ISSN 2347 - 3983
Volume 8. No. 5, May 2020
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter20852020.pdf
https://doi.org/10.30534/ijeter/2020/20852020

A Comparative Study on K-Means Clustering and Agglomerative Hierarchical Clustering

Karthikeyan B.[1], Dipu Jo George[2], G. Manikandan[3], Tony Thomas[4]
[1]School of Computing, SASTRA Deemed University, India, mbalakarthi@gmail.com
[2]Information Technology, Viswajyothi College of Engineering & Technology, India, dipugeorge97@gmail.com
[3]School of Computing, SASTRA Deemed University, India, manikandan@it.sastra.edu
[4]Information Technology, Viswajyothi College of Engineering & Technology, India, tony.thomas.vtt@gmail.com

ABSTRACT

Clustering is a well-established unsupervised data mining approach that group data points based on similarities. Clustering entities will give insights into the characteristics of different groups. Clustering results in minimization of the dimensionality of data set when you are dealing with a myriad number of data. The higher the homogeneity within the cluster and the higher the differences between the clusters, the finer the cluster will be. Clusters are mainly of two types: 1) Soft clustering: Based on the probability that a data point will belong to a specific cluster and, 2) Hard clustering: Data points are separated into independent clusters. Among hundreds of clustering algorithms, they can be labeled into one of following models such as connectivity, density, distribution and centroid model. This paper attempts to differentiate two widely used clustering techniques, k-means clustering and hierarchical clustering which belong to the centroid and connectivity models respectively. The comparison will be based on execution time and memory usage of both these algorithms when different sets of a delivery fleet driver data set are manipulated using these algorithms.

Key words : agglomerative hierarchical clustering, centroids, dendrograms, k-means clustering.

1. INTRODUCTION

Clustering is the method of separating given dataset into different sets in such a way that dataset within the cluster have greater similarities and dataset between clusters have more of dissimilarities [2]. Cluster analysis has some important functions in data mining and related fields such as pattern recognition, pattern classification, data discovery, vector quantization and data compression. Also, the role of clustering is inevitable in marketing, physics, biology, geography, and geology [19]. Clustering algorithms are generally categorized into two: hierarchical clustering and non-hierarchical clustering [3] [18].

K-means algorithm, a centroid-based algorithm, belongs to the non-hierarchical clustering group. The simplicity and the efficiency of k-means algorithm to upon even on large datasets is a major advantage over other clustering algorithms [1]. According to k-means algorithm, initially number of clusters k and centroid values will be defined. Then the algorithm separates the given dataset into k clusters using the minimum distance between centroid and a data point [2]. The algorithm iterates and recalculates centroid values after each iteration. This iteration continues till the value of centroid doesn't varies. The minimum distance calculation can be using many methods such as: Euclidean distance, Mahalanbois distance [6] and a number of other techniques. K-means have already found application in horticulture, astronomy, image processing, market division, weather forecasting, bioinformatics and computer vision [9] [7].

A method of bottom-up clustering where each cluster have their sub-clusters is known as agglomerative hierarchical clustering which will also have sub-clusters for the corresponding sub-clusters and so on [2]. Gene expression data also might exhibit hierarchy within a single cluster. In each of its further iterations, it then agglomerates the nearby cluster pairs by fulfilling some similarity criterions, till all the corresponding data are in one of these clusters [5]. Some of the major applications of agglomerative hierarchical clustering are tracking viruses with the help of Phylogenetic trees, charting evolution with Phylogenetic trees, and it is also used to generate DNA sequences from the described datasets. The major benefit of hierarchical clustering is that it produces an ordering for the objects that could be informative for the display of the data.

2. LITERATURE SURVEY

K-means is an unsupervised and iterative machine learning approach. It is functional for most practical data set but when it comes to large data set; selection of initial cluster centers, noise data points and the count of clusters k to be initialized remains to be an issue. In paper [1], authors proposed a refined k-means clustering that uses two additional data structures to store feature of the cluster center where the data

point is and the interval of data point to adjacent cluster in two separate arrays. This refined k-means algorithm reduced computational and time complexity without affecting the accuracy of clusters since unnecessary iterations were avoided.

Later, authors in the paper [2] tried to separate noise data points in a given sample to increase the accuracy of the resulting cluster. Initially, a preprocessing on dataset was conducted to separate outliers using outlier detection method based on LOF. This was done to exclude the participation of outliers in the estimation of cluster centers. And then, a refined k-means clustering developed by Aristidis Likas was applied on a previously generated data set. This new k-means algorithm was found to be highly efficient to exclude the interference caused by outliers but it cost more time when applied to large datasets.

In paper [3], authors came up with a new method to assign initial k-means cluster centers. This was accomplished by selecting two principal variables which are the maximum coefficient of dissimilarity and minimum coefficient of the association. Then, normalized data set was clustered until a defined number of times using the prime cluster centers. The newly proposed algorithm was effective and consistent as compared to random initialization of the cluster centers.

Authors of paper [4], tried to predict the likely behavior from the relationships found within a given data set. They suggested that clustering and classification together provide the best solution for revealing hidden patterns within a set. Clustering was used to group similar data under the same label. They tried to predict the weather behavior using k-means clustering and probability density function algorithm. In their work, they generated numerical results using probability density function algorithm for predictions, which were earlier clustered using the k-means clustering. K-means clustering was chosen due to reduced clustering errors.

Paper [5] showed that the gradient descent feature of the k-means clustering made the algorithm highly responsive due to assignment of cluster centers initially. Therefore, a study conducted in the paper [5] analyzed some well-established linear time complexity initial setting techniques using large and variant number of data sets using different performance criterion. Computational efficiency was the central unit of comparison. And finally, the experiment results were analyzed based on non-statistical tests and the recommendations were made. The study revealed that well-established initialization methods such as forgy, maximin and Macqueen didn't perform well in terms of computational efficiency.

Work done in the paper [6], tried to use Mahalanbois distance measurements instead of the traditional Euclidean distance metrics. During the experiment, authors found that it was a straight forward approach even though, the initial calculation of covariance matrices was a bit complicated. The experiment was conducted on various clusters having different shapes and it was found that the Mahalanbois distance, when applied in k-means clustering, worked well with clusters having elliptical shapes. Also, it was concluded that Mahalanbois distance won't work properly without strategic initialization of covariance matrices.

Paper [7] tried to stimulate the operation of k-means using three methods: 1) Distributed memory with the message passing (MPI), 2) Heterogeneous computing with NVIDIA GPU setup using the CUDA-C and, 3) Shared memory using the OpenMP. They evaluated all the three methods on images ranging from small images (300*300 pixels) to large images (1164*1200 pixels). All three of the methods gave nearly 35 times the speed a sequential k-means could have provided for the same data set. It was evident that the shared memory with the OpenMP worked well for small images while GPU with CUDA-C worked well for larger images.

Paper [8] gave insights into an empirical comparison between clustering using k-means algorithm and hierarchical clustering. This comparative study was performed using a numerical data set. The study concluded that the k-means outperformed other hierarchical methods regarding computational complexity.

Paper [9] presents a detailed comparison study performed upon clustering algorithms like fuzzy c-means, k-means and k-means++. Comparisons are based on elapsed time and number of iterations. These clustering algorithms are applied for sorted and unsorted data respectively. When sorted data was passed into these algorithms, elapsed time was lower than that for unsorted data. This was due to reduced time complexity and number of iterations for sorted data. Also, there will be a minimal fluctuation of cluster centers.

In paper [10], the authors present methods of grouping the data for numerical and categorical data since the grouping is different for categorical and numerical data due to their discrete characteristics. The similarity measurement is determined as the least occurrences of any attribute in multiple clusters that administers tight regulation on merging of these clusters whose intra similarity is very high. If 'n' number of tuples are found in data set, then the similarity matrix can be evaluated with a complexity of $O(n^2)$. The numerical data is mostly grouped with respect to geometric properties such as distances between them and the categorical attributes are grouped with the help of dissimilarity formula.

In paper [11], the authors propose a hierarchical approach which is built on the semi-supervised ultra-metric dendrogram measurements which is incorporable with triple-wise relative limitations. They also establish a connection within the hierarchical clustering with ultra-metric conversion of the dissimilarity matrix. They explicitly establish an agreement connecting hierarchical clustering with ultra-metrics and provides a blended framework integrating the ultra-metric fitting and the triple-wise relative limitations. The proposed structure entreats an approximate metric of dissimilarity thus representing a tuned dendrogram which agree with the given

limitations. Two techniques were developed in solving this problem.

Authors of paper [12] found that if the data are unlabeled, it will be difficult to handle those collections and hence the computational cost will be high. Therefore, they suggested a new method for agglomerative hierarchical clustering using centroids instead of raw data points. They tested this method on different clustering techniques, data arrangements, and distance estimates. The authors came up with a new KnA method which improves the efficiency of hierarchical clustering using a set of sub-clusters generated by k-means technique. This method has computational and cost related advantages over the standard hierarchical approach without sacrificing the performance of clustering. A strong association was found between using centroids and using objects.

Paper [13] discusses a clustering approach known as agglomerative hierarchical clustering with Tanagra tool. Tanagra tool contains numerous algorithms to execute clustering. In the experiment mastermind by the authors, hierarchical clustering is utilized for the examination of data. It classifies ownership with almost indistinguishable features into a cluster that executes on gap parameters and sum of the square. A class label is not required to be mentioned as clustering is an unsupervised learning. It functions by separating samples of datasets into classes or clusters. Inside the cluster, the separation between the centroids as well as the cluster instances must be minimum.

In paper [14], the association among objects is explained by proximity matrix where rows and columns comply with objects. The only one input provided to clustering algorithm in this paper is known as proximity matrix. This paper to study various types of hierarchical clustering algorithms. The main problem with hierarchical clustering is that, if the merge process is done, it can't be left. In order to enhance the standard of hierarchical clustering it could be integrated with other techniques by using any of the algorithms which was then followed by another algorithm. It was found that complexity of the agglomerative clustering is O(n3) and hence this technique is slower for huge data sets.

In paper [15], the authors introduced an advanced hierarchical algorithm implemented with the Euclidean distance and validated the method with various experiments with feigned data set of low dimensional and a fMRI data set which is data set of high dimensions. The proposed method assures that the nearby points come under the same cluster. The newly proposed algorithm is a category of bottom-up agglomerative hierarchical clustering method. The grid-based algorithm is one of the fastest algorithms having low time for processing and depends on the grid size instead of data set. Comparing with the results of k-means clustering, agglomerative clustering and the newly implemented hierarchical clustering method, the authors affirm that the accuracy of the newly purposed method is greater than other compared algorithms, but the time for computations are higher when working with data sets with higher dimensions.

Paper [16] explains the creation of spark oriented hierarchical clustering using numeric spaces. Framework like spark quickly processes huge datasets by splitting them into independent blocks that are addressed in parallel. It is usually difficult to parallelize clustering algorithms functionally due to high reliance on the data used. Authors of this paper demonstrates SHAS which is an algorithm for employing the Spark framework which minimises the single-linkage clustering problem to least spanning tree issue in the corresponding complete graph created by the input data set. The proposed algorithm was memory efficient and also linearly scalable. On evaluating SHAS with two data set generated from more than one distribution, it was seen that it attained a huge speedup.

In paper [17], the authors found that hierarchical clustering techniques are particularly used for analysing genetic data sets in advancements in biology studies because of the inherited hierarchical relations among related sequences derived from similar particles. With the help parallel computing technologies, using the latest methods to implement hierarchical clustering in a highly efficient way without sacrificing the efficiency of the results. They suggest using hierarchical clustering for small data points as well as medium data points in service of more resourceful clustering outputs. In order to choose a method for distance matrix Calculations, considering the long execution time they recommended the use of alignment-free distance.

3. EXPERIMENTAL SETUP

The experiments were conducted in Python 2.7.15 supported by Ubuntu 18.04.1. Our aim was to study and find differences in the k-means and the agglomerative hierarchical algorithm regarding execution time and memory usage (RSS, VMS, Shared and Data). A delivery fleet driver dataset was used for comparison, which contained the distance covered by driver and the average speed throughout the journey. All the values were numerical values and, were unsorted. RSS as called Resident Set Size is the measure of main memory (RAM) occupied by the executing process. VMS (Virtual Memory Size) is the measure of virtual memory held by the process. Shared memory is the measure of memory allocated for other processes. Data memory also known as Data Resident Size, is the measure of physical memory dedicated to the process excluding the executable process and ensures performance among all the processes. Five rounds of comparison were performed using a varying number of datasets; 200, 400, 600, 800 and finally 1000. Execution time was recorded in seconds and memory utilization was measured in kB (kilobytes).

4. COMPARISON RESULTS

The comparison between both the algorithms revealed certain characteristics. When the comparisons were based on execution time as given in Table 1 revealed that for smaller datasets both the algorithms showed satisfactory results even though k-means outperformed agglomerative hierarchical clustering. But as the number of datasets were gradually

increased the performance of agglomerative hierarchical started to decline gradually due to increased execution time but even then, k-means showed only minor increase in execution time.

Table 1: Comparison results

Number of Dataset	Execution Time (in seconds)	
	K-means Clustering	Agglomerative Hierarchical Clustering
200	0.0184512138367	6.6083688736
400	0.0326704978943	51.1555919647
600	0.0464200973511	174.483659983
800	0.0603318214417	410.195846081
1000	0.07493019104	744.990619898

Figure 1 gives the compared results of both the clustering based on memory utilization. For both the algorithms, difference in terms of shared and data memory are negligible. But in terms of VMS and RSS there are huge differences. As the number of datasets supplied are increased, the overall memory usage by k-means and agglomerative hierarchical increased; but this rate of increase was less in k-means as compared to that of agglomerative hierarchical. But the overall results revealed that k-means utilized more memory than agglomerative hierarchical clustering.

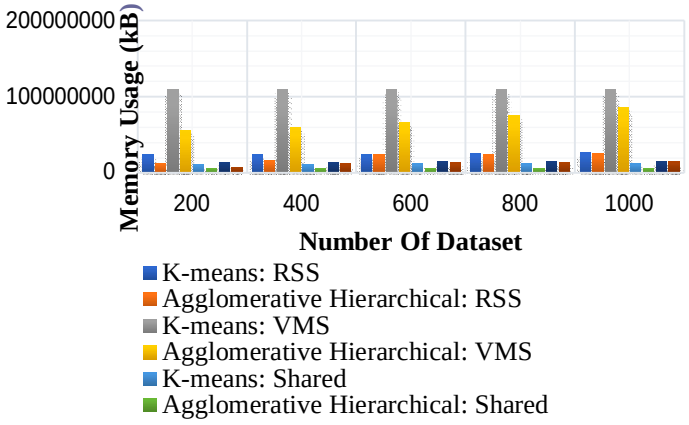

Figure 1: Comparison of k-means and agglomerative hierarchical clustering

5. CONCLUSION

The comparisons were based on execution time and memory utilized by the algorithms. Regarding the execution time, k-means performed better for small and large datasets. But for agglomerative hierarchical as number of datasets were increased the execution time increased rapidly. In terms of memory utilization, shared and data memory couldn't provide much information but RSS and VMS did. The rate of increase in memory utilization was higher for agglomerative hierarchical. But still, k-means utilized more memory than agglomerative hierarchical for all datasets. Therefore, it was concluded that k-means was more suitable for larger datasets due to lower execution time and lower rate of change in memory utilization. Also, it could be concluded that agglomerative hierarchical was suitable for smaller datasets due to lower overall memory utilization and execution time is not much of a concern due to reduced dataset.

REFERENCES

1. Liu Xumin, Shi Na, and Guan Yong, **Research on k-means clustering algorithm an improved k-means clustering algorithm**, *Third International Symposium on Intelligent Information Technology and Security Informatics*, pp. 63-67, 2010.
2. Juntao Wang, and Xiaolong Su, **An improved k-means clustering algorithm**, pp. 44-46, 2011.
3. Nazif Calis, Murat Erisoglu, and Sadullah Sakallioglu, **A new algorithm for initial cluster centers in k-means algorithm**, *Pattern Recognition Letters*, pp. 1701-1705, 2011.
   https://doi.org/10.1016/j.patrec.2011.07.011
4. Abhay Kumar, Vandana Bhattacherjee, Daya Shankar Verma, Ramnish Sinha, and Satendra Singh, **Modeling using k-means clustering algorithm**, *1st International Conference on Recent Advances in Information Technology*, 2012.
   https://doi.org/10.1109/RAIT.2012.6194588
5. M. Emre Celebi, Patricio A. Vela, and Hassan A. Kingravi, **A comparative study of efficient initialization methods for the k-means clustering algorithm**, *Expert Systems with Applications*, pp. 200-210, 2013.
   https://doi.org/10.1016/j.eswa.2012.07.021
6. Igor Melnykova, and Volodymyr Melnykov, **On k-means algorithm with the use of mahalanobis distances**, *Statistics and Probability Letters*, pp. 88-95, 2014.
   https://doi.org/10.1016/j.spl.2013.09.026
7. Ningfang Mi, Miriam Leeser, and Janki Bhimani, **Accelerating k-means clustering with parallel implementations and GPU computing**, 2015.
8. P. Praveen, and B. Rama, **An empirical comparison of clustering using hierarchical methods and k-means**, *International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics*, 2016.
9. Abhishek Singhal, and Akanksha Kapoor, **A Comparative Study of k-means, k-means++ and fuzzy**

c-means clustering algorithms, *3rd IEEE International Conference on Computational Intelligence and Communication Technology*, pp. 1-6, 2017.

10. Parul Agarwal, Ranjit Biswas, and M. Afshar Alam, **A hierarchical clustering algorithm for categorical attributes**, *Second International Conference on Computer Engineering and Applications*, pp. 365-368, 2010.
    https://doi.org/10.1109/ICCEA.2010.222
11. Li Zheng, and Tao Li, **Semi-supervised hierarchical clustering**, *11th IEEE International Conference on Data Mining*, pp. 982-991, 2011.
12. Athman Bouguettaya, Andy Song, Xiangmin Zhou, and Qi Yu, Xumin Liu, **Efficient agglomerative hierarchical clustering**, *Expert Systems with Applications*, 2014.
13. Smarika, Parul Kalra, Nisha Mattas, and Deepti Mehrotra, **Agglomerative hierarchical clustering technique for partitioning patent dataset**, 2015.
    https://doi.org/10.1109/ICRITO.2015.7359281
14. Sakshi Patel, Aman Jatain, and Shivani Sihmar, **A study of hierarchical clustering algorithms**, *2nd International Conference on Computing for Sustainable Global Development*, pp. 537-541, 2015.
15. Zahra Nazari, Yulwan Sung, M. Reza Asharif, Dongshik Kang, and Seiji Ogawa, **A new Hierarchical clustering algorithm**, *Track2: Artificial Intelligence, Robotics, and Human-Computer Interaction*, pp. 148-152, 2015.
    https://doi.org/10.1109/ICIIBMS.2015.7439517
16. Chen Jin, Ruoqian Liu, Alok Choudhary, Ankit Agrawal, William Hendrix, and Zhengzhang Chen, **A scalable hierarchical clustering algorithm using spark**, *IEEE First International Conference on Big Data Computing Service and Applications*, pp. 418-427, 2015.
17. Chee Keong Kwoh, and Thuy-Diem Nguyen, **Efficient agglomerative hierarchical clustering for biological sequence analysis**, 2015.
18. M Mallikarjuna, and R Prabhakara Rao, **Application of Data Mining Techniques to Classify World Stock Markets**, *International Journal of Emerging Trends in Engineering Research*, Vol. 7, No. 8, pp. 46-53, 2019.
19. J.Ruby Elizabeth, and S.Ebenezer Juliet, **A Survey on Various Segmentation Methods in Medical Imaging**, *International Journal of Emerging Technologies in Engineering Research*, Vol. 7, No. 11, 2019.