An Unbiased Approach for Clustering Large Data

A Time Complexity Efficient Approach

Rajarshi Dev Choudhury, Akash Goenka, Mr. S.Karthik
Student
SRM University, Chennai, India

***Abstract* - Clustering, in general, is an important data mining concept which can be used in the discovery of various groups, also known as clusters, of high dimensional data, on the basis of certain factors/domains on which they can be correlated from unsorted data. The existing clustering algorithms today, the generic ones, were last constructed and developed during the advent of Big Data, however most of them fail to incorporate today's changing pattern of data with umpteen dimensions in efficient time. We propose to carry out a detailed research into the existing clustering algorithms, and thus find out on a comparative basis, the algorithm most suited to all generic purposes. We intend to base our comparison on properly calculated time complexity and further substantiate the same on the basis of data sizes as well as the number of involved dimensions, while simultaneously taking into account their possibility of existing in multiple dimensions (fuzzy based algorithms). On the completion of our analysis, we intend to incorporate our learning and the appropriate features in such a way, so to create a new algorithm which can address a uniformly disturbed data space, such that not only special cases of outliers are addressed to the best but also common errors of faulty assumptions are reduced, along with increased dependency on a strong mathematical foundation.**

I. INTRODUCTION

Clustering is the task of grouping a set of objects in such a way that objects in the same group are more similar in some sense or another to each other than to those in other groups. Clustering can be considered the most important unsupervised learning problem since it deals with finding a structure in a collection. The goal of clustering is to determine the intrinsic grouping in a set of unlabeled data. There is no absolute "best" criterion which would be independent of the final aim of the clustering. It is the user which must supply this criterion, in such a way that the result of the clustering will suit their needs. Clustering algorithms can be applied in many fields such as marketing, biology, libraries, weather forecasting, WWW and such. The main requirements to be satisfied by a clustering algorithm are dealing different types of attributes, discovering clusters for arbitrary shape, dealing with noise and outliers, scalability, high dimensionality and such. The most common problems faced by current clustering algorithms are that they do not address all the requirements adequately, dealing with large dimensions and number of data can be problematic because of time complexity, the effectiveness depends on the definition of "distance" which is if an obvious distance measure doesn't exist it must be defined, which is not always easy in a multi-dimensional spaces and lastly the result of a clustering algorithms can be interpreted many different ways.

.

II. LITERATURE REVIEW

The goal of this survey is to provide a comprehensive review of different clustering techniques. As clustering is done, the data are divided into groups of similar objects. Although representing data by less number of clusters achieves simplification, but the price is the loss of certain fine details. Now to proceed with our idea of optimizing and exploring the unattended part of the clustering concept, it requires a substantial amount of research on the existing clustering algorithms. The two algorithms taken under consideration is the CURE and Denclue. Both algorithm has there pros and cons and can be improved.

III. CURE

Unlike traditional clustering algorithms which normally favours clusters of only similar shape and size CURE is more robust to outliers. It can identify even non-spherical clusters and that of variable sizes. This is achieved by representing each clusters with a certain fixed number of points known as 'representative points'. Representative points are generated by selecting well scattered points from the clusters and then shrinking them to a specific amount. The reason behind the easy approach towards non-spherical cluster is more representative point per clusters and the effect of outliers is dampen by the shrinking process. Dealing with large database by CURE includes the process of random sampling and partitioning. A random sample is first drawn from the dataset and then its partitioned. After that each partition is partially clustered. Now this process is repeated through a number of iteration until the desired result is yielded. The experimental result confirms that the quality of clusters obtained from CURE is much better than those of existing algorithm.

.

IV. DENCLUE

While considering multimedia database the effectivity and efficiency of existing algorithms is limited because of the high dimensional feature vector and substantial amount of noise. Compared to the other algorithms, Denclue effectively provide a somewhat better result since it put forward a new concept of density based clustering. The idea behind Denclue is to model the overall point density as the sum of influence functions of the data points. After defining the density attractors clusters can be identified and clusters of arbitrary shapes can be described with equations based on the overall density functions. The substantial advantages of Denclue over other algorithms are it has a solid mathematical foundation with good clustering properties in data sets. The arbitrarily shaped high dimensional clusters can be described with a compact mathematical model. Finally, Denclue is a lot fasters compared to the other existing algorithms.

V. SUGGESTED APPROACH

The initial idea was to use the best features of these methods, while encompassing the advantages of either and marginalizing the disadvantages to a minimum level. The approach entailed first finding the need for such an algorithm. Detailed search lead to discovery of no significant advancements in the field of clustering algorithms for large databases. The need was realized with the problem of applying clustering for a very large data set, with more or less uniform distribution of data points. Application of DENCLUE to the same, would lead to disadvantages as stated above. Application of CURE would lead to biased clustering, while taking a longer time than DENCLUE. Thus, to summarize, a need to find an approach to handle uniformly spread data was realized.

Since the ideas we are presenting are not yet practically implemented, the theoretical details of the 3 approaches are presented below

Approach 1

Our algorithm aims to segregate data points into 3 categories initially – Regions with high density, regions with moderate density and the regions with low density via applying the basic steps of DENCLUE clustering, whilst modifying the threshold parameter ξ, into three ranges to get appropriate clusters. Repeated experimentations of the algorithm on different data would lead to proper choice of parameters for future usage.

Now, the algorithm can offer improved performance regardless of the spread of the data items:

1. For uniform data, regions with high density are limited in nature and can be identified in minimal time. Similar approach is followed for the moderately dense regions. Basic DENCLUE (with a possibly better Attractor finding approach) is applied to determine the same with a time complexity of O(n log n) This would presumably result in the reduction of data items to be clustered by a factor of m, where m<n (n being the original number of data points). Now these can be classified as the region of low density. If they are of significant size, they are then sorted via a modified version of CURE in which there's no need to choose a random sample initially as the dataset has already been minimized by a certain factor. This ensures, high quality clusters, as the data points not taken into consideration have already been clustered. These clusters so formed also result in better correlation on the basis of distance between various outliers, as they are reduced by a certain factor towards the mean, which would have already been calculated during the pre-clustering step. Thereafter one may employ CURE in its originality with a time complexity:

O ((n/m) ^2 log (n/m))

2. In case the number of data points left to be clustered has a reduction factor m<<n, then they can just be considered as outliers and are taken as one cluster, without the need of applying any clustering methodology for the same, thus achieving a time complexity of:

O (n log n)

In either case, the space requirement would remain as O (n) only via employing linear data structures. The algorithm also provides the possibility of evolving results wherein larger data leads to a relatively higher value of m, thus reducing the time complexity even more in the initial case.

Approach 2

In order to handle large datasets an effective mechanism is needed for reducing the size of the input to CURE's clustering algorithm. One approach to achieve this is via random sampling.

1. Instead of pre-clustering with all the data points, CURE begins by drawing a random sample from the database. The random samples of moderate sizes preserve information about the geometry of clusters fairly accurately, thus enabling CURE to correctly cluster the input. In particular, assuming that each cluster has a certain minimum size, we use chernoff bounds to calculate the minimum sample size for which the sample contains, with high probability, at least a fraction f of every cluster. In order to further speed up clustering, CURE first partitions the random sample and partially clusters the data points in each

partitions. After eliminating the outliers, the pre-clustered data in each partition is then clustered in a final pass to generate the final clusters. Once clustering of the random sample is completed, instead of a single centroid, multiple representative points from each cluster are used to label the remainder of the data-set. The problems with DENCLUE pre clustering are eliminated by assigning each data point to the cluster containing the closest representative point.

One can argue that the reduction in input size due to sampling has an associated cost. Since the entire data set has not being considered, information about certain clusters may be missing in the input. As a result the pre clustering step may miss out certain cluster or incorrectly identify certain clusters. Even though random sample does have the trade-off between accuracy and efficiency, the result indicates that most of the considered data points with moderate sized random samples, very good quality clusters are obtained.

The prime advantage of using CURE's sampling instead of DENCLUE's pre clustering is less time and more robust sample clusters. By using CURE's random sampling can already omit the irrelevant data points which in terms can result as a more efficient sample clusters which act as hyper cubes for further application of DENCLUE algorithm. This approach already put a lesser constraints on defining the values of Ɛ (eps) and σ(edge length for hyper cubes).

The advantage of this approach is that it will combine the core points of both the algorithm. CURE's pre clustering possess an efficient way of constructing clusters while DENCLUE can optimize it completely by removing the outliers of the already dense cluster transforming the output into clusters with greater accuracy. As DENCLUE needs a large mathematical foundation and parameters, which can be lessened to a great degree through this approach. Since the pre clusters are constructed through CURE's random sampling, it lessens the constraints on Ɛ and σ.

The only additional part required here is to calculate the density attractors from the pre clusters obtained by CURE's sampling so that the base formula of DENCLUE can be applied.

Approach 3

The next approach is suggested to get a qualitative advantage in terms of the type of clusters so formed. This approach has been suggested as next in line as this may require a larger amount of time for its completion.

Our algorithm, stretching its idea on forward from the first approach, involves a mandatory application of both DENCLUE and CURE. In the above approach, the data was clustered accordingly to handle uniformly spread data. The clusters formed from the above approach have the problem of the choice of parameters being a bottleneck, since the classification of regions depends on them. Pragmatically, this approach is being formulated to avoid the failure of the conclusion subjected to faulty choice of parameters after repetitive experimentation. This approach involves:

1. The modification in this approach involves applying DENCLUE initially till STEP 1 (Refer to DENCLUE Algorithm) i.e. the pre clustering to obtain hypercubes with dense regions within the hyperspace. The minimal bounding (hyper-) rectangle of the data set is divided into d-dimensional hypercubes, with an edge length of 2 Ϭ. Only hypercubes which actually contain data points are determined. The number of populated cubes can range from 1 to N depending on the chosen Ϭ, but does not depend on the dimensionality of the data space. Thus, the clear improvement gained over CURE is that the pre clustering step can now handle high dimensional data. The hypercubes are numbered depending on their relative position from a given origin. In this way, the populated hypercubes (containing d-dimensional data points) can be mapped to one-dimensional keys. Following this method one can get hypercubes and their neighbours with high density along with their related key values.
2. These hypercubes serve as the preclustered data for the second step of CURE wherein better data than sampling can be obtained as the possibility of rejecting significant points which include valuable clustering information is negated. Thereafter, multiple representative points instead of one local maxima is chosen from each of the clusters such that the rest of the data (i.e. the neighbouring hypercubes) can be labelled in a much more efficient manner. These dense hypercubes are now merged on the basis of these representative points, which are shrunk towards the mean of the cluster by a factor α. The distance between to clusters is then the distance between the closest pair of points. These points will not only capture the geometry and the shape of the dense clusters but also will negate the effect of outliers and also get rid of surface abnormalities. The larger the values of α, the bigger will be the compaction. Thus, theoretically, better clusters will be obtained as the effect of outliers is reduced even more.

ALGORITHM

DENCLUE (D, *h*, $\xi_{1,2,3}$,ϵ)
$A_D \longrightarrow \Phi$
$A_M \longrightarrow \Phi$
$A_L \longrightarrow \Phi$
for each x ϵ D **do**
$x^* \longleftarrow$ **FIND ATTRACTOR** (x, *h*, ϵ)
if $f(x^*) \geq \xi_1$ **then**
$A_D \longleftarrow A_D \cup \{ x^* \}$
$N(x^*) \longleftarrow N(x^*) \cup \{ x \}$
Else if $\xi_1 \geq f(x^*) \geq \xi_2$ **then**
$A_M \longleftarrow A_M \cup \{ x^* \}$
$N(x^*) \longleftarrow N(x^*) \cup \{ x \}$
Else if $f(x^*) \leq \xi_3$ **then**
$A_L \longleftarrow A_L \cup \{ x^* \}$
$N(x^*) \longleftarrow N(x^*) \cup \{ x \}$
$M_1 = \{ A \subset A : \forall x_i^*, x_i^* \epsilon A_D$, x_i^* and x_i^* are density reachable)
$M_2 = \{ A \subset A : \forall x_i^*, x_i^* \epsilon A_M$, x_i^* and x_i^* are density reachable)
$M_3 = \{ A \subset A : \forall x_i^*, x_j^* \epsilon A_L$, x_i^* and x_j^* are density reachable)

$C_1 \longrightarrow \Phi$
$C_2 \longrightarrow \Phi$
$C_3 \longrightarrow \Phi$
for each A ϵ $M_{1, 2, 3}$
for each $x^* \epsilon$ A **do** $C_{1,2,3} = C_{1,2,3} \cup N(x^*)$
F $\longleftarrow$ **FIND POINTS**(C_3) // Calculates no. of points in C_3
If F ! << D **then**
Low density clustering and outlier recognition via **CURE**
$K = K \cup C_{1,2,3}$
else
Consider points as outliers
$K = K \cup C_{1,2}$
return K
FIND ATTRACTOR (x, *h*, ϵ)
$t \leftarrow 0$
$x_0 \leftarrow x$
repeat
if *Gradient Ascent* **then**
$x_{t+1} \leftarrow x_t + \delta \cdot \nabla f(x_t)$
else

$$x_{t+1} \leftarrow \frac{\sum_{i=1}^{n} K((x_t - x_i)/h) \cdot x_t}{\sum_{i=1}^{n} K((x_t - x_i)/h)}$$

$t \leftarrow t + 1$
until $\|x_t - x_{t-1}\| \leq \epsilon$
return x_t

CLUSTERING PROCEDURE: The algorithm initiates via implementing the DENCLUE procedure as mentioned in approach 1 by taking as input the Dataset D, the influence parameter *h*, the undirected edge length ϵ and the threshold values ξ_1 , ξ_2 , ξ_3. The initial step is to compute the density attractor x^* for each point x in the dataset by starting the Find Attractor routine. If the density at x^* is above the respective minimum density threshold ξ_1 , the attractor is added to the set of attractors A_D for the high density clusters. The process is then accordingly repeated to account for the medium density clusters and the low density clusters accordingly for ξ_2 and ξ_3 . The method also maintains the set of all points N(x^*)attracted to each attractor x^* . A point x^* is called a density attractor if it is a local maximum of the probability density function f. The idea is to calculate the

density gradient, the direction of the largest increase in the density, and to move in the direction of the gradient in small steps, until one reaches a local maximum. The process further extends by applying CURE to the low density regions to account for several representative points thus reducing the effect of outliers. This removes the possibility of any low quality clusters. It works on the principle of checking whether the number of points in the cluster are significant or not. In the second step, DENCLUE finds all the maximal subsets of attractors $A \subseteq A$, such that any pair of attractors in A is density-reachable from each other further, no more attractors can be added to A without compromising this property. The set of all these maximal subsets M defines the density-based clusters. Each final cluster C comprises all points $x \in D$ that are density attracted to some attractor in A, obtained as the union of all the $N(x^*)$ sets for each $x^* \in A$. The FIND ATTRACTOR process takes into account the gradient ascent process where a point x is density attracted to another point $x*$, if the function at x converges to x^*. That is, there exists a sequence of points $x = x_0 \rightarrow x_1 \rightarrow ... \rightarrow x_m$, such that $|| x_m - x* || \leq \epsilon$, and each intermediate point is obtained after a small move in the direction of the gradient vector

$$x_{t+1} \leftarrow x_t + \delta \cdot \nabla f(x_t)$$

where $\delta > 0$ is the step size.

The hill climbing process is further optimized by speeding up the iteration via :

$$x_{t+1} \leftarrow \frac{\sum_{i=1}^{n} K((x_t - x_i)/h) \cdot x_t}{\sum_{i=1}^{n} K((x_t - x_i)/h)}$$

Where t is used to specify the current iteration. For even faster iteration, the points can be lined up using a K-d tree such that the nearest neighbors are easily obtained without much hassle. As mentioned earlier, the approach speeds up for low dimensional data, however the effectiveness decreases with higher number of dimensions as the loop time increases plus all the points then appear to be close to X_t which leads to poor cluster quality.

Complexity : For each point $x \in D$, finding the density attractor takes O(nt) time, where t is the maximum number of hill climbing iterations. This is because each iteration takes O(n) time for computing the sum of the influence function over all the points $x_i \in D$. The total cost to compute density attractors is therefore $O(n^2t)$. It is assumed that for reasonable values of h and ξ, there are only a few density attractors, i.e., $|A| = m << n$. The cost of finding the maximal reachable subsets of attractors is $O(m^2)$, and the final clusters can be obtained in O(n) time. When the dimensionality is small, the use of a spatial index can reduce the complexity of finding all the attractors to O((n log n)t) . When the pre clustering as per CURE is applied then the complexity further decreases to:

O ((n/m)^2 log (n/m))

VI. CONCLUSION

In this work, clustering using density based clustering and hierarchical clustering was implemented. It was observed that the concept of neighborhood and representative points played a key role in determining the complexity. The approach so suggested is undergoing constant improvement which would be the original purpose of this paper to invoke research at a higher platform. Furthermore, the concept of incremental growth also plays a major role in this context and is undergoing further study.

VII. ACKNOWLEDGMENT

Many individual have been instrumental in the successful implementation of this paper. Special thanks goes to our guide Mr. S. Karthik for his support along with the software department, SRM University.What seemed like a theoretical idea , but still they supported us with all the necessary resources to proceed further with the idea. The approaches suggested are however in their originality an optimization, so the authors would also be forever indebted to the original creators of CURE and DENCLUE for their creativity and innovation.

REFERENCES

[1].An Efficient Approach to Clustering in Large Multimedia Databases With Noise Alexander Hinneburg, Daniel A. Keim Institute of Computer Science, University of Halle, Germany Fhinneburg, keimg@informatik.uni-halle.de
[2].CURE: An efficient algorithm for large database, www.cs.bu.edu/fac/gkollios/ada05/LectNotes/guha98cure.pdf
[3].Comparisions between clustering algorithm, iajit.org/PDF/vol.5,no.3/15-191.pdf
[4]. K means clustering, http://www.onmyphd.com/?p=k-means.clustering