Volume 4, Issue 1, January 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Performance Enhancement of K-Means Clustering Algorithms for High Dimensional Data sets

Amita Verma, Ashwani kumar

Computer Science Engineering Department

Guru Jambheshwar University Science &Technology, India

Abstract- Data mining has been defined as "The nontrivial extraction of implicit, previously unknown, and potentially useful information from data". Clustering is the automated search for group of related observations in a data set. The K-Means method is one of the most commonly used clustering techniques for a variety of applications. This paper proposes a method for making the K-Means algorithm more effective and efficient; so as to get better clustering with reduced complexity. In this paper, the most delegate algorithms K-Means and enhanced K-Means were examined and analyzed based on their basic approach. The best algorithm in each category was found out based on their performance using Distance measure.These proposed algorithm is implemented and analyzed using a clustering tool WEKA.

Keywords- Data mining, Clustering, K-Means Clustering, Distance measure.

I. INTRODUCTION

Clustering is a division of data into groups of similar objects. Each group, called a cluster, consists of objects that are similar between them- selves and dissimilar compared to objects of other groups. Cluster analysis is a very important technology in Data Mining. It divides the datasets into several meaningful clusters to reflect the data sets' natural structure. Cluster is aggregation of data objects with common characteristics based on the measurement of some kind of information. There are several commonly used clustering algorithms, such as K-means, Density based and Hierarchical and so on. [2] Data clustering is a process of putting similar data into groups. A clustering algorithm partitions a data set into several groups such that the similarity within a group is larger than among groups.[3] Clustering is an unsupervised classification mechanism where a set of patterns (data), usually multidimensional is classified into groups (clusters) such that members of one group are similar according to a predefined criterion.

Clustering of a set forms a partition of its elements chosen to minimize some measure of dissimilarity between members of the same cluster .Clustering algorithms are often useful in various fields like data mining, pattern recognition, learning theory etc[14].

Terms:

Cluster: A cluster is an ordered list of objects, which have some common characteristics. The objects belong to an interval [a, b], in our case [0, 1]

*Distance between Two Clusters***:** The distance between two clusters involves some or all elements of the two clusters. The clustering method determines how the distance should be computed.

*Similarity***:** A similarity measure SIMILAR (Di, Dj) can be used to represent the similarity between the documents. Typical similarity generates values of 0 for documents exhibiting no agreement among the assigned indexed terms, and 1 when perfect agreement is detected. Intermediate values are obtained for cases of partial agreement.

*Average Similarity***:** If the similarity measure is computed for all pairs of documents (Di, Dj) except when i=j, an average value AVERAGE SIMILARITY is obtainable. Specifically, AVERAGE SIMILARITY = CONSTANT SIMILAR (Di, Dj), where i=1, 2….n and j=1, 2….n and i < > j

*Threshold***:** The lowest possible input value of similarity required to join two objects in one cluster.

*Similarity Matrix***:** Similarity between objects calculated by the function SIMILAR (Di, Dj), represented in the form of a matrix is called a similarity matrix.

Dissimilarity Coefficient: The dissimilarity coefficient of two clusters is defined to be the distance between them. The smaller the value of dissimilarity coefficient, the more similar two clusters are.

*Cluster Seed***:** First document or object of a cluster is defined as the initiator of that cluster i.e. every incoming object's similarity is compared with the initiator. The initiator is called the cluster seed.[6]

II. RELATED WORK

Comparisons Between Data Clustering Algorithms

Osama Abu Abba, Computer Science Department, Yarmouk University, Jordan [2].This paper is intended to study and compare different data clustering algorithms. The algorithms under investigation are: k-means algorithm, hierarchical clustering algorithm, self-organizing maps algorithm and expectation maximization clustering algorithm. All these algorithms are compared according to the following factors: size of dataset, number of clusters, type of dataset and

type of software used. Some conclusions that are extracted belong to the performance, quality, and accuracy of the clustering algorithms.

A Comparative Study of Various Clustering Algorithms in Data Mining

Manish Verma, Mauly Srivastava, Neha Chack, Atul Kumar Diswar, Nidhi Gupta [3]. This paper reviews six types of clustering techniques- k-Means Clustering, Hierarchical Clustering, DB Scan clustering, Density Based Clustering, Optics, EM Algorithm. These clustering techniques are implemented and analyzed using a clustering tool **WEKA**. Performance of the 6 techniques are presented and compared.

Performance analysis of k-means with different initialization methods for high dimensional data

Tajunisha and Saravanan[4].In this paper, we have analyzed the performance of our proposed method with the existing works. In our proposed method, we have used Principal Component Analysis (PCA) for dimension reduction and to find the initial centroid for k-means. Next we have used heuristics approach to reduce the number of distance calculation to assign the data point to cluster. By comparing the results on iris data set, it was found that the results obtained by the proposed method are more effective than the existing method.

A New Method for Dimensionality Reduction using K-Means Clustering Algorithm for High Dimensional Data Set

D.Napoleon, S.Pavalakodi [5].K-means clustering algorithm often does not work well for high dimension, hence, to improve the efficiency, apply PCA on original data set and obtain a reduced dataset containing possibly uncorrelated variables. In this paper principal component analysis and linear transformation is used for dimensionality reduction and initial centroid is computed, then it is applied to K-Means clustering algorithm.

Evolving limitations in K-means algorithm in data mining and their removal"

Kehar Singh,Dimple Malik and Naveen Sharma[6] *K*-means is very popular because it is conceptually simple and is computationally fast and memory efficient but there are various types of limitations in k means algorithm that makes extraction somewhat difficult. In this paper we are discussing these limitations and how these limitations will be removed.

Comparative Analysis of Two Algorithms for Intrusion Attack Classification Using KDD CUP Data set

N.S.Chandolikar & V.D.Nandavadekar[7].This paper evaluate performance to two well known classification algorithms for attack classification. Bayes net and J48 algorithm are analyzed The key ideas are to use data mining techniques efficiently for intrusion attack classification.

Compression, Clustering, and Pattern Discovery in Very High-Dimensional Discrete-Attribute Data Sets

Mehmet Koyutu¨rk, Ananth Grama, and Naren Ramakrishnan[8].This paper presents an efficient framework for error-bounded compression of high-dimensional discrete-attribute data sets. Such data sets, which frequently arise in a wide variety of applications, pose some of the most significant challenges in data analysis. Sub sampling and compression are two key technologies for analyzing these data sets. The proposed framework, PROXIMUS, provides a technique for reducing large data sets into a much smaller set of representative patterns, on which traditional (expensive) analysis algorithms can be applied with minimal loss of accuracy. We show desirable properties of PROXIMUS in terms of run time, scalability to large data sets, and performance in terms of capability to represent data in a compact form and discovery and interpretation of interesting patterns. We also demonstrate sample applications of PROXIMUS in association rule mining and semantic classification of term-document matrices. Our experimental results on real data sets show that use of the compressed data for association rule mining provides excellent precision and recall values (above 90 percent) across a range of problem parameters while reducing the time required for analysis drastically. We also show excellent interpretability of the patterns discovered by PROXIMUS in the context of clustering and classification of terms and documents. In doing so we establish PROXIMUS as a tool for both preprocessing data before applying computationally expensive algorithms and directly extracting correlated patterns.

A Hierarchical Latent Variable Model for Data Visualization

Christopher M. Bishop and Michael E. Tipping[9]. We introduce a hierarchical visualization algorithm which allows the complete data set to be visualized at the top level, with clusters and sub clusters of data points visualized at deeper levels. The algorithm is based on a hierarchical mixture of latent variable models, whose parameters are estimated using the expectation-maximization algorithm. We demonstrate the principle of the approach on a toy data set, and we then apply the algorithm to the visualization of a synthetic data set in 12 dimensions obtained from a simulation of multiphase flows in oil pipelines, and to data in 36 dimensions derived from satellite images.

A Modified K-Means Algorithm for Circular Invariant Clustering

Dimitrios Charalampidis Member [10].This paper introduces a distance measure and a K-means-based algorithm, namely, Circular K-means (CK-means) to cluster vectors containing directional information, such as Fd, in a circular-shift invariant manner. A circular shift of Fd corresponds to pattern rotation, thus, the algorithm is rotation invariant. An efficient Fourier domain representation of the proposed measure is presented to reduce computational complexity. A split and merge approach (SMCK-means), suited to the proposed CK-means technique, is proposed to reduce the possibility of converging at local minima and to estimate the correct number of clusters. Experiments performed for textural images illustrate the superior performance of the proposed algorithm for clustering directional vectors Fd, compared to the alternative approach that uses the original K-means and rotation-invariant feature vectors transformed from Fd.

Enhanced Moving K-Means (EMKM) Algorithmfor Image Segmentation

Fasahat Ullah Siddiqui and Nor Ashidi Mat Isa[11]. This paper presents an improved version of the Moving K Means algorithm called Enhanced Moving K-Means (EMKM) algorithm. In the proposed EMKM, the moving concept of the conventional Moving K-Means (i.e. certain members of the cluster with the highest fitness value are forced to become

the members of the clusters with the smallest fitness value) is enhanced. Two versions of EMKM, namely EMKM-1and EMKM-2 are proposed. The qualitative and quantitative analyses have been performed to measure the efficiency of both EMKM algorithms over the conventional algorithms (i.e. K-Means, Moving K-Means and Fuzzy C-Means) and the latest clustering algorithms (i.e. AMKM and AFMKM). It is investigated that the proposed algorithms significantly outperform the other conventional clustering algorithms.

A Modified Version of the K-Means Algorithm with a Distance Based on Cluster Symmetry

Mu-Chun Su and Chien-Hsing Chou[12]. In this paper, we propose a modified version of the K-means algorithm to cluster data. The proposed algorithm adopts a novel non metric distance measure based on the idea of point symmetry. This kind of point symmetry distance can be applied in data clustering and human face detection. Several data sets are used to illustrate its effectiveness.

An Efficient k-Means Clustering Algorithm: Analysis and Implementation

Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y.Wu[13].In this paper, we present a simple and efficient implementation of Lloyd's k means clustering algorithm, which we call the filtering algorithm. This algorithm is easy to implement, requiring a kd-tree as the only major data structure. We establish the practical efficiency of the filtering algorithm in two ways. First, we present a data-sensitive analysis of the algorithm's running time, which shows that the algorithm runs faster as the separation between clusters increases. Second, we present a number of empirical studies both on synthetically generated data and on real data sets from applications in color quantization, data compression, and image segmentation.

A Modified *k*-means Algorithm to Avoid Empty Clusters

Malay K. Pakhira [14].This paper presents a modified version of the *k*-means algorithm that efficiently eliminates this empty cluster problem. We have shown that the proposed algorithm is semantically equivalent to the original *k*-means and there is no performance degradation due to incorporated modification. Results of simulation experiments using several data sets prove our claim.

Comparison the various clustering algorithms of weka tool

Narendra Sharma, Aman Bajpai, Mr.Ratnesh Litoriya [15].In this paper we are studying the various clustering algorithms. Cluster analysis or clustering is the task of assigning a set of objects into groups (called clusters) so that the objects in the same cluster are more similar (in some sense or another) to each other than to those in other clusters. Our main aim to show the comparison of the different-different clustering algorithms on WEKA and find out which algorithm will be most suitable for the users.

III. CLUSTERING TECHNIQUES

Traditionally clustering techniques are broadly divided in hierarchical and partitioning. Hierarchical clustering is further subdivided into agglomerative and divisive. The basics of hierarchical clustering include Lance-Williams formula, idea of conceptual clustering, now classic algorithms SLINK, COBWEB, as well as newer algorithms CURE and CHAMELEON. While hierarchical algorithms build clusters gradually (as crystals are grown), partitioning algorithms learn clusters directly. In doing so, they either try to discover clusters by iteratively relocating points between subsets, or try to identify clusters as areas highly populated with data. Partitioning Relocation Methods. They are further categorized into probabilistic clustering (EM framework, algorithms SNOB, AUTOCLASS, MCLUST), K-Medoids methods (algorithms PAM, CLARA, CLARANS, and its extension), and k-means methods(different schemes, initialization, optimization, harmonic means, extensions).Such methods concentrate on how well points fit into their clusters and tend to build clusters of proper convex shapes.

Partitioning algorithms of the second type are surveyed in the section Density-Based Partitioning. They try to discover dense connected components of data, which are flexible in terms of their shape. Density-based connectivity is used in the algorithms DBSCAN, OPTICS, DBCLASD, while the algorithm DENCLUE exploits space density functions. These algorithms are less sensitive to outliers and can discover clusters of irregular shapes. They usually work with low-dimensional data of numerical attributes known as spatial data. Spatial objects could include not only points, but also extended objects (algorithm GDBSCAN) numerical values for attributes. The WEKA Simple K-Means algorithm uses Euclidean distance measure to compute distances between instances and clusters.

3.2.1 K-means algorithm:

Clustering technique in data mining has received a significant amount of attention from machine learning community in the last few years and become one of the fundamental research areas. Among the vast range of clustering algorithms, K-means is one of the most popular clustering algorithms. The basic principle of the K-means algorithm is to know how different distance measure is defined. It is a critical issue for K-means users. For example, how can we select a unique distance measure method for an optimum clustering task?

K-means algorithm follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters). The main idea is to define k centroids, one for each cluster. The simple K-means algorithm chooses the centroid randomly from the data set. The next step is to take each data belonging to a given data set and associate it to the nearest centroid. The K-means clustering partitions a data set by minimizing a sum of squares cost function.

Simple k- means method:

(1) arbitrarily choose *k* objects from *D* as the initial cluster centers;
(2) repeat
(3) (re)assign each object to the cluster to which the object is the most similar,

based on the mean value of the objects in the cluster;
(4) update the cluster means, i.e., calculate the mean value of the objects for each cluster;
(5) until no change

Proposed algorithm:

Our proposed algorithm uses standard deviation that reduce the time to make the cluster in simple k-mean.The main contribution is to divide the square root distance with standard deviation by this we have developed a new algorithm enhance k-means that optimize a distance measure method that gives good result as compare to simple k-means clustering algorithm.

Enhance k- means algorithms:

Input:

Number of desired clusters, k, and a database D= {d1, d2,…dn}containing n data objects.

Output:

A set of k clusters with reduce execution time as compare to simple k- means.

Steps:

1 Randomly select k data objects from dataset D as initial cluster centers.
2 Repeat;
3 Calculate the distance between each data object di (1 <i<=n) and all k cluster centers cj(1<=j<=k) and assign data object di to the nearest cluster.
4 Initialize an integer variable p.
5 Assign the euclidean distance value to p.
6 Initialize another integer variable t.
7 Square of p divided by standard deviation that value is assigned to t integer.
8 For each cluster j (1<=j<=k), recalculate the cluster center.
9 until no changing in the center of clusters .

Finally, the generated results are superior in terms of incorrectly classified cluster instances. We consider a wide range of problems in our experiment. The generated were classes to cluster evaluation and found the higher accuracy of these clusters were observed. Therefore, this research contributed for K-means algorithm as a faster algorithm and optimal clustering performance. The effectiveness and efficiency of the new algorithm are demonstrated by our experiments on WEKA system environment.

IV. EXPERIMENTAL SETUP AND RESULTS

Four data sets from the Tunedit Repository are used, all of which contains only numeric attributes and class attributes. The information about the data sets is tabulated in Table 1.

TABLE 4.1: DATA SETS USED

Data set	**Size**	**Attribute**	**Class**
Breast cancer	28	10	2
Pen digits	724	17	10
Spice	511	62	3
Vehicle	64	19	4

These Four algorithms have their implemented source code in the Weka 3.6.6 version upon which simulations have carried out in order to measure the performance parameters of the algorithms over the datasets. The results are summarized in the following tables:

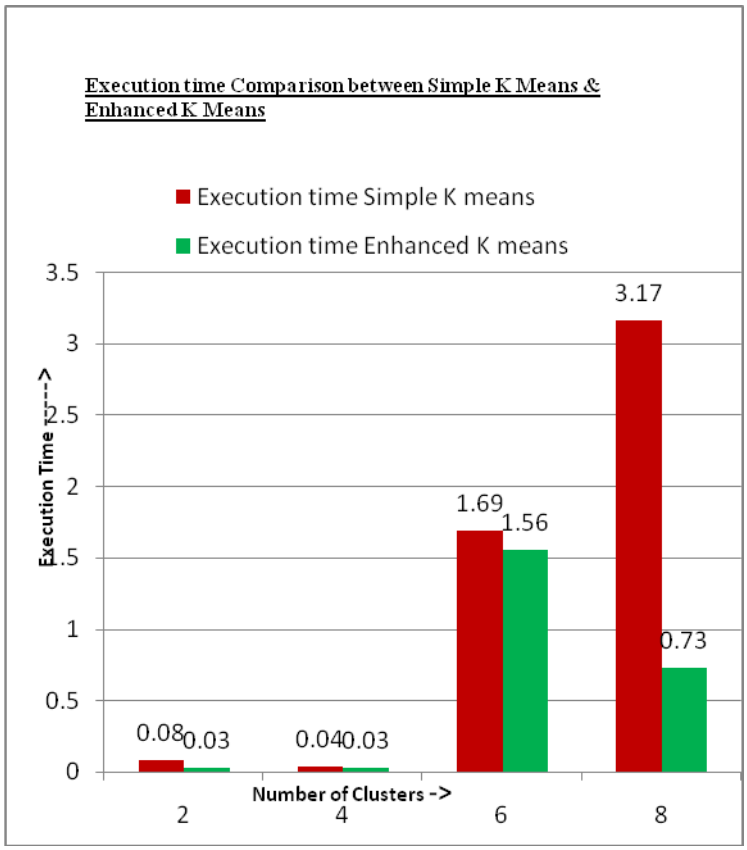
Experimental results:

Table 4.2: Simple K-Means Clustering Performance

Dataset	**No. of classes**	**No of clusters**	**Execution time**	**No. of iterations**
Breast cancer	2	4	0.04	4
Pen digits	10	8	3.17	17
Spice	3	6	1.69	7
Vehicle	4	2	0.08	8

Table 4.3: Enhanced K-Means Clustering Performance

Dataset	**No. of classes**	**No of clusters**	**Execution time**	**No. of iterations**
Breast cancer	2	4	0.03	4
Pen digits	10	8	0.73	3
Spice	3	6	1.56	7
Vehicle	4	2	0.03	3



V. CONCLUSION AND FUTURE SCOPE

We have proposed a modification in the simple K-means algorithm and the experiments prove that with this modification the clustering performance drastically increase, by changing the distance similarity measure. The performance of proposed algorithm is tested across Four real world datasets and the results are quite encouraging and have established the effectiveness of the proposed algorithms.

No work is perfect done for the first time. There is always a scope for the improvement. The proposed work can also be further explored in the light of the following suggestions:

- Application of other filters various filtering algorithm can be used according to the data sets requirement for data preprocessing.
- In the future, we plan to use some other distance measure which will enhance the performance and optimization of the clustering algorithm.
- In simple k-means algorithm there are different parameters used. To change these parameters will get the good results .

REFRENCES

[1] Johannes Grabmeier,Fayyad, Mannila, Ramakrishnan,“ *Techniques of Cluster Algorithms inData Mining*,”May 23 2001.

[2] Osama Abu Abbas, Jordan, **“***Comparisons Between Data Clustering Algorithms***,”**The International Arab Journal of Information Technology, vol. 5, no. 3, pp.320-326,Jul. 2008.

[3] Manish Verma, Mauly Srivastava, Neha Chack, Atul Kumar Diswar, Nidhi Gupta,“ *A Comparative Study of Various Clustering Algorithms in Data Mining*,” International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com, vol. 2, Issue 3, pp.1379-1384,May-Jun. 2012.

[4] Tajunisha and Saravanan,“*Performance analysis of k-means with different initialization methods for high dimensional datasets, ”*International Journal of Artificial Intelligence & Applications (IJAIA), vol. 1, no.4, pp.44-52,Oct. 2010.

[5] D.Napoleon,S.Pavalakodi,”*A New Method for Dimensionality Reduction using K-Means Clustering Algorithm for High Dimensional Data Set*,” International Journal of Computer Applications (0975– 8887),vol. 13, no.7, pp.41-46, Jan 2011.

[6] Kehar Singh, Dimple Malik and Naveen Sharma, “*Evolving limitations in K-means algorithm in data mining and their removal,* **”**IJCEM International Journal of Computational Engineering &Management, vol. 12, pp.105-109,Apr. 2011.

[7] N.S.Chandolikar,V.D.Nandavadekar,“*Comparative Analysis of Two Algorithms for Intrusion Attack Classification Using KDD CUP Dataset***,”**International Journal of Computer Science and Engineering(IJCSE),vol.1,pp.81-88,Aug 2012.

[8] Mehmet Koyutu¨rk, Ananth Grama and Naren Ramakrishnan,"*Compression, Clustering and Pattern Discovery in Very High-Dimensional Discrete-Attribute Data Sets*,"IEEE Transactions on Knowledge and A Data Engineering", vol. 17, no. 4, pp.447-461, Apr 2005.

[9] Christopher M. Bishop and Michael E. Tipping, "*A Hierarchical Latent Variable Model For Data Visualization*," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 3, pp.281-293, Mar. 1998.

[10] Dimitrios CharalampidisI,"*A Modified K-Means Algorithm for Circular InvariantClustering*," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 12, pp.1856-1865, Dec 2005.

[11] Fasahat Ullah Siddiqui and Nor Ashidi Mat Isa,"*Enhanced Moving K-Means (EMKM) Algorithmfor Image Segmentation*,"IEEE, pp.833-841.

[12] Mu-Chun Su and Chien-Hsing Chou, "*A Modified Version of the K-Means Algorithm with a Distance Based on Cluster Symmetry*,"IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, pp.674-680, Jun. 2001.

[13] TapasKanungo,David M.Mount ,NathanS. Netanyahu, Christine D.Piatko, Ruth Silverman, and Angela Y.Wu,"*An Efficientk-Mean Clustering Algorithm: Analysisan Implementation* ,"IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 881-891,Jul 2002.

[14] Malay K. Pakhira, "*A Modified k-means Algorithm to Avoid Empty Clusters*," International Journal of Recent Trends in Engineering, vol. 1, no. 1, pp.220-226, May 2009.

[15] Narendra Sharma ,Aman Bajpai,Mr.Ratnesh Litoriya*," Comparison the various clustering algorithms of weka tools*," International Journal of Emerging Technology and Advanced Engineering, vol. 2, pp.73-80, May 2012.

