

ENCHANCED K-MEANS ALGORITHM TO IMPROVE TIME EFFICIENCY

D. Narmadha¹, Dr. G. Naveen Sundar²,
S. Jebapriya³ and Dr. M. Malathy⁴

^{1 2 3}Karunya Institute of Technology and Sciences
(KITS)

narmadha@karunya.edu

⁴Vignana Bharathi Institute of Technology
(VBIT)

naveensundar@karunya.edu,

jebapriya@karunya.edu,

malathianandan@gmail.com

January 10, 2018

Abstract

Clustering means grouping the data which has similar features. Clustering can be done using many algorithms. One such algorithm is Centroid-Based clustering which includes k-means, k-modes, kmedoids. This paper gives an insight about k-means algorithm, its iterations and solution to decrease the iterations and also increases its accuracy. The working principles of K-means algorithm is based on random selection of initial cluster centers. Using the initial cluster centers the distance of each data point from its centroid will be calculated and the grouping will be done. This leads into more number of iterations in the algorithm. When the number of iteration increased the time efficiency also get decreased. Here we proposed an enhanced K-means algorithm which focuses on decreasing the number of iterations.

The examined optimized k-means algorithm shows not only the speed gets increased but also the accuracy which is parallelly increased.

Key Words : clustering, Centroid based clustering, k-means clustering

1 Introduction

Clustering is group of data in which each item present in the same group is almost similar. The centroid of each clusters and its members from various clusters are never be closest to one another. Clustering algorithms are mainly divided into two main classifications such as hierarchical and partition based clustering. Data is clustered into reduced subsets using hierarchical manner. However, the data is clustered into smaller clusters[1] in a single step. Information retrieval from a huge amount of data is very important. Dimensionality reduction is carried out by using linear and non-linear dimensionality reduction techniques. By reducing the dimensionality reduction, the processing time will be reduced. This can be done using MapReduce in Hadoop. MapReduce has two major functions, they are mapped with similar functions and reduce, where map function is used to map the key value pair, the output of the map function will be given as input for the reduce function. Finally the reduced dataset will be the output of the Hadoop MapReduce method [2]. Cluster analysis method is used to group N number of observations into K number of clusters in which each number of observations (N) belongs to the clusters which is having the closest average value or mean value. There are enormous numbers of cluster analysis routines, from which we can get different results. In order to reduce the average distance based on the random value of k from each data point to its closest center, most commonly used statistical method are used. To categorize groups of objects from a large data points, we need a set of n data points in dimensional space and an integer k. Lloyd's algorithm is one of the popular experiential method for k-means clustering. Lloyd's k-means clustering algorithm is an effectual method which is also referred as filtering algorithm. The implementation of Lloyd's k-means clustering algorithm is simple to apply in any of the applications. The

major data structure like kd-tree is the basic requirement for the implementation of this algorithm. Establishing the practical efficiency of this algorithm is done in two ways. The very first way is to analyze this particular algorithm's running time, this will show us the difference between the clusters performance and hence the algorithm's running time increases and becomes faster while executing. The methodology is also applicable to a quite a lot of other interdisciplinary research problems such as finding the drug for the diseases, treating patients with similar kind of symptoms. In this paper, an improved k-means algorithm is introduced based on storing of values which repeatedly comes during iterations. This paper includes analyzing of traditional k-means algorithm and shows the accuracy and time taken by the traditional k-means algorithm. The next section introduces the optimized k-means algorithm. Both algorithms take the same data type for the experiments to have good and accurate results.

2 Literature Review

Hung et al [3] performed simple partitioning in the k-means clustering algorithm. They proposed that the entire data is converted into unit block (UB) and the centroids of the unit blocks (CUB) are found using simple calculations. Hence the process are done step by step using k means algorithm. The performance is good compared to other clustering algorithm. The Proposed system by (Hung et al 2005) has three processes. They are partitioning the data into blocks, calculating centroids of (Unit blocks) UB to from simple view of dataset, calculating the final clusters. The results are compared with direct k-means and Asabis algorithm and it is found that the execution time and numbers of iteration are very much less on k-means algorithm.

Fahim et al [4] improved the k-means algorithm. The problem in k-means algorithm is to determine the set of k points. The proposed method is most suitable for large number of clusters because k-means algorithm while working on the large dataset makes the computation process very expensive and the processing time also exceeded. Hence k-means algorithm is not well suitable for large data sets. In the proposed method, the distance will be calculated

between each data point, based upon the distance, the data points will be associated to the nearest clusters. For the next iteration, instead of going for the data points distance, the clusters will compute the distance to the previous nearest clusters. If the new distance which is calculated among the cluster points is less than or equal to the previous distance, there is no change in data point stay, and it will remain in its same cluster, and there is no need for the distance to be computed among other cluster centers. This process saves the time required for the distance computation to $k1$ cluster centers.

Ahmad Al-Taani et al [5] they proposed a system which has five stages, First step is training the data. Next step leads to noise Elimination, thirdly learning process. Fourth step is the classification methods and finally testing stages. The clustering scheme is built based upon the already known category which derived from the training documents. Fuzzy relation is used to test the similarity between a test document and the category. This relation is called fuzzy term-category relation, where the set of membership degree of words to a particular category represents the cluster prototype of the learned model. Based on this relation and using fuzzy conjunction and disjunction operators, the similarity between a document and a category's cluster center will be calculated. The membership degree of the test document to the category is calculated using similarity representation, and each membership function takes a value between 0 and 1. By using this value, we can test the testing different test documents in order to come out with the weak points as well as the strength points.

Marcus Vinicius et al [6] proposed two approaches in hierarchical methods. First approach is agglomerative, otherwise called as Bottom-up. second approach refers divisive, otherwise called as Top-down ones. In the agglomerative hierarchical or the bottom-up approach, the data objects were randomly distributed that leads to the representation of cluster. These clusters are recursively clustered enacting into consideration based upon some measures of similarity which leads the example to be merged into one single cluster. Hence, in the beginning, the clusters exist in reduced numbers with a high degree of similarity between their elements but throughout the process these groups start to increase and their elements become dissimilar data.

Deepayan Chakrabarti et al [7] focus on automatic template de-

tection by using a training data. Production of web pages using the improved utilization of content management systems has considerably enhanced the browsing skill of end users. By Using Site navigation links, copyright notices, sidebars and timestamps, the users are provided with easy access and essential content. The presence of templates diverts the users from the major theme. Moreover, templates can degrade the performance of search engines, including the ranking function, index and duplicate detection. The first observation in the page level template detection method is to build a page level classifier. Initially, site level template detection method is applied on numerous randomly chosen sites. Then, suitable features for these site-level templates are defined and extracted. Lastly, automatically generated data is used to train a classifier in order to allocate a templateness score to all nodes in a Document Object Model tree. The classifier can determine templates that are obvious only at the page-level. The second observation in this approach is the usage of global property that creates a relationship between the templateness scores across nodes of the DOM tree. Templateness relates to a monotone property: If all the children of a node are template, then the node in the DOM tree is assumed to be a template. When page level template detection is used as a pre-processing step in web mining applications, accuracy can be improved significantly. Template detection can be performed incrementally which consumes very less storage. However, template is detected only with the help of past content.

Song et al [8] proposed focuses on jointly optimizing wrapper generation and template detection. The method utilizes similarities between web pages for template detection as well as for generating wrappers. Both the processes will occur in a single step and it will be quite accurate, when compared to earlier methods that sense templates based on similarity measures of URLs. It performs both wrapper generation and template detection in one step by utilizing similarities between pages. The approach requires manually labeled training data which is an overhead.

3 TRADITIONAL KMEANS ALGORITHM

The traditional algorithm consists of two functions. The first phase of kmeans algorithm is to select k centroids arbitrarily and the second phase is to group every data based on its nearby centroid value where k value is generated randomly. Using the distance between every cluster centers and each data object, the Euclidian distance can be calculated. The k-means algorithm frequently congregates to the resident minutest. For each and every iteration the distance between every data objects from the cluster center can be calculated. In case if there are large number of iterations present in the traditional k-means algorithm, then the algorithm is in need of kn comparisons for every data. This results in the drastic increase in the computational time of the traditional k-means algorithm. The time complexity of this algorithm can be denoted as $O(nkd)$ where d denotes the sum of all data points. The K-means algorithm has its some shortcomings in its time efficiency. This paper shows that in an easy way using the above analysis while in some iterations data object will belongs to the same cluster and rest of the iterations, data object belongs to another cluster. If a data object x remains for a long period of time in same cluster, then algorithm takes $x(n-1)$ time more to run the iteration.

4 OPTIMIZED KMEANS ALGORITHM

In traditional k-means algorithm, the main disadvantage is spending most of its time by calculating distance from each clusters centre during each iterations. This spare time can be hold back by introducing a new solution in which the iterations gets reduced and obviously the speed is increased. If the distance between the data object and a new cluster center is less than the distance between the data object and an existing cluster center then it can come to the conclusion of belonging to same cluster, so in these circumstances the distance calculation of the data point from the other cluster center is not needed. It is done by opting two simple data structures as we discussed earlier which is used for the storing the label of clusters and distance between the data object with respective

cluster center and the stored value can be used for the upcoming iterations, which helps in avoiding the repeated calculations.

Input:

The number of clusters is given as an input which is termed as k , for the dataset D_n (where n is the natural number)

Output:

Data has been divided into k clusters

1. Initial clustering is done for the given dataset with random centroids
2. Distance is calculated from each and every data point from all centroids
3. For every data point, calculating the nearest center c_j with the data object and assigning the value d_i to the cluster centre c_j .
4. Accumulate the label and distance as: Set $\text{cluster}[i]=j$, and $\text{dist}[i]=d(d_i, c_j)$.
5. For every cluster, once again the calculate of cluster centre is needed.
6. For every data point. Calculate its distance between the data object with the existing cluster; if the calculated distance is less than or equal to $\text{dist}[i]$, the data object from the cluster stays in the initial cluster;
7. Step 5 till same centers found

The optimized and the enhanced algorithm requires two types of data structure cluster and $\text{dist}[i]$ is used to store label of clusters and distance metrics of the clusters respectively for each iteration which is used in further iterations. The solution we discussed in this paper avoids calculating distance for each and every iteration. The time complexity which is used in this optimized algorithm is $O(nk)$. If a range of the point present in the same cluster, then the time complexity will be represented as $O(1)$ otherwise $O(k)$. If some data points in original cluster group move to other cluster then the time complexity can be calculated by $(nk/2)$. The proposed

optimized algorithm results in increasing the performance speed of standard k-means algorithm. For iteration process, however this algorithm also requires the value of k well in advance. The optimal solution for this algorithm can only be obtained when we test for different values of k.

5 RESULTS AND DISCUSSION

The dataset was collected from UCI machine learning repository. This dataset includes electricity in prominent locations such as automobile industry, University, Food Industry, Chemical Industry and apartments. In the upcoming graph, values taken while experimenting two algorithms are computed. The operating system used for this experiment is Ubuntu and the tool used is NetBeans for performing the coding and storing data in a specific data structure.

Table 1: No of Records vs Execution Time for various Algorithm

Algorithm	No of records	Execution time(sec)
KM	50	23
	100	24
	500	26
	1000	28
Enhanced K-Means	50	18
	100	19
	500	19
	1000	20

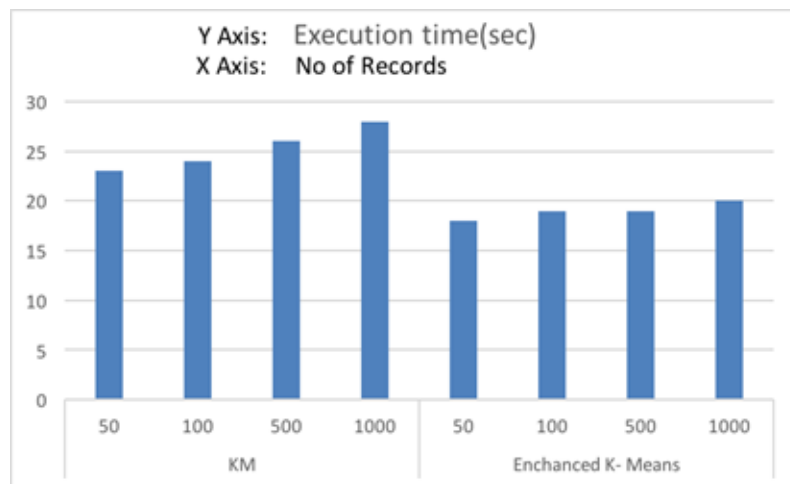


Fig 1: No of Records vs Execution Time (sec)

We evaluated both the algorithms with the same dataset. The original k-means algorithms computational time and the enhanced k-means algorithms computational time has been calculated. Table 1 shows the values which represents the performance comparison between these algorithms. The results attain from these values shows that the proposed algorithm has less computational time which exposed in the graph.

6 CONCLUSION

K-means algorithm is one of the traditional clustering algorithms. In this paper, the time complexity has been analyzed for the traditional K-means algorithm and we developed an improved method for reducing the computational time. The proposed optimized algorithm uses the formulae of $O(nk)$ for reducing the time complexity. While comparing the time complexity between the traditional K-means algorithm and the improved method, the computational time is less in enhanced K-means algorithm. Our result shows that the time taken for executing the total iterations is less than that of traditional k-means. Thus it concludes that the algorithm explained is feasible.

References

- [1] Chen Zhang, Shixiong Xia, *K-means Clustering Algorithm with Improved Initial center*, in Second International Workshop on Knowledge Discovery and Data Mining (WKDD), pp. 790-792, 2009.
- [2] Alina Ene Sungjin Im Benjamin Moseley, *Fast Clustering using MapReduce* Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, Pp. 681-689 .
- [3] Ming-Chuan Hung, Jungpin Wu, Jin-Hua Chang & Don-Lin Yang, *An Efficient k-Means Clustering Algorithm Using Simple Partitioning*, Journal Of Information Science And Engineering, vol. 21, pp. 1157-1177(2005).
- [4] Fahim, AM, Salem, AM, Torkey, FA & Ramadan, MA, *An efficient enhanced k-means clustering algorithm*, Journal of Zhejiang Univ.-Sci. A, vol. 7, no. 10, pp. 1626-1633(2006).
- [5] Ahmad T Al-Taani & Noor Aldeen K Al-Awad, *A Comparative Study of Web-pages Classification Methods using Fuzzy Operators Applied to Arabic Web-pages*, World Academy of Science, Engineering and Technology International Journal of Computer, Electrical, Automation, Control and Information Engineering, vol. 1, no. 7, pp. 2160- 2162(2007).
- [6] Marcus,V.C. Guelpele, Ana Cristina Bicharra Garcia & Flavia Cristina Bernardini 2007, *An Analysis of Constructed Categories for Textual Classification Using Fuzzy Similarity and Agglomerative Hierarchical Methods*, Third International IEEE Conference on Signal-Image Technologies and Internet-Based System, pp.92-97.
- [7] Deepayan Chakrabarti, Ravi Kumar & Kunal Punera 2008, *A graph-theoretic approach to webpage segmentation*, Proceeding of the 17th international conference on World Wide Web, pp. 377-386.

- [8] Song, R, Wen, J, Wu, D & Zheng, S 2007, *Joint Optimization of Wrapper Generation and Template Detection*, Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, California, USA, pp. 894-902.

