A Detailed Study of Clustering Algorithms

Kamalpreet Bindra[1], Anuranjan Mishra[2]

[1,2]CSE Department , Noida International University
Plot 1, Sector-17 A, Yamuna Expressway, Gautam Budh Nagar, Uttar Pradesh 203201
[1]kamalpreet.bindra@gmail.com, [2]amc290@gmail.com

Abstract — **the foremost illustrative task in data mining process is clustering. It plays an exceedingly important role in the entire KDD process also as categorizing data is one of the most rudimentary steps in knowledge discovery. It is an unsupervised learning task used for exploratory data analysis to find some unrevealed patterns which are present in data but cannot be categorized clearly. Sets of data can be designated or grouped together based on some common characteristics and termed clusters , the mechanism involved in cluster analysis are essentially dependent upon the primary task of keeping objects with in a cluster more closer than objects belonging to other groups or clusters. Depending on the data and expected cluster characteristics there are different types of clustering paradigms. In the very recent times many new algorithms have emerged which aim towards bridging the different approaches towards clustering and merging different clustering algorithms given the requirement of handling sequential ,extensive data with multiple relationships in many applications across a broad spectrum. Various clustering algorithms have been developed under different paradigms for grouping scattered data points and forming efficient cluster shapes with minimal outliers. This paper attempts to address the problem of creating evenly shaped clusters in detail and aims to study, review and analyze few clustering algorithms falling under different categories of clustering paradigms and presents a detailed comparison of their efficiency, advantages and disadvantages on some common grounds. This study also contributes in correlating some very important characteristics of an efficient clustering algorithm.**

Keywords — **clustering, proximity, similarity, CF tree, KDD, optimization.**

I. INTRODUCTION

Data is the goldmine in today's ever competitive world. Everyday large amount of information is encountered by organizations and people. An indispensible means to handle this data is to categorize or classify them into a set of groups, partitions or clusters. "Basically classification systems are either supervised or unsupervised ,depending on whether they assign new inputs to one of the finite number of discrete supervised classes or unsupervised categories respectively"[26][27] .Supervised classification is basically machine learning task of deducing a function from training data set. Whereas unsupervised classification is the exploratory data analysis where there is no training data set and extracting hidden patterns in data set with no labelled responses is achieved. The prime focus is to increase proximity in data points belonging to the same cluster and increase dissimilarity among various clusters and all this is achieved through achieved through some similarity measure.

Exploratory data analysis is related to a wide range of applications such as "text mining, engineering, bioinformatics, pattern recognition, mechanical engineering, voice mining, spatial data analysis, textual document collection, image segmentation, artificial intelligence" [2]. This diversity explains the importance of clustering in scientific research but this diversity can lead to contradictions due to different nomenclature and purpose. Many clustering approaches which are evolved with the intention of solving specific problems can many a times make presuppositions which are biased towards the application of interest.

These favoured assumptions can result in performance degradation in some problems that do not satisfy the approved assumptions in certain premises. As an example the famous partitioning algorithm K -means generates clusters which are hyper spherical in shape but when the real clusters appear in varied shapes and geometric forms, it fails miserably. This way the chosen algorithm K- means cannot be employed and other clustering schemes have to be considered [3]. So the number, nature and approach of each clustering algorithm differ on a lot of performance measures, Right from the measure of similarity or dissimilarity to feature selection or extraction to cluster validation to complexity (time/ distance). Interpretation of results is equally valuable as cluster analysis is not a one way process, in most cases lots of trials and repetitions are required. Moreover there are no standard or set of criteria for feature selection and clustering schemes. The paper attempts to survey many popular clustering schemes as there exists a vast amount of work that has been done on clustering schemes and discussion of all those numerous algorithms will be out of the scope of the paper.

Taxonomy of Clustering Algorithms:

There exist many measures and initial conditions which are responsible for numerous categories of clustering algorithms [4, 2]. A widely accepted classification frames clustering techniques as:

- Partitional clustering (sum of squared error based)

978-1-5090-3012-5/17/$31.00 ©2017 IEEE

- Hierarchical clustering
- Density based

These classifications are based on a number of factors and few algorithms have been developed bridging the multiple approaches also. In very recent times an extensive amount of algorithms have been developed to provide solution in different fields, however there is no single universal solution provided by an algorithm that solves all prevalent clustering problems. It has been very difficult to advance an integrated composition (clustering) at a specialized level and enormously diverse clustering algorithms have been seen [5]. It becomes ever crucial to discuss the various characteristics that an efficient clustering algorithm must poses in order to solve the problem at hand .Some of the characteristics are listed here:

Scalability: this is the ability of an algorithm to perform well with a large amount of data objects or say tuples, in terms of memory requirements and execution times. This feature especially distinguishes data mining algorithms from the algorithms used in machine learning [3]. Scalability remains one of the major milestones to be covered as many clustering algorithms have shown to do bad when dealing with large situations and data sets.

Prior domain knowledge: There are many clustering algorithms which require some basic domain knowledge or user is expected to provide some input parameters e.g. the number of clusters. However more often the user is unable to provide such domain knowledge, also this over sensitivity towards input parameters can degrade the performance of the algorithm as well.

Discovering arbitrary shaped clusters: One of the most challenging tasks is to identify clusters of varied shapes; few algorithms like K-means do poorly in this identification. Data attributes can be of different dimensionality altogether and a good clustering algorithm should be able to isolate data points resulting in different sizes and shapes. Few density based algorithms like DBSCAN are able to achieve this using a concept of Minpts. Many algorithms based on either centroid or medoid based approach fail to satisfy these two clustering criteria of developing widely different clusters and converging concave shaped clusters. The paper will throw some light on this topic in coming sections.

Similarity or dissimilarity measures: These measures are real valued functions that quantify the similarity between two objects .Plenty of literature can be found on these measures some of the popular measures are enlisted in table I [26].

II. HIERARCHICAL CLUSTERING

This category is a paradigm of cluster analysis to generate a sequence of nested partitions (clusters) which can be visualized as a tree or so to say a hierarchy of clusters known as cluster dendrogram. Hierarchical trees can provide a view of data at different levels of abstraction [23]. This hierarchy when laid down as a tree can have the lowest level or say leaves and the highest level or the root. Each point that resides in the leaf node has its own cluster whereas the root contains of all points in one cluster. The dendrogram can be cut at intermediate levels for obtaining clustering results; at one of these intermediate levels meaningful clusters can be found. The hierarchical approach towards clustering can be divided into two classes: a) agglomerative b) divisive. Hierarchical clustering solutions have been primarily obtained using agglomerative algorithms [19][27].

Agglomerative clustering strategies function in bottom up manner i.e. in this approach merging of the most similar pair of clusters is achieved after starting with each of the K points in a different cluster. This process is repeated until all data points converge and become members of a same cluster. There exist many different variations of agglomerative algorithms but they are primarily dissimilar in how the conflict of similarity between existing clusters and merged clusters is updated. Many different agglomerative algorithms exist depending upon the distance between 2 clusters. Very well-known methods are: 1) single linkage and 2) complete linkage technique [6]. Divisive approach on the other hand is exactly in contrast which means it works in top down manner. In divisive approach a cluster is recursively split after starting with all the points in the same cluster, this step is repeated until all points are in separate clusters. For a cluster with M objects, there are 2^{M-1} possible two-subset divisions, which incurs a high computation cost [7].

Therefore divisive clustering is considered a burden computationally. Although it has its drawbacks but this approach is implemented in two popular hierarchical algorithms DIANA and MONA [2]. Discussion of these algorithms in detail is beyond the scope of this paper. In hierarchical algorithms, previously taken steps, whether merging or splitting are irreversible even if they are erroneous. The hierarchical algorithms are relatively more prone and sensitive to noise and outliers as well as lack of robustness. To hierarchical clustering, there are many advantages and disadvantages which are reflected in many representative examples of hierarchical technique. The detailed discussion is possible only by examining popular hierarchical algorithms. Some of the representative examples are:

1. BIRCH 2.CURE 3.ROCK 4. CHAMELEON BIRCH

"Balanced iterative reducing and clustering using hierarchies" (birch) [23]. The Birch addresses two huge issues in clustering analysis 1.it is able to scale while dealing with voluminous data and 2. Robustness against noise and outliers [23].Birch introduces a novel data structure for achieving the above goals, (CF) tree clustering feature. CF tree can be assumed as a tuple, summarizing the information that is maintained about a cluster. Instead of directly using the data originally, CF tree can compress data and develop numerous tiny points or nodes .These nodes work as tiny clusters and depict the summary of original data. This CF tree is height balanced tree with two parameters: branching factors B and T,

TABLE I: measures for finding similarity

S. N.	Popular similarity measures for data points		
	Measure	**Form**	**Example**
1	Euclidean distance	$D = \left(\sum_{i=1}^{d} \| x_a - x_{b_i} \|^{1/2} \right)^2$	k-means, PAM.
2	Minkowski distance	$D = \sum_{i=1}^{d} \|x_a - x_b\|^{1/n}$ n	Fuzzy c-means
3	Cosine distance	$S_{ij} = \cos\alpha = x_i^T x_j / \|\|x_i\|\| \|\|x_j\|\|$	k-means, many partitional algorithms

It is the threshold. In the CF tree every internal vertex comprises of entries defined as $[CF_i, child_i]$, I = 1….k, where CF_i is a summary of the cluster i and is defined as a tuple $CF_i = (N_i, LS, SS)$ where N_i = no. of data objects in the cluster. LS= the linear sum of the N data points, SS= squared sum of the objects.

The CF's are saved as leaf nodes whereas non leaf nodes comprise of summation of all the CF's of their children. A CF tree is built dynamically and incrementally, also it requires a single scan of the entire data set, when an object is inserted in the closest leaf entry, the two parameters B and T control the maximum number of children per non leaf node and the maximum diameter of sub clusters stored in the leaf node. In this manner birch constructs a framework which can be stored in main memory , after building a CF tree the next step is to employ an agglomerative hierarchical or any of the sum of square error based algorithm to perform clustering. It is quite fast and with multiple scans it gives improved results but again the inability to deal with non- spherical shaped clusters stands as the biggest drawback of birch. Arbitrary shaped clusters cannot be identified by birch as it uses the principal of measuring diameter of all clusters for determining their boundary. Encountering the above drawback, *guha,rastogi and shim* developed CURE.

CURE

It was developed for identifying more complex cluster shapes [15]. It is more robust to outliers. Cure is an agglomerative method. Instead of using a single centroid it assumes many separate fixed points as clusters and a fragment of *m* is used to shrink these diverse points towards centroids. These scattered points after shrinking represent the cluster at each iteration and the pair of clusters with the closest representatives are merged together. This feature enables CURE to identify clusters correctly and makes it sensitive to outliers. This algorithm uses two enhancements 1) random sampling and 2 partitioning. These enhancements help improve cure's scalability. Random sampling can affect memory requirements making it an expensive choice for data bases.

ROCK

Guha et al. suggested and proposed another agglomerative hierarchical algorithm, ROCK. This algorithm scales quite well with increase in dimensionality. With ROCK concept of "links" was introduced. "links are used for measuring proximity between a pair of data points with categorical attributes"[10].This algorithm uses a "goodness measure" for determining how proximate or similar clusters are. A criteria function is calculated, more the value of this function the better a cluster is. This link based approach used in rock claims to be a global approach to the clustering problem. ROCK algorithm can measure the similarity of two clusters by comparing a user specified inter connectivity measure which is static with aggregate inter connectivity of any two clusters. ROCK is highly likely to generate ambiguous results if choice of parameters is provided in the static model differ from the data set being clustered. Again, clusters that are of different sizes and shapes cannot be accurately defined by this algorithm.

CHAMELEON

This algorithm is capable of "measuring the similarity of two clusters based on a dynamic model" [14] it improves the clustering quality by using more detailed merging criteria in comparison to CURE. This method works in two phases, and measures the similarity of two clusters based on a dynamic model. During the 1st phase a graph is created which contains links between each point and its N-nearest neighbor. After that during the 2nd phase graph is recursively split by a graph partitioning algorithm resulting in many tiny unconnected sub–graphs. This algorithm iteratively combines two most similar clusters. During the 2nd phase when each sub graph is considered as an initial sub cluster, two clusters can be merged but only if the resultant cluster has similar inter connectivity and closeness to the two parent clusters prior to merging .The overall complexity of this algorithm is dependent on the exact duration of time required to develop an N-nearest neighbor graph combined with the time required to complete both phases. Chameleon is a dynamic merging algorithm and hence considered more functional as compared to Cure While dealing with arbitrary shaped clusters of uneven density.

Advantages and disadvantages: as discussed in above section, hierarchical clustering can cause trouble while handling noisy high dimensional data. In HC when merges are final they cannot be undone at a later time preventing global optimization. After the assignment of an object or data point is done, that data point is highly unlikely to be reconsidered in future even if this assignment generates a bad clustering example. Most agglomerative algorithms are a liability in terms of storage and computational requirements. The computational complexity of most HC algorithm is at least $O(N^2)$. Not only this, HC algorithms could be severely degraded when applied in high dimensional spaces due to

curse of dimensionality phenomenon. The above mentioned algorithms have each introduced their novelties towards overcoming many shortcomings of HC approach. Hierarchical approach can be a great boon when used in taxonomy tree problems for example when vertical relationships are present in data.

III. PARTITIONAL CLUSTERING

Partitional clustering is highly disimilar to hierarchical approach which yields an incremental level of clusters with iterative fusions or divisons, partitional clustering assignes a set of objects into K clusters with no hierarchical structue[3]. Research from very recent years acknowleges that partitional algorithms are a favoured choice when dealing with large datasets. As these algorithms have comparatively low computational requirements[21]however when it comes to the coherence of clustering ,this approach is less effective then agglomerative aproach. These algorithms deduce the shape of clusters as hyper-ellipsoidal and basically experiment with cutting data into n number of clusters so that partitioning of data optimizes a given criterion. Centroid based techniques as used by K-MEANS and ISODATA assign some points to clusters so that the mean squared distance of points to the centroid of the choosen cluster is minimized.The sum of squared error function is the dominant criteria function in partitional approach. It is used as a meaure of variation with in a cluster. One of the most popular partitioning clustering algorithms implementing SE is k – means.

K-MEANS

K-means is undoubtedly a very popular partitioning algorithm. It has been discovered, rediscovered and studied by many experts from different fields, by steinhaus(1965),Ball and Hall (1965),Lloyd(proposed 1957 – published 1982) and Macqueen(1967). It is distance-based and by definition data is partitioned into pre- determined groups or clusters. The distance measures used could be Euclidean or cosine. Originally a fixed K cluster centroids [11, 12] are marked at random; k-means reassigns all the points to their closest centroids and re-computes centroids of newly created groups. This iteration continues till the squared error converges. Following steps can summarize the function of k-means.

1. Initialize a K partition based on previous information. A cluster prototype matrix A=[$a_{i....}a_j$] is created. Where a1,a2,a3 …are cluster centers. Data set D is also initialized.
2. In the next step assignment of each data point in the dataset(d_i) to its nearest cluster(a_i) is performed.
3. Cluster matrix can be recalculated considering the current updated partition or until a_i,a_j,a_k….. Show no further change.
4. Repeat 2 and 3 until convergence has been reached [20].

K-means is probably the most wildly studied algorithm this is the reason why there exists too many variations and improved versions of k-means yet it can show some sensitivity towards noise and outliers present in data sets. Even if a point is at a distance from the cluster centroid, it could still be enforced to the centre and can result in distorted cluster shape. K- means does not clearly defines a universal method of deciding total number of partitions in the beginning , this algorithm relies heavily on user to provide in advance ,the number of k clusters. Also, k-means is not applicable to categorical data. Since k-means presumes that user will provide initial assignments it can produce replicated results upon every iteration.(the k-means ++ addresses this problem by attempting to choose better starting clusters [12,13].

K-MEDIODS

Unlike the k-means, in the k-medioids or partition around medoids (PAM)[13,14] method , a medoid represents any cluster . This characteristic object called the medoid, is the most centrally located point within the cluster [13]. Medoid show better results against outliers as compared to centroids [2]. K-means finds the mean to define accurate centre of the cluster which can result in extreme values but k-medoid calculates the cluster centre using an actual point. Primarily this algorithm attempts to minimize the average dissimilarity of objects against their closest object. The following steps can sum up this algorithm:

1. **Initialize**: a random *k* is selected of the *n* data points as the medoid.
2. **Assign**: each data point should be associated with the closest medoid.
3. **Update:** for every m medoid and data point d, swapping of m and d can be done compute average dissimilarity of d to all the data points associated with m.

Steps 2 and 3 can be repeated multiple times until there is no further change left in assignments.

PAM uses a greedy search resulting in failure in finding an optimum solution.

CLARANS

“Clustering large applications based on randomized search”, this method combines the sampling techniques with PAM[15].This method employs random searching techniques for finding clusters and no supplementary structure is used., a feature that makes this algorithm robust against increase in dimensionality of data .In previously discussed techniques it is assumed that distance function has to be an Euclidean but CLARANS uses a local search technique and prohibits any particular distance function. CLARANS claims to identify polygon shaped objects very effectively. A method known as “IR – approximation” is used for grouping non convex polygon as well as convex polygon objects.

ISODATA

An interesting technique called ISODATA "Iterative self organizing data analysis technique", developed by ball and Hall [17] also evaluates k(no of clusters)and is iterative in nature. A variant of k-means algorithm, ISODATA works by dynamically adjusting clusters through the process of splitting and merging depending on some thresholds defined a priori. like , C : (no. of clusters desired),D_{min} : minimum number of data points for each cluster ,V_{max}: maximum variance for splitting up clusters and M_{min} : minimum distance measure for merging.

The newly calculated clusters are used as it is for next iteration. ISODATA is able to handle the problem of outliers much better than the k-means through the splitting procedure and ISODATA can eliminate the possibility of elongated clusters as well.

Advantages and disadvantages: The most prominent advantage of partitioning based methods is that they can be used for spherical based clusters for very small to bigger sized data sets. Algorithms like k-means can tend to show high sensitivity to noise and outliers whereas other methods which show resistance against noise can prove to be computationally costly.

Apart from the computational complexity, algorithms can be compared on some other common grounds, like the capability to converge to a optimum clustering solution or the potential to create clusters of different density and shapes. Although all clustering algorithms attempt to solve the same problem but there exist performance issues which can be discussed. Table II lists some comparisons in brief. The next table lists various time and space complexity.

TABLE II: Performance evaluation of various algorithms.

Algorithm	shape	convergence	capability
k- means	Cannot handle arbitrary clusters.	K is required in advance .performance degrades with increased dimensionality.	Simple and efficient
Birch	Cannot handle arbitrary clusters.	Deals fairly with robust data. Performs strongly against noisy data.	Most famous HC algorithm
Rock	Random sampling has an impact on selection of cluster shapes.	Can handle large data sets	Usage of links gives better results with scattered points
Cure	Finds richer cluster shapes.	Cannot scale well compared to birch. Merging phenomenon used in cure makes various mistakes when handling large data sets with comparison to chameleon.	Less sensitive to outliers. A bridge between centroids based and all points approach.
Dbscan	Handles concave clusters.	Targets low dimensional spatial data.	Very popular density algorithm.
Denclue	Influence function can affect shapes	Faster and handles large data sets.	Shows better result with outliers then dbscan.
Chameleon	Fair	Scales with increase in size of data	Modify clusters dynamically. curse of dimensionality

TABLE III: Comparisons of complexity of various algorithms

S.N.	Time and space complexity		
	Algorithm	Time	Space
1	K-means	O(idkn) i = iterations	O((d +k)n) d = data points, n= attributes
2	Birch	O(d*e*p) d =data points, e= entries per node, p= path from root to leaf.	O(N)
3	Cure	$O(n^2 \log n)$	O(n)
4	Chameleon	$O(n^2)$	O(n log n)
5	Dbscan	$O(n^2)$ (worst case), O(n * t) , t= time for Eps neighbourhood.	O(n)
6	Denclue	O(n log n)	O(n)

IV. DENSITY BASED ALGORITHMS

This paradigm of clustering conceptualizes the theory of "neighborhood", Clusters are spotted such that every given N-neighborhood for some given N>0 must have some minimum number of points i.e. the "density" in N-neighborhood of points has to exceed some initial criterion (Ester et al.1996). Closeness of objects is not the benchmark here rather "local density" is primarily measured. A cluster is viewed as a set of data points scattered in the data space. In Density based clustering contiguous region of low density of objects and data exists and the distance between them is calculated. Objects which are present in low density region constitute of

outliers or noise .these methods have better tolerance towards noise and are capable of discovering non convex shaped clusters. Two most known representatives of density based algorithms are density based spatial clustering of applications with noise (DBSCAN)[17] and density based clustering (DENCLUE)[11,17].

DBSCAN

It was proposed by Martin Ester,Hans-PeterKrigel, JorgSander and Xiawoei in 1996.one of the most popular density based algorithm. It requires that the density in the neighborhood of an object should be high enough if it belongs to a cluster[16].A cluster skeleton is created with this set of core objects with overlapping neighborhood. Points inside the neighborhood of core objects represent the boundary of clusters while rest is simply noise. It requires two parameters 1) ε is the starting point and 2) Minpts, is the minimum number of points required to form a dense region. The following steps can elaborate the algorithm further:

1 An unvisited random point is usually taken as the initial point.

2. A parameter ε is used for determining the neighborhood (data space)

3. If there exist sufficient data points or neighborhood around the initial random point then algorithm can proceed and this particular data point is labeled as visited or else the point is labeled as a flaw in data or outlier.

4. If this point is considered a part of the cluster then its ε neighborhood is also the part of the cluster and step 2 is repeated for all ε. this is repeated until all points in the cluster are determined.

5. Another initial data point is processed and above steps are restated until all clusters and noise are discovered.

Although this algorithm shows excellent results against noise, it can be a failure when tested in high dimensional data sets and shows sensitivity to Minpts.

This algorithm fairs well as compared to k-means in terms of creating clustering of varied shapes.

DENCLUE

Density based clustering (DENCLUE) was developed by hinneburg and keim[11].This algorithm buys heavily from the concepts of density and hill climbing[24]. In this method there is an "influence function" which is the distance or influence between random points. Many influence functions are mainly calculated and added up to find out the "density function"[11]. So it can be said that influence function is the influence of a data point in its neighborhood and density function is the total sum of all influences of all the data points. Clusters are determined by density attracters, local maxima of the overall density function. DENCLUE has a fairly good scalability, a complexity of (O(N)) it is able to spot and converge clusters with unpredictable shapes But suffers with a sensitivity towards input parameters. DENCLUE suffers from curse of dimensionality phenomenon.

Advantages and disadvantages: density based methods can very effectively discover arbitrary shaped clusters and capable of dealing with noise in data much better than hierarchical or partitional methods, these methods do not require any pre defined specification for the number of partitions or clusters but most density based algorithms show decrease in efficiency if dimensionality of data is increased although algorithm like DENCLUE shows some escalation while dealing with high dimensionality[2] but it is still far from completely effective.

V. CONCLUSION

Cluster analysis is a very crucial paradigm in the entire process of data mining and paramount for capturing patterns in data. This paper compared and analyzed some highly popular clustering algorithms where some are capable of scaling and some of the methods work best against noise in data. Every algorithm and its underlying technique have some disadvantages and advantages and this paper has comprehensively listed them for the reader. Every paradigm is capable of handling unique requirements of user application. An extensive research and study has been done in the field of data mining and there exist popular real life examples such as Netflix, market basket analysis studies for business giants, biological breakthroughs which use complex combinations of various algorithms resulting in hybrids also and subsequently cluster analysis in the future will unveil more complex data base relationships and categorical data. There is an alarming need of some sort of benchmark for the researchers to be able to measure efficiency and validity of diverse clustering paradigms. The criteria should include data from diverse domains (text documents, images, CRM transactions, DNA sequences and dynamic data). Not just a measure for benchmarking algorithms, consistent and stable clustering is also a barrier as a clustering algorithm irrespective of its approach towards handling static or dynamic data should produce consistent results with complex datasets. Many examples of efficient clustering methods have been developed but many open problems still exist making it playground for research from broad disciplines.

REFERENCES

[1] RuiXu, Donald Wunch, "survey of clustering algorithms", IEEE transactions on neural networks , vol 16 no.3 may 2005.

[2] Amandeep Kaur Mann, and Navneet Kaur, "Survey Paper on Clustering Techniques", IJSETR: International Journal of Science,Engineering and Technology Research (ISSN: 2278-7798), vol. 2,Issue 4, April 2013.

[3] Ma hong ,kang jing,liu xiong "research on clustering algorithms of data streams",ICIME,the 2nd IEEE international conference .2010.

[4] J. Kleinberg, "An impossibility theorem for clustering," in *Proc. 2002Conf. Advances in Neural Information Processing Systems*, vol. 15,2002, pp. 463–470.

[5] Arun K. Pujari,Data mining techniques-a reference book ,pg. no.-114-147.

[6] Miao Guojun, Lijun Daun , Wang Shi, ``principal and algorithm of data mining" published in tsinhua university press ,2007

[7] He, Z., Xu, X. and Deng, S., Scalable algorithms for clustering large datasets with mixed type attributes. International Journal of Intelligence Systems.2005 v20. 1077-1089

[8] A. Hinneburg and D. A. Keim. A general approach to clustering in large databases with noise. Knowledge and Information Systems, 5(4):387-415,2003

[9] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.

[10] S. Guha, R. Rastogi, and K. Shim. ROCK: a robust clustering algorithm for categorical attributes. I18n Proc. of the 15th Int'l Conf. on Data Eng., 1999.

[11] Abdellah Idrissi,Hajar Rehioui *An improvement of denclue algorithm for the data clustering*. Information & Communication Technology and Accessibility (ICTA), 2015 5th International Conference. IEEE Xplore 10 march 2016.

[12] Renato Cordeiro de amorium "Asurvey on feature weighting based k-means algorithms" Springer journal ,vol 33 ,issue 2,pp 210-242,july 2016.

[13] Guifen chen,Yuqin yang,hang cheng, "Analysis and research of k-means algorithm in soil fertitlity based on hadoop platform",Springer, international conference on computer and computing technologies .pp304-312 . 2014

[14] CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling , George Karypis , Eui-Hong Han, Vipin Kumar IEEE Computer 32(8): 68-75, 1999

[15] Guha, R. Rastogi, and K. Shim, "CURE: An efficient clustering algorithm for large databases," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1998, pp. 73–84

[16] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc.2nd Int. Conf. Knowledge Discovery and Data Mining (KDD'96)*,pp. 226–231,1996.

[17] Xindong wu, Vipin kumar, Joydeep gosh,Qiang Yang,"top 10algorithms in data mining " Springer knowl Inf Syst. 2008

[18] Dongming Chen, "A novel clustering algorithm for graphs",IEEE Xplore digital library ,2009

[19] C.H.Q Ding ,X F he ,H Y Zha et al, "A min max cut algorithm for graph pratitioning and data clustering",Proc. Of ICDM 2001

[20] Xiao –shui xing,Zhu Li, "A novel hybrid clustering algorithm incorporating K-means into canonical immune programming algorithm".ICMT ,IEEE Xplore 2010.

[21] Zhi-Hua Zhou,Hang Li , "Advances in knowledge discovery and data mining",11th pacific asia conference ,may 2007.

[22] Keerthiram murugesan ,Jun Zang, "hybrid hierarchical clustering :an experimental analysis".technical report dept, of computer science ,university of Kentucky.2011.

[23] Tian Zhang , Raghu Ramakrishnan , MironLivny, BIRCH: an efficient data clustering method for very large databases, Proceedings of the 1996 ACM SIGMOD international conference on Management of data, p.103-114, June 04-06, 1996, Montreal, Quebec, Canada

[24] Ying zhao,George karypis,usama fayyad *Hierarchical clustering algorithms for document datasets*,min knowl disc(2005) 10 : 141.doi:10.1007/s10618-005-0361-3

[25] A K toor, A singh , "An advanced clustering algorithm for clustering large data set to achieve high dimensionality",intl. journal of applied information systems,2014.

[26] P. Berkhin. (2001) Survey of clustering data mining techniques. [Online].Available:http://www.accrue.com/products/rp_cluster_review.phttp://citeseer.nj.nec.com/berkhin02survey.html

[27] U Fayyad , Gregory Piatetsky, "from data mining to knowledge discovery in databases", AI magazine 1996.

[28] R jain "a hybrid clustering algorithm for data mining ",Cornell university, 2012.