Efficient Incremental Density Based Algorithm using Boltzmann Learning Technique for Large Data Sets

Lovepreet Singh[1] Anshu Sharma[2] Sarabjit Kaur[3]
[2,3]Assistant Professor
[1,2,3]Department of Computer Science and Engineering
[1,2,3]CTITR, Jalandhar, India

Abstract— In dynamic information environment, such as web the amount of information is rapidly increasing. Thus it will be need of time that we step towards incremental clustering algorithm rather than traditional algorithm. In this paper, an enhanced version of incremental density based and competent incremental density based clustering algorithm have been introduced. This paper reveals a good clustering method should allow a significant density variation within the cluster because, if we go for homogeneous clustering, a large number of smaller unimportant clusters may be generated.

Key words: Asymmetric Clustering, Classification

I. INTRODUCTION

Clustering means putting objects having similar properties into one group and objects having dissimilar properties into another [1]. For example, object having values above threshold values can be placed in one cluster and values below into another cluster. For example in e-commerce it is used to find the relationship b/w their customer [2]. In search engine, clustering is used to find similar objects. while in networking, it is used to find analyzing the traffic and classification of collision in the network[4]. Other things, is used to detect outlier and detecting the arbitrary shapes [7]. Traditional algorithm requires the static dataset before running the algorithm. However in online platform the time factor is essential, which is not feasible in traditional algorithm.

In such algorithms, objects are processed one at a time and incrementally assigned to their prospective clusters while they progress with minimized overload. Experimental results show that the INC dbscan algorithm speeds up the incremental clustering process with a factor up to 3.2 compared to traditional density based algorithm.

A. Boltzman Machine

Invented by Geoff Hinton restricted Boltzmann is algorithm is one of the best algorithm use for dimension reduction, classification, regression. RBM is two layer neutral network In rbm there are two layers which are interconnected by two here one is the visible layer and other one hidden layer both are interconnected to each other but in this the main focus is on the forward and backward passes between hidden and visible layer. In the very fiest phase, the activations of hidden layer no. 1 become the input in a backward pass. These are multiplied by the same weights, one per internodes edge, just like x in weight-adjusted on the forward pass. The sum of these results products is added to a visible-layer bias at each visible node, and the output of those operations is a reconstruction.

activation f((weight w * input x) + bias b) = output

II. RELATED WORK

Clustering is an unsupervised technique of data mining. Existing clustering algorithm can be classified into hierarchal and partitioning method on the basis of their characterization. Partitioning algorithm also called as centroid based algorithms such as PAM[11], BIRCH[12], CLARANS[13] are simple and more optimal towards local points, however these algorithms have problem of pre-defining the number of clusters. Hierarchal based are CURE [15], CHAMELEON[16] are suffer from two main problems, firstly they can't undo whatever have done before, secondly the complexity will be high during complex data set.

In this paper, an incremental density based algorithm is introduced for dynamic environment in which updating and deletion of the cluster taken place during the runtime. The proposed algorithm enhances the clustering process incrementally portioning the dataset to reduce the search space of the neighborhood rather than whole dataset. There are the two main parameters of this algo are Eps and M inputs[].

- Definition1. The neighborhood (Eps) of an object p is donated by $N_{Eps}(p)$, is defined by $N_{Eps}(p) = \{q|dis(p, q)$ 6 Eps$\}$.
- Definition2. An object p is directly reachable to the object q. $N_{Eps}(q)$ and $|N_{Eps}(q)|$ P Minpts (i.e. q is a core object).
- Definition3. An object p is density-reachable from an object q if it follow the chain of objects $p_1, p_2, \ldots, p_n$ such that p_{i+1} is directly density reachable from p_i and $p_1 = p$ and $p_n = q$.

Whenever there is updation and deletion of the cluster taken place at that of time there will be change in only neighborhood (Eps) in one cluster. There are the following conditions on which updation and deletion of cluster points are depend.

A. Updation of Cluster Points

Here p named as cluster, and q consider as new data-point in data-set.

1) If cluster p is empty, then q consider as a noise object.
2) If cluster p contain core object but q does not possess the same properties of p, then create a new cluster [14].
3) If cluster p contain core object, and also matches the same properties of cluster p then fall into p cluster
4) if cluster p contain core object, that are member of several clusters, then new q will merges all these clusters into one.

B. Deletion of Cluster Points

1) If cluster p is empty then new data point just removed.

2) If cluster p is directly density reachable to cluster q then deletion of q makes it as noise.
3) If the objects in cluster p are not directly density-reachable from each other, then these objects belong to one cluster named c, before the deletion of q, so a check should be performed whether these objects are density connected by other objects in the former cluster c. Depending on such density connections, the cluster c may be split or not

Proposed algorithm: Initially the algorithm requires k-partitions of the datasets. After that it consider the K objects to be centroid of k-partitions are long as the distance between them are larger than neighborhood. After subsequently, to reduce the scalability the algorithm partition the data-set [18].

Algorithm 1 introduces the detail of the proposed algorithm.

1) M←List of objects that may change their centriods
2) D← Most dense regions in the dataset
3) For each point p(i) in P do
4) C←nearest centriod ()
5) Function nearest centriod ()
6) initpopulation P
7) evaluate P ;
8) Network ConstructNetworkLayers()
9) InitializeWeights(Network, testcases)
   For (i=0;i=P;i++)
10) SelectInputPattern(Inputfaultvalues)
11) ForwardPropagate(p)
    BackwardPropagateError(P)
    UpdateWeights(P)
    End
    Return (P)
12) C←P;
13) M←update centriod
14) End
15) For each r to M
16) For each ri in M do
17) c <-rinew_centroid
18) Co<-riold_centroid
19) Apply incDbscanDel to remove ri from co
20) Apply incDbscanAdd to insert ri to cn
21) Add updated dense regions to D
22) end for
23) For each di in D do
24) For each dj in D and i – j do
25) If inter_connectivty (di,dj) > a merge
26) merge(di,dj)
27) end if
28) end for
29) end for

III. PROPOSED METHODOLOGY

Density based clustering is applied on the arbitrary shapes in spatial clustering. In Density based clustering there are three parameters Eps known as radius, Minpoints requires minimum no of points required to form cluster and distance formula. These Parameters are play a vital role in the better performance against other algorithms. The Proposed Competent Density based clustering algorithm works on distance calculation. Traditionally all algorithms use the eucledian distance formula.

$$\text{Dist}_{xy} = \sqrt{\sum_{k=1}^{m}\left(X_{ik} - X_{jk}\right)^2} \qquad (1)$$

But somehow in complex nature of dataset of dbscan distance formula also consider some noise points a points of datasets.

To accomplish the need and resolve the problem of outlier points and performance, A proposed algorithm is discovered whom works on the boltzman learning technique. Using this technique a optimal level of distance is reached and also there is significant removal of noise in the final output.

Output: create/update clusters after insertions of P

1) The main step is partioning of the data-set, this step is so crucial that it provides the stability to the algorithm. The optimal portioning done on the basis of these algorithms [20]. The dynamic nature of cluster makes changes of old objects that leads to change in the positioning of the objects toward centroid.
2) Incremental DBSCAN algorithm is used to update dense region. Given new object p, the insertion module of dbscan find dense region at nearest partition so object p can join it. For old objects witch change their portion delete by using the delete module and inset module to their new dense region(line 8-14)
3) To form the final cluster the last step is to merge the dense region. The final number of cluster here formed may be equal to the dense region or one less them. To find interconnectivity closely connected cluster there is pre-defined threshold on the basis of that the new cluster is formed.
4) The final stage is to remove the noise from the cluster, if the data point is not connected with any core object of any cluster formed then it consider as the noise object [15].

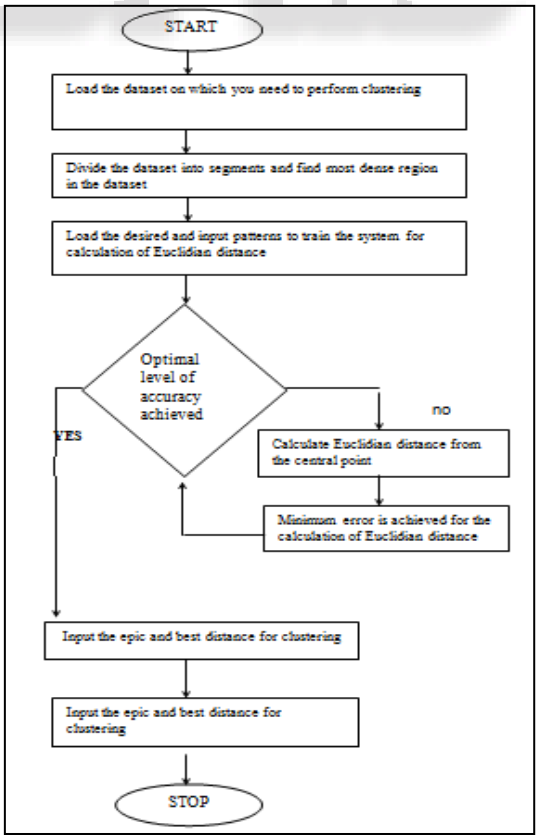

Fig. 1: Proposed Methodology

IV. RESULTS AND DISCUSSION

The incremental density based clustering algorithm speed-up clustering process by partitioning of the data-set rather to scan the whole dataset this all is by using divide and conquers. Following the dense region at each partition the algorithm uses interconnectivity merge dense tocombine the different types of clusters. The value of eps is 1.2986.

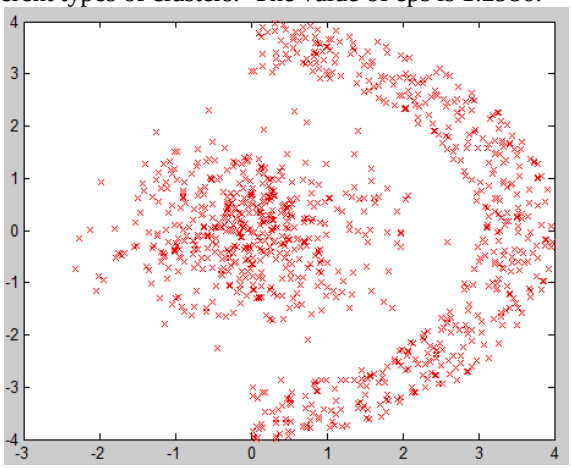

Fig. 2: Incremental Density Based Clustering Algorithm

The Proposed Competent Incremental Density Based Clustering algorithm works same like the Incremental Dbscan. The difference comes while calculating the distance formula because eucledian distance also consider outlier points as the data points which enhances the noise of the cluster. The quality and accuracy of the cluster also decreases.

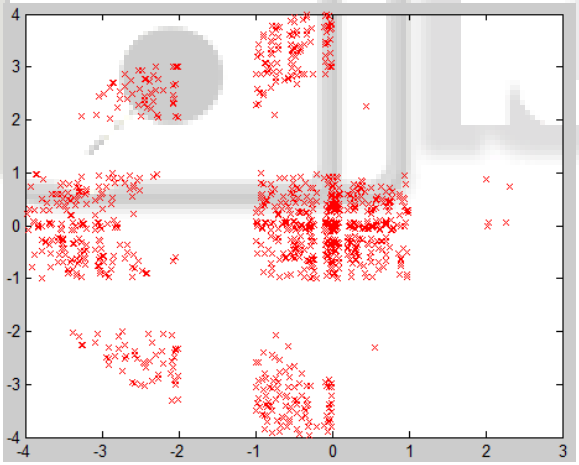

Fig. 3: Competent Incremental Density Based Clustering Algorithm

From the above fig 2 it describes the interconnectivity between sub parts of the clusters, by using their difference one can easily find relationship b/w them. Here we are calculating the best distance formula which help to enhance the quality and accuracy of the clusters.

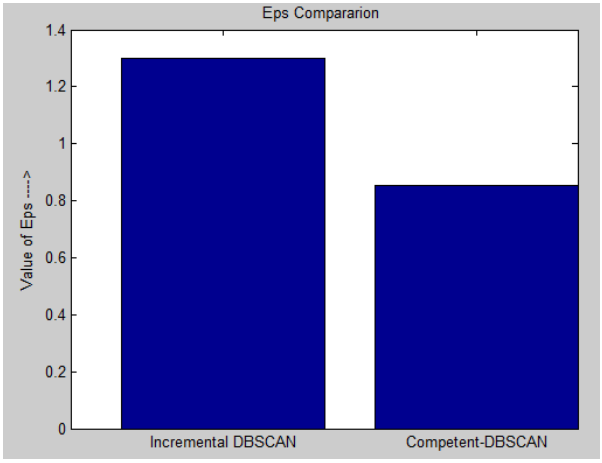

Fig. 4: Comparision b/w the results of eps values of Incremental DBSCAN and Competent Dbscan

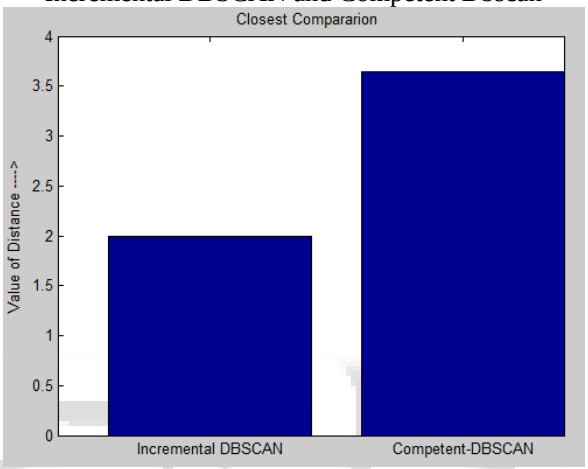

Fig. 5: Comparison b/w the results of distance values of Incremental DBSCAN and Competent Dbscan

V. CONCLUSION AND FUTURE SCOPE

In this paper a competent incremental density based clustering algorithm is proposed that is enhancement of Incremental Density based clustering algorithm. The results are compared with existing of Incremental Density based clustering algorithm. After comparing the results of both the algorithm it is observed that the results of proposed algorithm are much better and showing superiority over the existingalgorithm. In future the proposed algorithm can be used in various clustering algorithm because by using the proposed distance calculation scenario in which optimal level of best distance calculation is reached. So there is reduction of the noise points and find the relationship of inter-cluster points which is very difficult in the traditional distance formulas.

REFERENCES

[1] A. Nagpal, A. Jatain, D. Gaur, Review based on data clustering algorithms, in: IEEE Conference on Information & Communication Technologies (ICT), April 2013, pp. 298–303.

[2] W. Yu, G. Qiang, L. Xiao-Li, A kernel aggregate clustering approach for mixed data set and its application in customer segmentation, in: International Conference on Management Science and Engineering ICMSE, Oct 2006, pp. 121–124.

[3] Z. Nafar, A. Golshani, Data mining methods for protein– protein interactions, in: Canadian Conference on Electrical and Computer Engineering, CCECE, May 2006, pp. 991–994.
[4] Ahmad M. Bakr, Noha A. Yousri, Mohamed A. Ismail, Efficient incremental phrase-based document clustering, in: International Conference on Pattern Recognition ICPR, Nov 2012, pp. 517–520.
[5] S. Nithyakalyani, S.S. Kumar, Data aggregation in wirelesssensor network using node clustering algorithms a comparativestudy, in: IEEE Conference on Information &CommunicationTechnologies (ICT), April 2013, pp. 508–513.
[6] S. Hyuk Cha, Comprehensive survey on distance/similarity measures between probability density functions, Int. J. Math. Models Methods Appl. Sci. 1 (2007) 300–307.
[7] C. Bahm, K. Haegler, N.S Maller, C. Plant, CoCo: coding cost for parameter-free outlier detection, in: The 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, June 2009, pp. 149–158.
[8] D. Wang, S. Zhu, T. Li, Y. Chi, Y. Gong, Integrating clustering and multi document summarization to improve document understanding, in: The 17th ACM CIKM Conference on Information and Knowledge Management, Oct 2008.
[9] H.-P. Kriegel, M. Pfeifle, Effective and efficient distributed model-based clustering, in: Proceedings of the 5th International Conference on Data Mining (ICDM'05), 2005, pp. 285, 265.
[10] K.M. Hammouda, M.S. Kamel, Efficient phrase-based document indexing for web document clustering, in: IEEE TransKnowledge and Data Eng., vol. 16, no. 10, Oct. 2004, pp. 1279–1296.
[11] Zhe Zhang, Junxi Zhang, HuifengXue, Improved K-means clustering algorithm, in: Congress on Image and Signal Processing CISP, vol. 5, May 2008, pp. 169–172.
[12] L. Li, J. You, G. Han, H. Chen, Double partition around medoids based cluster ensemble, in: International Conference on Machine Learning and Cybernetics, vol. 4, July 2012, pp. 1390–1394.
[13] H. Du, Y. Li, An improved BIRCH clustering algorithm and application in thermal power, in: International Conference on Web Information Systems and Mining (WISM), vol. 1, Oct 2010, pp. 53–56.
[14] R.T. Ng, J. Han, CLARANS: a method for clustering objects for spatial data mining, IEEE Trans. Knowl. Data Eng. 14 (5) (2002) 1003–1016.
[15] S. Guha, R. Rastogi, K. Shim, CURE: an efficient clustering algorithm for large databases, in: Proceedings of the ACMSIGMOD International Conference Management of Data (SIGMOD'98), Oct 1998, pp. 73–84.
[16] G. Karypis, H. Eui-Hong, V. Kuma, Chameleon: hierarchical clustering using dynamic modeling, Computer 32 (8) (1999) 68–75.
[17] M. Ester, H. Kriegel, J. Sander, X. Xu, A density-basedalgorithm for discovering clusters in large spatial databaseswith noise, in: Proc. 2nd International Conference onKnowledge Discovery and Data Mining, Oct 1996, pp. 226–231.
[18] Z. Wang, Y. Hao, Z. Xiong, F. Sun, SNN clustering kernel technique for content-based scene matching, in: 7th IEEE International Conference on Cybernetic Intelligent Systems, Sept 2008, pp. 1–6.
[19] E. Achtert, C. Bhm, Aˆ H. Kriegel, P. KrAˇ ger, I. Maller- Gorman, A. Zimek, Detection and visualization of subspace cluster hierarchies, in: Advances in Databases: Concepts, Systems and Applications, Lecture Notes in Computer Science, 2007, pp. 152–163
[20] M. Ester, H.P. Kriegel, J. Sander, M. Wimmer, X. Xu, Incremental clustering for mining in a data warehousing environment, in: Proceedings of the 24th VLDB Conference,Institute for Computer Science, University of Munich,Germany, New York, USA, Aug 1998.

