ISSN:0975-9646

Optimization of DBSCALE Clustering Algorithm for Weather Forecasting

Archana Tomar[1], Neetesh Gupta[2], Amit Sinhal[3]

[1]*I.T. Department TIT Bhopal,*
[2]*I.T. Department TITS Bhopal,*
[3]*C.S.E. Department TIT'C Bhopal*

***Abstract*— Cluster analysis in data processing could be a main application of business. This investigation describes to gift DBSCALE algorithmic rule that extends enlargement seed choice into a DBSCAN algorithm rule. And also describes the density primarily based cluster conception and also describes its hierarchical extra space OPTICS has been planeed recently, and one among inthe main triumphant approaches to cluster. Aim of this analysis work is to manoeuvre on the advances cluster. During this work the planned procedure focuses on decrease the quantity of seeds points and additionally reduce the execution time cost of looking out neighbourhood information. And hierarchical cluster procedure are often helpful to those attention-grabbing subspaces so as to calculate a latitude for north and south cities and additional calculate line of longitude of various cities and also clustered them.**

***Keywords*— Data mining, Data Clustering Analysis, Density Based Clustering, Optics Algorithm, DBSCAN, IDBSCAN, KIDBSAN, DBSCALE.**

I. INTRODUCTION

In this work can cares with data mining: extracting helpful insights from massive and detailed collections of data. With the magnified potentialities in trendy society for companies and institutions to assemble data cheaply and with efficiency, this subject has become of increasing importance. This interest has impressed a speedily maturing research field with developments each on a theoretical, also as on a practical level with the supply of a range of business tools. Unfortunately, the widespread application of this technology has been restricted by a crucial assumption in thought data mining approaches. This assumption all information resides, or are often created to reside, in an exceedingly single table prevents the utilization of those data processing tools in sure vital domains, or needs extended massaging and fixing of the data as a pre-processing step. This limitation has spawned a comparatively recent interest in richer data processing paradigms that do enable structured data as against the normal flat representation.

Over the last decade, we have seen the emergence of Data Mining techniques that cater to the analysis of structured data. These techniques are generally upgrades from documented and accepted data processing techniques for tabular knowledge, and target focus on the richer representational setting. within these techniques, that we are going to put together refer to as Structured data mining techniques, and also identify a number of paradigms or 'traditions', each of which is inspired by an existing and well known choice for representing and manipulating structured data. For example, Graph Mining deals with data stored as graphs, whereas Inductive Logic Programming builds on techniques from the logic programming field.

This work specifically focuses on a convention that revolves around relational database theory: optimization of DBSCALE clustering algorithmic program for weather forecasting.

II. REVIEW

A. Optimization Problem

Optimization is that the act of getting the simplest result below given circumstances. In design, construction, and maintenance of any engineering system, engineers have to be compelled to take several technological and social controls, call at many stages. The last word goal of all such decisions is either to reduce the trouble needed or to maximise the required profit. Since the trouble needed or the profit desired in any sensible scenario will be expressed as a operate of certain

Decision variables, improvement will be outlined because the method of finding the conditions that provide the utmost or minimum price of a operate.

Optimization, in engineering style, may be a mathematical tool that works iteratively on solutions such objectives like value /performance /efficiency etc. area unit improved.

B. DBSCAN

DBSCAN-DBSCAN, planned by ester et al. in 1996 [8], was the first clustering algorithm to employ density as a condition.

DBSCAN: Density Based Spatial Clustering of Applications with Noise:

In this section, we present the algorithm DBSCAN (Density Based Spatial Clustering of Applications with Noise) which is designed to discover the clusters and the noise in a spatial database we would have to know the appropriate parameters ε and MinPts of each cluster and at least one point from the respective cluster. Then, we could retrieve all points that are density-reachable from the given point using the correct parameters. But there is no easy way to get this information in advance for all clusters of the database. However, there is a simple and effective heuristic

to determine the parameters Eps and MinPts of the "thinnest", i.e. least dense, cluster in the database. Therefore, DBSCAN uses global values for Eps and MinPts, i.e. the same values for all clusters. The density parameters of the "thinnest" cluster are good candidates for these global parameter values specifying the lowest density which is not considered to be noise.

C. DBSCAN Algorithm:

Density-Based Spatial Clustering and Application with Noise (DBSCAN) was a clustering algorithm based on density. It did clustering through growing high density area, and it can find any shape of clustering (Rong et al., 2004) [8]. The idea of it was:

1. ε-neighbor: the neighbors in ε semi diameter of an object
2. Kernel object: certain number (MinP) of neighbors in ε semi diameter
3. To a object set D, if object p is the ε-neighbor of q, and q is kernel object, then p can get "direct density reachable" from q.
4. To a ε, p can get "direct density reachable" from q; D contains Minp objects; if a series object p1,p2 ,...,pn ,p1 =q,pn =p, then pi-1 can get "direct density reachable" from pi , pi $\in$ D, $1 \leq i \geq n$.
5. To ε and MinP, if there exist a object o (o $\in$ D), p and q can get "direct density reachable" from o, p and q are density connected.

Advantages

1. DBSCAN does not require you to know the number of clusters in the data a priori, as opposed to k-means [10].
2. DBSCAN can find arbitrarily shaped clusters. It can even find clusters completely surrounded by (but not connected to) a different cluster. Due to the MinPts parameter, the so called single link effect (different clusters being connected by a thin line of points) is reduced
3. DBSCAN has a notion of noise.
4. DBSCAN requires just two parameters and is mostly insensitive to the ordering of the points in the database.

Disadvantages:

1. DBSCAN can only result in a good clustering as good as its distance measure is in the function getNeighbors(P,epsilon). The most common distance metric used is the euclidean distance measure. Especially for high-dimensional data, this distance metric can be rendered almost useless.
2. DBSCAN does not respond well to data sets with varying densities (called hierarchical data sets)

D. IDBSCAN Algorithm :

IDBSCAN is a density-based data clustering scheme developed by Borah et al. in 2004 [11]. This method applies a Marked Boundary Object to determine the data point of an expansion seed when searching for neighborhood to add in expansion seeds. Assuming that the core point is P(O,O), the eight marked objects may be defined as: A(0, ε), B(ε /√ 2, ε /√ 2), C(ε,0), D(ε /√ 2 ,– ε /√ 2}), E(0,- ε), F(-ε /√ 2 ,- ε /√ 2), G(-ε, 0), H(- ε /√ 2 , ε /√ 2)

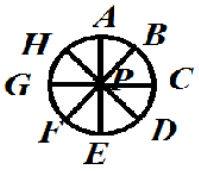

Figure: eight marked boundary object of IDBSCAN

E. KIDBSCAN Algorithm:

KIDBSCAN is a density-based clustering method presented by Tsai and Liu in 2006[19]. They searched for marked boundary objects with IDBSCAN, and found that inputting data sequentially from low-density database causes remnant seed searching, resulting in poor expansion results.

To decrease the number of sample instances, KIDBSCAN performs expansion by inputting elite points. It has three parameters, elite point, radius and MinPts. The execution steps are as follows.

(1)Adopt K-means algorithm to find the K numbers of the centroid within the database, then find the K data points closest to these centroid and define them as elite points, because K-means can discover these elite points quickly.
(2) Move the K elite points to the very front of the database.
(3) Execute the IDBSCAN algorithm. Experimental results prove that KIDBSCAN performs data clustering quickly.

F. K-MEAN Algorithm

The naive k-means algorithm partitions the dataset into 'k' subsets such that all records, from now on referred to as points, in a given subset "belong" to the same center. Also the points in a given subset are closer to that center than to any other center [11].

The algorithm keeps track of the centroids of the subsets, and proceeds in simple iterations. The initial partitioning is randomly generated, that is, we randomly initialize the centroids to some points in the region of the space. In each iteration step, a new set of centroids is generated using the existing set of centroids following two very simple steps. Let us denote the set of centroids after the ith iteration by C(i). The following operations are performed in the steps:

(i) Partition the points based on the centroids C(i), that is, find the centroids to which each of the points in the dataset belongs. The points are partitioned based on the Euclidean distance from the centroids.

(ii) Set a new centroid c(i+1) C (i+1) to be the mean of all the points that are closest to c(i) C the new location of the centroid in a particular partition is referred to as the new location of the old centroid.

The algorithm is said to have converged when recomputing the partitions does not result in a change in the partitioning. In the terminology that we are using, the algorithm has converged completely when C(i) and C(i – 1) are identical. For configurations where no point is equidistant to more than one center, the above convergence condition can always be reached.

This convergence property along with its simplicity adds to the attractiveness of the k-means algorithm. The k-means needs to perform a large number of "nearest-neighbour" queries for the points in the dataset. If the data is 'd' dimensional and there are 'N' points in the dataset, the cost of a single iteration is O(kdN). As one would have to run several iterations, it is generally not feasible to run the naïve k-means algorithm for large number of points.

Sometimes the convergence of the centroids (i.e. C(i) and C(i+1) being identical) takes several iterations. Also in the last several iterations, the centroids move very little. As running the expensive iterations so many more times might not be efficient, we need a measure of convergence of the centroids so that we stop the iterations when the convergence criteria are met. Distortion is the most widely accepted measure.

Procedure:

Problem: Cluster the following eight points (with (x, y) representing locations) into three clusters A1(2, 10) A2(2, 5) A3(8, 4) A4(5, 8) A5(7, 5) A6(6, 4) A7(1, 2) A8(4, 9). Initial cluster centers are: A1(2, 10), A4(5, 8) and A7(1, 2). The distance function between two points a=(x1, y1) and b=(x2, y2) is defined as: $\rho(a, b) = |x2 – x1| + |y2 – y1|$.

Use k-means algorithm to find the three cluster centers after the second iteration.

Iteration 1

Table 2.1 show results of k-mean cluster

		(2, 10)	(5, 8)	(1, 2)	
A1	**(2, 10)**	**0**	**5**	**9**	**1**
A2	**(2, 5)**	**5**	**6**	**4**	**3**
A3	**(8, 4)**	**12**	**7**	**9**	**2**
A4	**(5, 8)**	**5**	**0**	**10**	**2**
A5	**(7, 5)**	**10**	**5**	**9**	**2**
A6	**(6, 4)**	**10**	**5**	**7**	**2**
A7	**(1, 2)**	**9**	**10**	**0**	**3**
A8	**(4, 9)**	**3**	**2**	**10**	**2**

Cluster 1	Cluster 2	Cluster 3
(2, 10)	(8, 4)	(2, 5)
	(5, 8)	(1, 2)
	(7, 5)	(6, 4)
	(4, 9)	

Next, we need to re-compute the new cluster centers (means). We do so, by taking the mean of all points in each cluster.

For Cluster 1, we only have one point A1 (2, 10), which was the old mean, so the cluster center remains the same.

For Cluster 2, we have ((8+5+7+6+4)/5, (4+8+5+4+9)/5) = (6, 6)

For Cluster 3, we have ((2+1)/2, (5+2)/2) = (1.5, 3.5)

III. PROPOSED METHODOLOGY AND TECHNIQUES

A. *Introduction Of Methodology*

Geo Cluster project primarily developed in yii framework that is developed on PHP artificial language. From here the info are going to be fetched by the EgeoNames api functions.

After searching all the records on weather, fetched by the EgeoNames, we will apply DBSCALE clustering algorithm on this data in this algorithm requires n bytes of memory in the entire database.

The procedure of searching for neighborhood data is very time consuming in the DBSCAN algorithms. So, reduce the time consumed, this dissertation work represents "Optimize DBSCALE Clustering algorithm for weather forecasting ". In this work reduces the number of searches for neighbors and reduce the number of expansion seed point.

B. *Procedure for optimizing any Problem Using DBSCALE*

Search all the records on weather like Latitude, Longitude,Time Zone, sunset time, sunrise time etc.

Apply clustering algorithm on this record.

Steps of clustering algorithm

1. Initialization: - Initialize all parameters, and define a new Cluster ID. We will set generator counter is set to G=1.
2. Evaluation: - Begin scanning all data points within the entire database. For data points belonging to the ClusterID Of those unclassified data, implement the Expand Cluster processing procedure. The database is the set of data points; the Point represents the core point; the Cluster ID denotes the current cluster ID;e indicates the radius, and MinPts represents the minimum number of included points.
3. Replacement:- If the data point returned by the expansion procedure function is a Noise data point, then go directly to Step2, until the Datasets database has been fully scanned. If an expansion data point is returned, then update the new Cluster ID, and alter the index array of unclassified data, and then go to Step 2.
4. Termination: - End the algorithm when all data points have been processed.

Procedure for solving Problem using clustering technique

The implementation steps for the Expand Cluster processing procedure are as follows.

Search for neighborhood data within the range of radius e in the unclassified cluster index. If the number of neighborhood data is less than MinPts, then leave the procedure, and return the core point as the noise data point. Otherwise, go to Step 2

Set the core point as the current ClusterID.

If the seed data points is empty then end the expansion processing procedure, otherwise go to Step 4.

Search for the marking boundary point within neighborhood data, and add in the expansion seeds.

Set all unclassified data points and neighborhood data points that are noise data as the current ClusterID.

Extract the first seed from the expansion seeds; define it as the core point, and then delete it.

In the unclassified data index, search for neighborhood data within the range of radius e of the core point. If the number of neighborhood data is greater than Min Pts, then go to Step 3.

IV. RESULTS & DISCUSSION

A. *Introduction*

This chapter discusses the implementation of Optimization of DBSCALE Clustering Algorithm which implementations Weather Forecasting data to find the location of various cities with latitude and longitude in world. The performance of the proposed algorithm is very satisfactory compared with other techniques such as DBSCAN, IDBSCAN, and KIDBSCAN. The method is implemented on system. It is observed that this optimization technique gives better results.

Test System I :

The Table 5.1 presents the results for different cities.

S. No.	Location	Latitude	Longitude	Sunrise Time	Sunset Time
1	Algiers	36.7525	3.04197	2013-11-16 07:26:00	2013-11-16 17:38:00
2	Republic of Burundi	-3.5	30	2013-11-16 05:36:00	2013-11-16 17:53:00
3	Delhi	28.6538	77.229	2013-11-16 06:44:00	2013-11-16 17:27:00
4	Kolkata	22.5626	88.363	2013-11-16 05:49:00	2013-11-16 16:52:00
5	Mumbai	19.0728	72.8826	2013-11-16 06:46:00	2013-11-16 18:00:00

B. *Experimental Result*

Geo Cluster project basically developed in yii framework which is developed on PHP programming language. In the given image below show you a form which is for searching data related to city name. From here the data will be fetched by the EgeoNames api functions.

After search on city name "Delhi" the data will be shows like below screen where the location, latitude and longitude will be fetched by the EgeoNames search function by passing the city name and country name after it we pass the latitude and longitude value in EgeoNames's time zone function and collect the data of Time Zone, Current Time, Sunrise Time and Sunset Time. This latitude and longitude value will be used for creating a map of that city by Gmap3 function. By using latitude and longitude calculate the bounding box value and pass to the EgeoNames's weather function for getting Temperature and Humidity values.

All these values are stored in the database and after that we retrieve last 10 details and put it into the query High charts graph.

And then the clustered data will be generated by the clustering algorithm which is generated in Python language and it retrieves data from database directly. Python code executed in PHP by shell_exec command. And the clustered data will be displayed by the High Charts graph.

And In the Export Details there is two options first for download data in pdf form which is generated by the yii pdf extension and another for CSV format. and In the Last Search Data panel shows old 15 location search details.

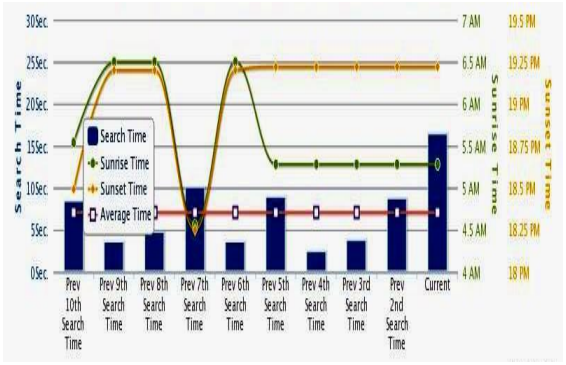

Figure: Graph show last 10 execution result

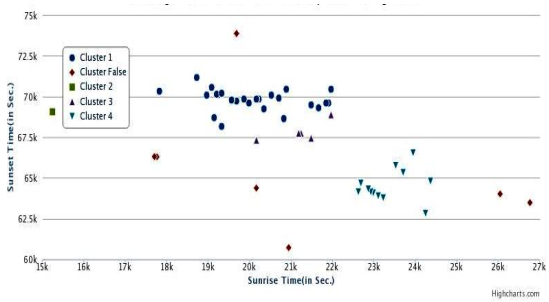

Figure: Final clustering Result for all clusters by DBSCALE Algorithm

--Total execution time for DBSCALE Clustering algorithm for weather forecasting : 0.263

--Total Execution Time for DBSCAN Clustering: 0.327

V. CONCLUSIONS AND FUTURE SCOPE

This dissertation introduces an Optimization of DBSCALE Clustering Algorithm for Weather Forecasting approach for the problem related to data mining.

This algorithm provides efficient working of data clustering with expansion of seed point efficiently. The working principal of this algorithm reduces eight Marked Boundary Objects, which resultant into coverage increment. The proposed algorithm performs excellently for arbitrary shapes and if the no of data points increases the computational time does not increase for large data sets there is no limitation by memory. The proposed algorithm "Optimization of DBSCALE Clustering Algorithm for Weather Forecasting" results has a lower execution time cost than DBSCAN, IDBSCAN and KIDBSCAN clustering algorithms.

The cluster technique can be implemented for solving real life, multi-objective problems where the objectives are conflicting in nature. The Technique approach may be extended to provide solution for larger systems and losses may be considered.

And also a lot of scope for the proposed DBSCALE clustering algorithm in different application areas such as medical image segmentation and medical data mining and many more.

REFERENCES:

[1] Prof. Aashish H Kacha, "Security and privacy challenges in Data Mining", Global Research Analysis, Volume 2, Issue 8, Aug 2013.

[2] J. Han and M. Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, 2001.

[3] Vilalta R., Stepinski T. and Achari M. "An Efficient Approach to External Cluster Assessment with an Application to Martian Topography", Technical Report, No. Uh-Cs-05-08, Department Of Computer Science, University Of Houston (2005).

[4] J. Handl, J. Knowles, and D. B. Kell, "Computational Cluster Validation in Post-Genomic Data Analysis", Journal of Bioinformatics Volume 21(15), 2005.

[5] Hans-Peter Kriegel, Stefan Brecheisen, Eshref Januzaj, Peer Kröger, and Martin Pfeifle, "Visual Mining Of Cluster Hierarchies", 3rd Int. Workshop On Visual Data Mining, 2003.

[6] Hans-Peter Kriegel, Peer Kroger, Zahi Mashael, Martin Pfeifle, Marco Potke And Thomas Seidl, "Effective Similarity Search On Voxelized Cad Objects", Proceedings Of The Eighth International Conference On Database Systems For Advanced Applications (Dasfaa'03).

[7] Inokuchi, A., Washio, T. And Motoda H. "An Apriori-Based Algorithm For Mining Frequent Substructures From Graph Data", Published In 2000.

[8] Martin Ester, Hans-Peter Kriegel, Jiirg Sander, Xiaowei X, "A Density-Based Algorithm For Discovering Clusters In Large Spatial Databases With Noise", Published In 2nd International Conference On Knowledge Discovery And Data Mining, 1996.

[9] Leandro Nunes De Castro,Fernando José Von Zuben, " Artificial Immune Systems: Part I – Basic Theory And Applications", Tr – Dca 01/99, December, 1999.

[10] K. Mumtaz And Dr. K. Duraiswamy, R., "An Analysis On Density Based Clustering Of Multi Dimensional Spatial Data", Indian Journal Of Computer Science And Engineering, Vol 1, 2010.

[11] K. Mumtaz And Dr. K. Duraiswamy, "A Novel Density Based Improved K-Means Clustering Algorithm – Dbkmeans", Ijcse International Journal On Computer Science , Vol. 02, No. 02, 2010.

[12] Neelamadhab Padhy , Rasmita Panigrahi, "Multi Relational Data Mining Approaches: A Data Mining Technique", International Journal Of Computer Applications, Volume 57– No.17, November 2012.

[13] Miyahara, T., Shoudai, T., Uchida, T., Kuboyama, T., Takahashi, K., Ueda, H.Discovering, "New Knowledge From Graph Data Using Inductive Logic Programming, In Proceedings Of Ilp '99, Lnai 1634, 1999.

[14] Titik Khawa Abdul Rahman, Saiful Izwan Suliman And Ismail Musirin, "Artificial Immune-Based Optimization Technique For Solving Economic Dispatch In Power System", Springer-Verlag Berlin Heidelberg, 2006.

[15] Pankaj Saxena , Vineeta, Dr. B. R. Ambedkar , Sushma Lehri, "Evolving Efficient Clustering Patterns In Liver Patient Data Through Data Mining Techniques", International Journal Of Computer Applications, Volume 66, March 2013.

[16] Zwe-Lee Gaing , Kao-Yuan, "Particle Swarm Optimization To Solving The Economic Dispatch Considering The Generator Constraints", Power Systems, Ieee Transactions On Volume:18, Issue:3, 2003.

[17] Dimitrls Bertsimas and John Tsitsiklis, "Simulated Annealing" In Stastical Science, Vol. 8, 1993.

[18] Tippayachai, J.,Ongsakul, W., "Parallel Micro Genetic Algorithm For Constrained Economic Dispatch" , Power Systems, Ieee Transactions On Volume:17 , Issue: 3, Aug 2002.

[19] Tsai, C.F., Liu, C.W., "Kidbscan: A New Efficient Data Clustering Algorithm For Data Mining In Large Databases" Published In Lecture Notes In Artificial Intelligence, Vol. 4029, 2006.

[20] Tsai, C.F., Yen, C.C., "Angel: A New Effective And Efficient Hybrid Clustering Technique For Large Databases" Published In Lecture Notes In Artificial Intelligence, Vol. 4426, 2007.