

DECLARATION

I certify that all the material in this thesis that is not my own work has been identified, and that no material is included for which a degree has previously been conferred on me.

The contents of this thesis reflect my own personal views, and are not necessarily endorsed by the University.

Name: Hany Hanafy Mahmoud Said

Signature:

Date: 15 / 11 / 2008

We certify that we have read the present work and that, in our opinion, it is fully adequate in scope and quality as a thesis towards the partial fulfillment of the Master's Degree requirements in

Computer Engineering

From the

College of Engineering and Technology (AASTMT)

Date: 15 / 11 / 2008

Supervisor:

Name: Prof. Dr. Yasser El Sonbaty

Position: Professor – College of Computing & Information Technology – AASTMT

Signature:

Examiners:

Name: Prof. Dr. Mohamed Ismail

Position: Professor – Faculty of Engineering – Alexandria University

Signature:

Name: Prof. Dr. Amin Shokry

Position: Professor – Faculty of Engineering – Alexandria University

Signature:

ACKNOWLEDGMENTS

I owe a great gratitude to my supervisor, Prof. Dr. Yasser El Sonbaty, for his excellent advising and guidance through my thesis. I thank my mother from the bottom of my heart for her standing with me all the time. I am grateful to my father for his pushing me always to reach my goals. I also thank my colleagues in Arab Academy for their support.

ABSTRACT

Clustering is one of the important data mining techniques for extracting knowledge from datasets in various applications. Most of the clustering algorithms do not achieve the majority of the clustering requirements as scalability; discovering clusters of different shapes, dealing with noise, dealing with high dimensional data.

Density-based method is one of the most effective paradigms in cluster analysis that detects clusters with arbitrary shapes without acquire the number of clusters in advance. DBSCAN discovers the dense regions and identifies them as clusters that are separated by low density regions. Several algorithms improve DBSCAN algorithm such as fast hybrid density based algorithm “L-DBSCAN” and fast density-based clustering algorithm.

Since clustering algorithms discover clusters, which are not known a priori, the final partition of a dataset requires some sort of evaluation in most applications. *CDbw* validity index measures the average density within clusters, $Intra_dens(c)$ with the separation of clusters $Sep(c)$, where c is the number of clusters discovered in the dataset. *CDbw* increases when clusters are highly dense, well separated and the densities in the areas among them are low.

In this thesis, an enhanced density based algorithm is proposed that improves fast density based clustering algorithm. The experimental results show that the new algorithm is superior to L-DBSCAN and fast density-based clustering and DBSCAN algorithms in more specific aspects, like running time, clusters’ compactness in terms of intra-cluster density and clusters’ separation in terms of inter-cluster density.

TABLE OF CONTENTS

CHAPTER I: INTRODUCTION	1
CHAPTER II: CLUSTER ANALYSIS	8
2.1 Introduction.....	8
2.2 Clustering applications.....	9
2.3 Requirements of Clustering in Data Mining.....	10
2.4 Data Structure in the Cluster analysis.....	12
2.5 Kinds of Data Types.....	13
2.5.1 Continuous Variables.....	13
2.5.2 Binary Variables.....	15
2.5.3 Nominal Variables.....	18
2.5.4 Ordinal Variables.....	19
2.5.5 Ratio-scaled Variables.....	21
2.5.6 Variables of mixed Types.....	21
2.5.7 Vector Objects.....	23
CHAPTER III: CLUSTERING PARADIGMS	24
3.1 Introduction.....	24
3.2 A categorization of the major clustering algorithms.....	25
3.2.1 Partitioning Algorithms.....	25
1 The K-means Algorithm.....	26
2 PAM (Partitioning Around Medoids Algorithm)	27
3 CLARA (Clustering LARge Applications).....	30
4 CLARANS (Clustering Large Applications based on RANDOMized Search)	31

3.2.2 Hierarchical Algorithms	34
5 BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies)	37
6 CURE (Clustering Using REpresentatives)	39
7 CHAMELEON (Hierarchical Clustering Algorithm using Dynamic Modeling)	41
3.2.3 Grid-Based Algorithms.....	44
8 STING (STatistical INformation Grid)	45
9 CLIQUE (Clustering In Quest)	48
3.2.4 Density-Based Algorithms	50
10 DBSCAN (Density Based Spatial Clustering of Applications with Noise)	50
11 OPTICS (Ordering Points to Identify the Clustering Structure)	56
12 Efficient Density Based Clustering Algorithm.....	60
13 Improved sampling-based DBSCAN for large spatial databases.....	68
14 Fast Density-Based Clustering Algorithm for Large Databases.....	71
15 A Fast Hybrid Density Based Clustering Method.....	74

CHAPTER IV: CLUSTERING VALIDITY

ASSESSMENT	81
4.1 Introduction.....	81
4.2 Clustering Validity Methods.....	85
4.2.1 <i>SD</i> Validity Index.....	85
4.2.2 <i>S_Dbw</i> Validity Index.....	89
4.2.3 <i>CDbw</i> Validity Index.....	96

CHAPTER V: PROPOSED ALGORITHM.....	105
5.1 Introduction.....	105
5.2 The Stages of Enhanced Density Based Algorithm	106
5.2.1 Find Density Map using leader approach.....	107
5.2.2 Update Density Map.....	108
5.2.3 Obtaining Clusters using modified fast DBSCAN.....	110
5.3 Scalability of the efficient density based algorithm	113
5.4 Illustrative example	114
CHAPTER VI: EXPERIMENTAL RESULTS.....	117
6.1 Introduction.....	117
6.2 Synthetic Datasets.....	118
6.3 Real Datasets.....	125
6.3.1 Letter Recognition Dataset.....	126
6.3.2 Pen-Based Recognition of Handwritten Digits Dataset.....	129
CHAPTER VII: DISCUSSIONS, CONCLUSIONS AND FUTURE WORK	133
REFERENCES.....	137

LIST OF ABBREVIATIONS

KDD	Knowledge Discovery in Databases
PAM	Partitioning Around Medoids Algorithm
CLARA	Clustering LARge Applications
CLARANS	Clustering Large Applications based on RANdomized Search
BIRCH	Balanced Iterative Reducing and Clustering using Hierarchies
CF	Clustering Feature
CURE	Clustering Using REpresentatives
CHAMELEON	Hierarchical Clustering Algorithm using Dynamic Modeling
RI	Relative Interconnectivity
RC	Relative Closeness
STING	Statistical Information Grid
CLIQUE	Clustering In Quest
DBSCAN	Density Based Spatial Clustering of Applications with Noise
OPTICS	Ordering Points to Identify the Clustering Structure
IDBSCAN	Improved sampling-based DBSCAN for large spatial databases
MBO	Marked Boundary Objects
FDBSCAN	Fast Density-Based Clustering Algorithm for Large Databases
CDbw	Compose Density Between and Within Clusters

LIST OF FIGURES

Figure 3.1:	The generalized iterative relocation partitioning algorithm	25
Figure 3.2:	The k-means partitioning algorithm	26
Figure 3.3:	Clustering of a set of objects based on the k-means algorithm	27
Figure 3.4:	Four cases of the cost function for k-medoids clustering	29
Figure 3.5:	The PAM partitioning algorithm	30
Figure 3.6:	The CLARA clustering algorithm	31
Figure 3.7:	CLARANS searching for better solution	32
Figure 3.8:	The CLARANS clustering algorithm	33
Figure 3.9:	Determining the Maximum Number of Neighbors	34
Figure 3.10:	Agglomerative versus divisive hierarchical clustering	35
Figure 3.11:	Agglomerative versus divisive hierarchical clustering	39
Figure 3.12:	Merging two clusters in CURE algorithm	40
Figure 3.13:	Basic Steps of CURE	41
Figure 3.14:	The effect of closeness on cluster merging choices	41
Figure 3.15:	The effect of connectivity on cluster merging choices	42
Figure 3.16:	Chameleon: Hierarchical clustering based on k-nearest neighbors	43
Figure 3.17:	Steps of Grid-based Clustering Algorithms	44
Figure 3.18:	A Hierarchical Structure for Sting Clustering	45
Figure 3.19:	Different Grid Levels during Query Processing	47
Figure 3.20:	Illustration of CLIQUE definitions	48
Figure 3.21:	Identification of clusters in subspace	49
Figure 3.22:	Core points and border points	51
Figure 3.23:	Density-reachability and density connectivity	52
Figure 3.24:	DBSCAN: Core, Border, and Noise Points	52
Figure 3.25:	DBSCAN algorithm	53
Figure 3.26:	DBSCAN expand cluster algorithm	54
Figure 3.27:	Three sample databases	55
Figure 3.28:	The discovered clusters by CLARANS	55
Figure 3.29:	The discovered clusters by DBSCAN	55
Figure 3.30:	Sorted 4-dist graph	56
Figure 3.31:	Clusters wrt. Different density parameters	57
Figure 3.32:	Illustration of “nested” density-based clusters	58
Figure 3.33:	Core-distance(o), reachability-distances $r(p_1,o)$, $r(p_2,o)$ for $MinPts=4$	59
Figure 3.34:	Illustration of the cluster-ordering	60

Figure 3.35:	Stages of the enhanced DBSCAN algorithm	61
Figure 3.36:	The relative inter-connectivity between two dense regions based on border objects ..	63
Figure 3.37:	The effect of applying one global threshold in DBSCAN	64
Figure 3.38:	k-distance histogram for 100 random objects	65
Figure 3.39:	Enhanced DBSCAN clustering algorithm	65
Figure 3.40:	Synthetic datasets	66
Figure 3.41:	Speed up factor on synthetic dataset (1)	66
Figure 3.42:	Speed up factor on synthetic dataset (2)	66
Figure 3.43:	Speed up factor on synthetic dataset (3)	67
Figure 3.44:	Core object P with eight distinct MBO points	69
Figure 3.45:	Improved Sampling-Based DBSCAN algorithm	70
Figure 3.46:	Clustering of objects with intersections	72
Figure 3.47:	Fast Density-Based Clustering Algorithm	73
Figure 3.48:	Comparisons of run time of the FDBSCAN and DBSCAN algorithms for increasing database size	74
Figure 3.49:	Approximation errors	76
Figure 3.50:	Coarse-counts and fine-counts are used to reduce approximation errors	77
Figure 3.51:	Coarse-fine-leaders Algorithm	77
Figure 3.52:	L-card Algorithm	78
Figure 3.53:	Synthetic dataset	79
Figure 4.1:	Synthesis dataset	81
Figure 4.2:	The clustering scheme using K-Means	82
Figure 4.3:	Partitioning dataset in three clusters using k-means, CURE and DBSCAN	82
Figure 4.4:	The different partitions resulting from running DBSCAN with different input parameter values	83
Figure 4.5:	Quality index SD when the dataset is two dimensions and four dimensions	87
Figure 4.6:	The best clustering scheme for the iris dataset and the graph of SD versus the number of clusters	88
Figure 4.7:	The best clustering scheme for the synthetic dataset and the graph of SD versus the number of clusters	89
Figure 4.8:	The best clustering scheme for the given dataset S, partition dataset into two clusters, partition dataset into four clusters and falsely partition dataset in three clusters	92
Figure 4.9:	The synthetic dataset and S_Dbw as a function of number of clusters	93
Figure 4.10:	The synthetic dataset and S_Dbw as a function of number of clusters	93
Figure 4.11:	S_Dbw as a function of the number of clusters for a six dimensional dataset consisting of two clusters	94
Figure 4.12:	The synthetic dataset and S_Dbw as a function of number of clusters	94
Figure 4.13:	The definition of Inter-Cluster Density	98

Figure 4.14:	Sample Synthetic Datasets and CDbw as a function of number of clusters for the given dataset	100
Figure 4.15:	Sample Synthetic Datasets and CDbw as a function of number of clusters for the given dataset	101
Figure 4.16:	CDbw as a function of number of clusters for the dataset	102
Figure 4.17:	Sample Synthetic Datasets and CDbw as a function of number of clusters for the given dataset	102
Figure 5.1:	Shows outlier objects as leader and demonstrates the wrong objects association to their leader	106
Figure 5.2:	The stages of the Proposed Algorithm	107
Figure 5.3:	Synthetic dataset and the sorted K-dist graph	108
Figure 5.4:	Leader Position before and after getting medoid	109
Figure 5.5:	Synthetic Dataset, density map before and after distributing dataset on Medoids	109
Figure 5.6:	Density Connectivity, cluster i expansion and Normal Distribution Curve	110
Figure 5.7:	Global distance threshold and Dynamic distance	111
Figure 5.8:	Number of region queries processed by efficient density based algorithm	112
Figure 5.9:	Average search space required by efficient density based algorithm	113
Figure 5.10:	Synthetic dataset	114
Figure 5.11:	Density map before distributing dataset on the leaders	114
Figure 5.12:	Density map after distributing dataset on leaders	115
Figure 5.13:	The discovered clusters for the given dataset	115
Figure 5.14:	The final clustering scheme for the given dataset	115
Figure 5.15:	Enhanced Density Based Algorithm	116
Figure 6.1:	Synthetic dataset contains three clusters	118
Figure 6.2:	Speed up ratio and Validity Ratio between Proposed algorithm and DBSCAN	119
Figure 6.3:	Speed up ratio and Validity Index Ratio between Proposed algorithm and fast Density DBSCAN	119
Figure 6.4:	Speed up ratio and Validity Index Ratio between Proposed algorithm and L-DBSCAN	120
Figure 6.5:	Synthetic dataset contains three clusters with different densities	120
Figure 6.6:	Speed up ratio and Validity Ratio between Proposed algorithm and DBSCAN	121
Figure 6.7:	Speed up ratio and Validity Index Ratio between Proposed algorithm and fast Density DBSCAN	122
Figure 6.8:	Speed up ratio and Validity Index Ratio between Proposed algorithm and L-DBSCAN	122
Figure 6.9:	Synthetic dataset contains forty clusters	123
Figure 6.10:	Speed up ratio and Validity Ratio between Proposed algorithm and DBSCAN	123
Figure 6.11:	Speed up ratio and Validity Index Ratio between Proposed algorithm and fast Density DBSCAN	124

Figure 6.12:	Speed up ratio and Validity Index Ratio between Proposed algorithm and L-DBSCAN	125
Figure 6.13:	Speed up ratio and Validity Ratio between Proposed algorithm and DBSCAN	127
Figure 6.14:	Speed up ratio and Validity Index Ratio between Proposed algorithm and fast Density DBSCAN	128
Figure 6.15:	Speed up ratio and Validity Index Ratio between Proposed algorithm and L-DBSCAN	129
Figure 6.16:	Speed up ratio and Validity Ratio between Proposed algorithm and DBSCAN	130
Figure 6.17:	Speed up ratio and Validity Index Ratio between Proposed algorithm and fast Density DBSCAN	131
Figure 6.18:	Speed up ratio and Validity Index Ratio between Proposed algorithm and L-DBSCAN	132

LIST OF TABLES

Table 2.1:	A contingency table for binary variables	16
Table 2.2:	Dataset containing mostly binary variables	17
Table 2.3:	Sample dataset contains one categorical variable	18
Table 3.1:	Comparison of IDBSCAN & DBSCAN for increasing size of datasets	70
Table 3.2:	L-DBSCAN vs. DBSCAN	79
Table 4.1:	Optimal partitioning found by S_Dbw for K-mean, CURE and DBSCAN algorithms	93
Table 4.2:	Optimal partitioning found by S_Dbw for K-mean, CURE and DBSCAN algorithms	95
Table 4.3:	Optimal partitioning found by CDbw for different clustering algorithms	100
Table 4.4:	Optimal partitioning found by CDbw for different clustering algorithms	101
Table 4.5:	Optimal number of clusters proposed by validity indices compared with CDbw	103
Table 6.1:	Attribute Information of the Letter recognition dataset	126
Table 6.2:	Class distribution of the letter recognition dataset	127
Table 6.3:	Class distribution of the pen-based recognition of handwritten digits dataset	130