International Journal of Artificial Intelligence and Applications for Smart Devices
Vol.2 , No.1 (2014), pp.1-10
http://dx.doi.org/10.14257/ijaiasd.2014.2.1.01

Improved Multi Threshold Birch Clustering Algorithm

NidalIsmael[1], Mahmoud Alzaalan[2] and WesamAshour[3]

[1, 2, 3]*Computer Engineering, Islamic University, Gaza, Gaza Strip, Palestine*
[1]*eng.nismail@gmail.com*, [2]*mzaalan@gmail.com*, [3]*washour@iugaza.edu.ps*

Abstract

BIRCH algorithm is a clustering algorithm suitable for very large data sets. In the algorithm, a CF-tree is built whose all entries in each leaf node must satisfy a uniform threshold T, and the CF-tree is rebuilt at each stage by different threshold. But using a single threshold cause many shortcomings in the birch algorithm, in this paper to propose a solution to this shortcoming by using multiple thresholds instead of a single threshold.

Keywords: *BIRCH algorithm clustering threshold heterogeneous attributes data mining*

1. Introduction

The clustering process usually can divide into the following steps: first we need to represents the objects by an appropriate form by identifying the most effective subset of the original features to use in clustering and transformations of the input features to produce new salient features; then we define suitable similarity function, The most commonly used measure of similarity is the Euclidean distance or its square, Other distance measures are also available as city-block or Manhattan, Using of different distance measures may lead to different clustering results; after that we need to choose clustering algorithm, There are many clustering algorithms available in the clustering word based on different standard methods, The most widely used standard methods are the hierarchical clustering [2], partitioning clustering [3], hybrid method [4], incremental or batch methods, monothetic vs. polythetic methods [5], crisp and fuzzy clustering [6], and finally run the algorithm and get the results.

Many clustering algorithms were developed in the last decades that work in different standard methods, In Partition-based algorithms, cluster similarity is measured in regard to the mean value of the objects in a cluster, center of gravity, (K-Means [7]) or each cluster is represented by one of the cluster objects located near its center (K-Medoid [8]). Hierarchical algorithms such as BIRCH [9] and CURE [10] produce a set of nested clusters organized as a hierarchical tree. Grid-based algorithms such as STING [11], CLIQUE [12] and Wave Cluster [13] are based on multi-level grid structure on which all clustering operations are performed. In Model-based algorithms (COB-WEB [14], *etc*.,), a model is hypothesized for each of the clusters to find a model that best fits all other clusters. The Density-based notion is a common approach for clustering which is based on the idea that objects which form a dense region should be grouped together into one cluster. Algorithms such as DBSCAN [15], DENCLUE [16], CURD [17] and OPTICS [18], search for regions of high density in a feature space that are separated by regions of lower density. Most of the clustering methods require setting of user specified parameters or prior knowledge to produce their best results.

Determining parameters are hard task, but have a significant influence on clustering results. Furthermore, for many real-data sets there is no global parameter setting that describes the intrinsic clustering structure accurately.

ISSN: 2288-6710 IJAIASD

In many clustering problems we need to deal with very large and complex data sets (gigabytes or even terabytes). The data sets may contain millions of objects described by tens, hundreds of attributes or variables, a little number of the clustering algorithm have a good efficiency according to huge data sets, The BIRCH algorithm is a good robust solution in the case of huge data sets.

It generates a hierarchical partitioning of the data set and ensures that the maximum distance between two elements within a cluster is less than the given single threshold. The efficiency and the accuracy of the birth algorithm depend mainly on this threshold value, using a single threshold cause many shortcomings in the birch algorithm and in this paper we try to overcome these shortcomings by using multiple threshold birch algorithm.

The rest of the paper is organized as follows. Section 2 surveys related work and summarizes BIRCH'S algorithm, Section 3 introduce the Shortcomings of single threshold birch algorithm, Section 4 describes our Improved multi threshold Birch Algorithm .Section 5 describes datasets and experiments, respectively.

2. Related Work

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) algorithm [9] is an integrated hierarchical clustering algorithm. It uses the clustering features (Clustering Feature, CF) and cluster feature tree (CF Tree) two concepts for the general cluster description. Clustering feature tree outlines the clustering of useful information, and space is much smaller than the meta-data collection can be stored in memory, which can improve the algorithm in clustering large data sets on the speed and scalability and is very suitable for handling discrete and continuous attribute data clustering problem.

In the BIRCH tree a node is called a Clustering Feature. It is a small representation of an underlying cluster of one or many points. BIRCH builds on the idea that points that are close enough should always be considered as a group. Clustering Features provide this level of abstraction.

Clustering Features are stored as a vector of three values: CF = (N;LS; SS). The linear sum (LS), the square sum (SS), and the number of points it encloses (N). All of these metricscan be calculated using only basic math:

$$\overrightarrow{LS} = \sum_{i=1}^{N} \overrightarrow{X_i} \quad (1)$$

$$SS = \sum_{i=1}^{N} \vec{X}_i^2 \quad (2)$$

If divided by the number of points in the cluster the linear sum marks the centroid of the cluster. As the formulas suggest both of these values can be computed iteratively. Any Clustering Feature in the tree can be calculated by adding its child Clustering Features:

$$CF_1 + CF_2 = (N_1 + N_2, \overrightarrow{LS_1} + \overrightarrow{LS_{12}}, SS_1 + SS_2) \quad (3)$$

A CF tree is a height balanced tree that has two parameters namely, a branching factor, B, and threshold, T. The representation of a non-leaf node can be stated as {CFi, childi}, where ,i = 1,2, …., B,

Childi : A pointer to its ith child node.

CFi: CF of the sub cluster represented by the ith child.

The non-leaf node is provides a representation for a cluster and the contents of the node represents all of the sub clusters. In the same manner a leaf-node's contents represents all of its sub clusters and has to confirm to a threshold value for T.

The BIRCH clustering algorithm is implemented in four phases. In phase1, the initial CF is built from the database based on the branching factor B and the threshold value T.

Phase2 is an optional phase in which the initial CF tree would be reduced in size to obtain a smaller CF tree. Global clustering of the data points is performed in phase3 from either the initial CF tree or the smaller tree of phase2.

As has been shown in the evaluation good clusters can be obtained from phase3 of the algorithm. If it is required to improve the quality of the clusters, phase4 of the algorithm would be needed in the clustering process. The execution of Phase1 of BIRCH begins with a threshold value T.

The procedure reads the entire set of data points in this phase, selects the data points based on a distance function. The selected data points are stored in the nodes of the CF tree. The data points that are closely spaced are considered to be clusters and are thus selected. The data points that are widely placed are considered to be outliers and thus are discarded from clustering. In this clustering process, if the threshold limit is exceeded before the complete scan of the database, the value is increased and a much smaller tree with all the chosen data points is built.

An optimum value for threshold T is necessary in order to get good quality clusters from the algorithm. If it is required to fine tune the quality of the clusters, further scans of the database is recommended through phase4 of the algorithm. The worst case time complexity of the algorithm is O(n). The time needed for the execution of the algorithm varies linearly to the dataset size.

There are many enhancements was proposed in the birch algorithm , A new improved BIRCH hierarchical clustering algorithm has been presented in [19] which introduced tuple ID propagation and shared nearest neighbor density to make up the shortcomings of traditional BIRCH algorithm. In [20] the authors try to made some improvements on BIRCH algorithm by changing the CF structure so that heterogeneous attributes could be manipulated and suggest a heuristic method of getting initial threshold and increasing threshold of second stage of the algorithm, A different way was proposed in [21] that presents some optimization algorithms to optimize the key parameters (like branching factor, quality threshold and selection of the separator line) of the BIRCH clustering algorithm, But the previous enhancements was still using a single static threshold value that increases only when memory available to clustering becomes full, finally a dynamic approach was proposed in [22] to establishing threshold value to the central point where its data formed a cluster dynamically through the experiment.

3. Shortcomings of Single Threshold Birch Algorithm

The threshold value is the most important parameter of the BIRCH algorithm, and it is the most effective factor of the efficiency and accuracy results.

Test results of clustering on the base of BIRCH algorithm revealed that the time requirement of the procedure considerably depends on parameters of the threshold value and the maximal branching factor in the algorithm. if the threshold value parameter is decreased from its optimal value then the number of sets resulted by BIRCH algorithm is increased exponentially.

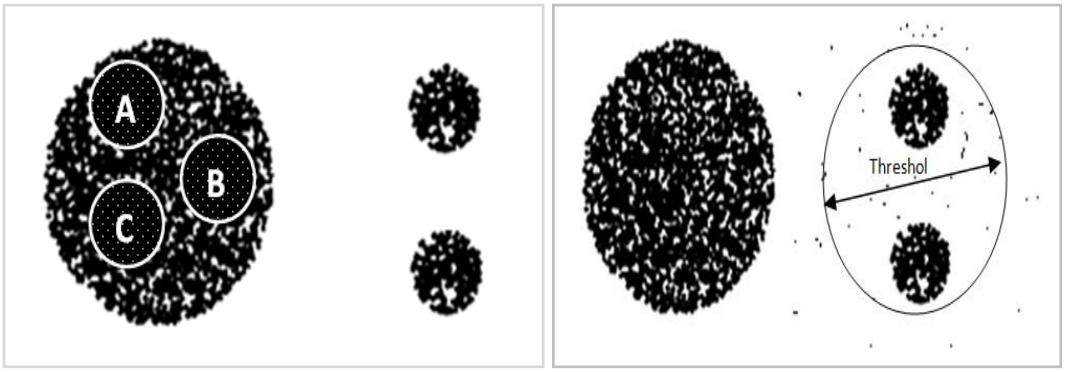
The exponential increase in the number of sets results in an exponential increase also in the cost of the algorithm based on the pre-clustering of BIRCH algorithm. If the threshold value

parameter is larger than the optimal value, then the number of points put into sets is increased which require a continuously increasing extra cost while the leaf nodes representing sets are being clustered.[21], Also beginning with a good initial value for threshold would save about 10% time.[9].

The accuracy shortcomings of the single threshold approach can appeared clearly when clusters have different sizes as shown in Figure[1], When the threshold is small the big size clusters will be divided to many clusters according to the threshold value, as seen in Figure1 (a) the big cluster divided into many clusters A,B,C…*etc.*, On the other hand when the threshold value is increased the small clusters can be merged into a single cluster and absorb the noisy data points around it, Figure1 (b) shows that the tow small clusters was merged in one cluster in addition to many noise point due to using unsuitable large threshold .

In the most particular situations the size and density of the clusters will be different and that lead to conclude there is no an optimal threshold value to use for all CF entries in bulling the Birch CF tree.

If we go back to the basic idea of the Birch algorithm we find that it always trying to maximize RAM usage. The algorithm starts with maximum precision at Threshold = 0 and as the CF tree grows larger than the available memory it iteratively tries to find suitable cluster sizes.



(a)(b)

Figure 1. Problem of Single Threshold

Threshold has to be increased to be larger than the smallest distance between two entries in the current tree. This will cause at least these two entries to be merged into a coarser one, it reduces the amount of clusters produced, and clusters will grow larger in diameter.

In the other side if we look to the modern computers we see that it can has many hundreds of RAM and that mean the single threshold witch increase only when the memory is full may not increase (or increase few times) according to the big ram available and size of dataset, the number of CF entry will be very large and every CF entry will contain a very small number of data points and that will lead to a weak clustering.

4. Improved Birch Algorithm

The new proposed enhanced birch algorithm is based on the fact that every Clustering Feature entry that used in Clustering Feature Tree is a small representation of an underlying cluster of one or many points. Unfortunately in most particular situations the sizes of these clusters are not equal, So there is no an optimal threshold is suitable to use in building the

whole CF tree and its CF entries, using a single threshold in building the CF tree – as in original birch algorithm – will cause many shortcomings as described in the previous section.

To solve this problem and overcome the previous shortcomings we present an enhanced CF tree that use multiple different thresholds where every threshold belongs to a specific leaf CF entry, In other words the number of thresholds that used in the CF tree will be equal to the number of the CF entries in that tree and these threshold will not be equal and will be dynamically changed during the clustering operation, This approach will lead to modify the original leaf CF entry structure and original insertion algorithm behavior as described below.

4.1. Modified leaf CF Entry

In original birch algorithm a Clustering Feature (CF) entry is a triple summarizing the information that we maintain about a sub cluster of data points, as described in previous sections the structure CF entry is described by the following formula, CF = {N, LS, SS}, In the modified leaf CF entry (MLCF) we add a fourth value to represent the threshold value of the leaf CF entry, MLCF entry is described by the following formula.

MLCF = {N, LS, SS, T}

Where:

N: is the number of points in the data set.

LS: is the linear sum of points in the data set.

SS: is the square sum of points in the data set.

T: is the threshold value of the leaf CF entry.

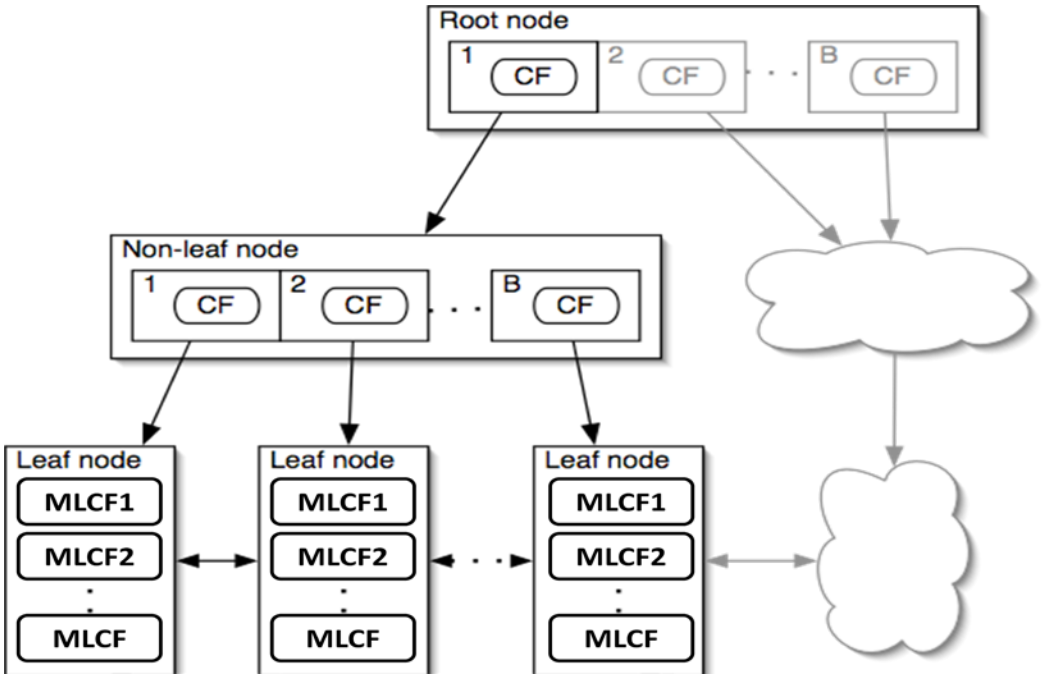


Figure 2. The Structure of Modified CF Tree

The structure of the modified CF tree will be as shown in Figure 2 and will contain two types of CF entries, normal CF entries for the root and non-leaf nodes and MLCF entries in the leaf nodes. In the insertion algorithm every MLCF entry will use different threshold value to compare with maximum diameter of an entry and decide whether the data point will be absorbed or rejected, In addition to that the MLCF entry will not be static and will increase dynamically as described in the improved algorithm below.

4.2. Modified Algorithm

The proposed improved algorithm is shown in Figure 3, and the Algorithm described as follows:

1. Before scanning any data point from database we must initialize the initial CF tree threshold, this threshold will be used as initial threshold value for every new created MLCF entry and will not be changed during the clustering process.
2. For each new scan of data point, Start from the root, recursively descend the CF-tree by choosing the closest child node, Upon reaching a leaf node, find the closest leaf entry and then test whether it can absorb the new point without violating the local MLCF threshold condition, If so, update the CF entry to reflect the insertion of the new data point and complete normally as origin birch algorithm.
3. If the chosen closest MLCF entry can't absorb the new data point (local threshold condition violated) then:
    a. Try to increase the local threshold by multiplying with threshold Modifying Factor (MF), this Modifying Factor value will depend on the value of MLCF threshold, if the threshold is small the Modifying Factor will be relatively large and if the threshold is large the Modifying Factor will be relatively small.
    b. If the chosen MLCF entry can absorb the new data point with the modified threshold update the CF entry to reflect the insertion of the new data point, and update the threshold to the new threshold value. Find the nearest neighbor entry to the current MLCF entry and test whether it can absorb the nearest neighbor entry centroid with the new threshold, if so merge the two MFLC entries and update the path to the root.
    c. If the chosen MLCF entry can't absorb the new data point with the modified threshold we keep the old threshold and add a new MLCF entry and complete normally as origin birch algorithm.

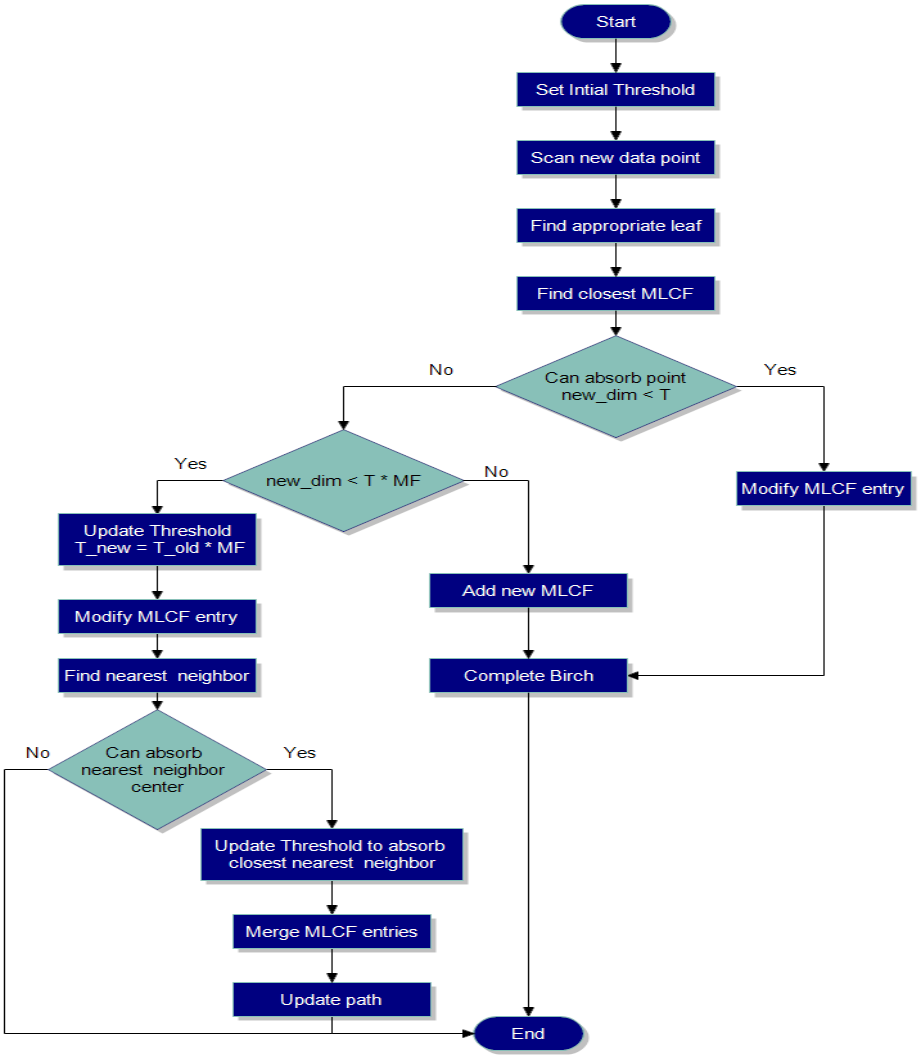


Figure 3. Improved Birch Algorithm

5. Performance evaluation

5.1. Data Sets

To evaluate our improved birch algorithm we implement the basic birch algorithm and the multi threshold birch algorithm, we try to use big data set to be compatible with the birch environment, and the experiments depend on two different types of data sets (Artificial and Real datasets). Values of the generated artificial dataset are used to assess the level of the algorithm success. The real data sets used in the experiment are: Statlog (Shuttle) Data Set and Abalone Data Set.

Statlog (Shuttle) Data Set is a real data set that contains about 58,000 samples divided into 7 clusters, every sample is described by 9 real numbers attributes, Approximately 80% of the data belongs to class 1. Therefore the default accuracy is about 80% which mean there is a big dominant cluster and that is suitable for our work. Abalone Data Set Predicting the age of abalone from physical measurements. Contains 4177 sample with 8 real integer attributes.

5.2. Experiments Results

The experimental result for the basic and improved birch algorithms is shown in table (1) , and table(2) and aim to show the main characteristics of the produced CF tree created by the two algorithms, we fix the branching factor to the value 7 which is suitable for our experiments to eliminate its effect in building the CF tree, Table (1) shows the Total CF-Nodes, Total CF-Entries, and Total CF-Leaf_Entries of the CF tree built by the basic birch algorithm and improved multi threshold birch algorithm for Statlog (Shuttle) Data Set using initial threshold equal to 2 and 3 respectively, the value of the chosen thresholds depends on the natural of the data set samples, As shown in table(1) the size of the CF tree built by the improved multi threshold birch is about 60% less than the CF tree built by basic birch algorithm, when we set the initial threshold to 3 the size of the CF tree built by the improved multi threshold birch is about 25% of the size of CF tree built by basic birch algorithm, the experiments shows that less than 1% of the samples was absorbed in the wrong sub cluster in the improved birch algorithm.

Table 1. CF Trees Characteristics of Shuttle Data Set

Algorithm	**Total CF-Nodes**	***Total CF-Entries***	***Total CF-Leaf Entries***
Initial threshold 2			
Birch	3	3138	10960
Improved Birch	1153	1167	4176
Initial threshold 3			
Birch	2431	2442	8726
Improved Birch	535	546	1914

Table 2. CF Trees Characteristics of Abalone Data Set

Algorithm	**Total CF-Nodes**	***Total CF-Entries***	***Total CF-Leaf Entries***
Initial threshold 0.1			
Birch	334	345	1257
Improved Birch	104	121	405
Initial threshold 0.2			
Birch	136	156	544
Improved Birch	53	62	203

Table (2) shows the characteristics of the CF tree built by the basic birch algorithm and improved multi threshold birch algorithm for Abalone Data Set using initial threshold equal to 0.1 and 0.2. When the initial threshold value is 0.1 the size of the CF tree built by the improved multi threshold birch is about 30% of the size of CF tree built by basic birch algorithm, and When the initial threshold value is 0.2 the size of the CF tree built by the improved multi threshold birch is about 25% of the size of CF tree built by basic birch algorithm.

The decreasing in the CF tree size will increase the efficiency of the birch algorithm in the different phases, the improved multi threshold birch algorithm ensure that the accuracy of the clustering process will not be negative affected by the decreased CF tree size.

6. Conclusions

Clustering is used in many fields such as data mining, knowledge discovery, statistics and machine learning. This paper presented enhancement to birch algorithm by using multiple threshold instead of the single threshold used in basic birch algorithm. Experimental results demonstrate that the medications appear to give good performance and overcome many of the shortcomings in basic birch algorithm.

References

[1] J. Han and M. Kamber,"Data Mining: concepts and techniques", Beijing: China Machine Press, **(2006)**.
[2] A. Jain, M.Murty andP. J. Flynn, "Data Clustering: A review", ACM Computing Surveys, vol. 31, no 3, **(1999)**, pp. 264-321.
[3] W. Day, "Complexity Theory: An Introduction for practitioners of classification", Clustering and Classification, World Scientific Publ., **(1992)**.
[4] M. Murty andG. Krishna, "A computationally efficient technique for data clustering", Pattern Recognition, vol. 12, **(1980)**, pp. 153-158.
[5] G. Salton, "Developments in automatic text retrieval", Science, vol. 253, **(1991)**, pp. 974-980.
[6] E. Ruspini, "A new approach to clustering", Information Control, vol. 15, **(1969)**, pp. 22-32.
[7] J. MacQueen, "Some methods for classification and analysis of multivariate observations", Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, **(1967)**, pp. 281–297.
[8] H. Vinod, "Integer programming and the theory of grouping", Journal of the American Statistical Association, vol. 64,no. 326,**(1969)**,pp. 506-519.
[9] T. Zhang, R. Ramakrishnan and M. Linvy, "BIRCH: an efficient data clustering method for very large databases",Proceeding ACM SIGMOD International Conference on Management of Data, **(1996)**, pp. 103-114.
[10] S. Guha, R. Rastogi and K. Shim, "CURE: an efficient clustering algorithms for large databases", Proceeding ACM SIGMOD International Conference on Management of Data, Seattle, WA, **(1998)**, pp. 73-84.
[11] W. Wang, J. Yang and R. Muntz, "STING: a statistical information grid approach to spatial data mining", Proceedings of 23rd International Conference on Very Large Data Bases (VLDB),**(1997)**, pp. 186-195.
[12] R. Agrawal, J. Gehrke, D. Gunopulos and P. Raghavan,"Automatic subspace clustering of high dimensional data for data mining applications",SIGMOD'98.
[13] G. Sheikholeslami, S. Chatterjee and A. Zhang, "WaveCluster: a multi-resolution clustering approach for very large spatial databases",Proceedings of International Conference on Very Large Databases (VLDB'98), New York, USA, **(1998)**, pp. 428–439.
[14] D. Fisher, "Knowledge acquisition via incremental conceptual clustering", Machine Learning,vol. 2, no. 2,**(1987)**,pp. 139-172.
[15] M. Ester and H.-P.Kriegel. "A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise",Proc of 2nd Int. Conf. on K9'96, **(1996)**, pp. 226-231.
[16] A. Hinneburg and D.A. Keim, "An efficient approach to clustering in large multimedia databases with noise", Proceedings of 4th International Conference on Knowledge Discovery and Data Mining, New York City, NY, **(1998)**, pp. 58–65.
[17] S. Ma, T.J. Wang, S.W. Tang, D.Q. Yang and J. Gao, "A new fast clustering algorithm based on reference and density", Proceedings of WAIM, Lectures Notes in Computer Science, 2762, Springer, **(2003)**, pp. 214-225.
[18] M. Ankerst, M. Breunig, H.-P.Kriegel andJ. Sander, "OPTICS: Ordering Points To Identify the Clustering Structure",SIGMOD' 99.
[19] Z. Yu-Yan, G. Jing-Feng, Z. Li-Zhen and L. Jing, "Improved BIRCH Hierarchical Clustering Algorithm", Computer Science, vol.35, no.13, pp.180-183.
[20] J. Shen-yiand L. Xia, "Improved BIRCH clustering algorithm", Journal of Computer Applications, vol.29,no. 1, **(2009)**,pp.293-296.
[21] L.Kovács and L. Bednarik, "Parameter Optimization for BIRCH Pre-Clustering Algorithm", CINTI 2011 12th IEEE International Symposium on Computational Intelligence and Informatics, Budapest, Hungary, **(2011)**November21-22.

[22] D.HaiZhou and L.YongBin, 2010 International Conference on Web Information Systems and Mining.

Authors

NidalIsmael,Received the BS degree in Computer Engineering in 2007 from the Islamic University, Palestine. He is currently working toward the MS degree at the Islamic university. His current areas of interest include image processing, and cloud computing.

Mahmoud Alzaalan, received the BS degree in Computer System Engineering in 2008 from the Alazhar University, Palestine. He is currently working toward the MS degree at the Islamic University. He has 1 publication in international journal. His current areas of interest include pattern recognition.

WesamAshour, received the engineering degree in computerscience in 2000 from Islamic University of Gaza – Gaza, Palestine, the MS degree in Multimedia Computer Systems, University of Birmingham, UK, and won the prize of the best MSc project, in 2004 and the PhD degree in Data Mining: "Local versus Global Interactions in Clustering Algorithms", in 2008 from University of the West of Scotland, UK. He was teaching assistant in Electrical and Computer Engineering department at the Islamic University of Gaza from 15/02/2001 to 15/8/2003. From 15/8/2004 to 15/8/2005 he was working as a lecturer in Electrical and Computer Engineering department at the Islamic University of Gaza. He was working as a lab demonstrator in School of Computing at the University of the West of Scotland and from 15/8/2008 till now he is working as a lecturer in Computer Engineering department at the Islamic University of Gaza, and he is a Researcher in Applied Computational Intelligence Research Unit, The University of the West of Scotland, UK since October, 2005. He was awarded The Distinguished Student Award of the Arab Fund Fellowship Program – Arab Fund for Economic and Social Development 2012. He has more than 35 published papers in international conferences and journals. His current research interests are in Data Mining, Artificial Intelligence, Neural Networks, Recognition, Image processing.

