. 1027-1040.
9. Ramze Rezaee B P F, Lelieveldt and Reiber J H C (1998), "A New Cluster Validity Index for the Fuzzy C-Mean", *Pattern Recognition Letters*, Vol. 19, pp. 237-246.
10. Shehata S, Karray F and Kamel M (2006), "Enhancing Text Clustering Using Concept-based Mining Model", in Proceedings of the 6th IEEE International Conference on Data Mining (ICDM).
11. Shih-Sian Cheng, Yi-Hsiang Chao, Hsin-Min Wang and Hsin-Chia Fu (2004), "A Prototypes-Embedded Genetic K-means Algorithm".

International Journal of Engineering Research and Science & Technology
Hyderabad, INDIA. Ph: +91-09441351700, 09059645577
E-mail: editorijlerst@gmail.com or editor@ijerst.com
Website: www.ijerst.com