Clustering Mixed Data Set Using Modified MARDL Technique

Mrs. J.Jayabharathy
Senior Lecturer
Department of CSE
Pondicherry Engg., College

Dr. S. Kanmani
Professor & Head
Department OF IT
Pondicherry Engg., College

S. Pazhaniammal
Computer Science and Engineering
Pondicherry Engineering College
Puducherry

ABSTRACT

Clustering is tend to be an important issue in data mining applications. Many clustering algorithms are available to cluster datasets that contain either numeric or categorical attributes. The real life database consists of numeric, categorical and mixed type of attributes. It is an essential task to cluster these data sets to extract significant knowledge from the existing database or to obtain statistical information about the database. Clustering large database is a time consuming process. Sampling is a process of obtaining a small set of data from the large database. Applying sampling technique would not cluster all the data points. Labeling non- clustered data point is an issue in data mining process. This paper mainly focuses on clustering mixed data set using modified MARDL (MAximal Resemblance Data Labeling) technique and to allocate each unlabeled data point into the corresponding appropriate cluster based on the novel clustering representative namely, N-Nodeset Importance Representative (NNIR). Accuracy and Error rate are considered as the metrics for evaluating the performance of the existing and proposed algorithm for mixed data set. The experimental result shows that MARDL for mixed data set algorithm performs better than the existing enhanced k-means.

Keywords – Data mining, Clustering, Mixed Type Attributes, Data labeling, MARDL

I INTRODUCTION

Clustering of data is a method by which large sets of data are grouped into clusters of smaller sets of similar data based on some similarity measurement. Clustering typically groups data into sets in such a way that the intra-cluster similarity is maximized while the inter-cluster similarity is minimized. The clustering technique has been deemed as an important issue in the data mining [2], statistical pattern recognition [3], data streams and information retrieval [5] because of its usage in a wide range of applications [4]. Consider a set of data points, the goal of clustering is to partition those data points into several groups according to the predefined similarity measurement. Finding the optimal clustering result has been proved to be an NP-hard problem [6]. As the size of data grows at rapid pace, clustering a very large database inevitably incurs a time-consuming process. In order to improve the efficiency of clustering, sampling is usually used to scale down the size of the database [15].

In particular, sampling has been employed to speed up clustering algorithms. A typical approach to utilize sampling techniques on clustering is to randomly choose a small set from the original database, and then the clustering algorithm is executed on the small sampled set [1]. The clustering result, which is expected to be similar obtained from the original database, can hence be efficiently obtained. However, the problem of how to allocate the unclustered data into appropriate clusters has not been fully explored in the previous works [7]. As per the clustering concept and without loss of generality, the goal of clustering is to allocate every data point into an appropriate cluster. A partial clustering result which is obtained from the sampled database is not the expected result what we really wants.

For example, when we perform clustering for "customer" database, if we apply sampling technique a part of customer database is alone sampled. However, the other customer's datasets which are not sampled will not obtain the cluster label and, thus, do not belong to any groups. In such a case, an efficient method which is able to allocate the unclustered data into appropriate clusters is required. The capability to deal with the datasets contain both numeric and categorical attributes and it is undoubtedly important fact that the datasets with mixed types of attributes are also common in real life data mining applications. In case of numerical domain, there is a common solution to measure the similarity between an unclustered data point to allocate a cluster based on the distance between the unclustered data point and the centroid of that cluster like k-means algorithm [4]. Each unclustered data point can be allocated to the cluster with the minimal distance.

Most of the earlier works on clustering have been mainly focused on numerical data whose inherent geometric properties can be exploited to naturally define distance functions between data points. Recently, the problem of clustering categorical data draws attention of major researchers. However, the computational cost makes most of the previous algorithms unacceptable for clustering very large databases. The categorical attributes also prevalently exist in real data. In the categorical domain, the above said procedure is infeasible because finding the centroid of cluster is difficult. As a result, a mechanism named MAximal Resemblance Data Labeling (abbreviated as MARDL) is proposed to allocate each

categorical unclustered data point into the corresponding proper cluster.

Most traditional clustering algorithms are limited to handling datasets to certain limit that contain either numeric or categorical attributes. Existing clustering algorithms for datasets with mixed types of attributes contain certain drawbacks in their own ways. Many clustering algorithms have been proposed for clustering mixed data set. However, there is no appropriate algorithm for clustering large dataset. In addition there is no appropriate method to cluster unlabeled data i.e., no data labeling technique. In this paper we refer the unclustered data points as unlabeled points. Here, we propose an efficient technique named MARDL that supports both clustering large data set with mixed attributes and data labeling method.

Our contributions

The existing MARDL clustering technique for categorical data set is extended and we apply the modified MARDL technique for mixed data set.

- A cluster representative named NIR (Nodeset Importance Representative) and a generalized representative named NNIR which is extended from NIR are used in this paper. NIR and NNIR consider both intra-cluster similarity and inter-cluster similarity to represent the cluster, and give us a rough concept of the significant components of the cluster.
- We proposed modified MARDL, which is a framework for clustering large database with sampling and data labeling techniques that supports mixed dataset. The main characteristics of MARDL are: 1) *high efficiency and* 2) *retaining cluster characteristics*.

This paper is organized as follows: Section 2 gives us the detailed survey of existing clustering algorithm. Section 3 states the overview of the proposed framework MARDL for clustering mixed dataset. Section 4 shows the performance study on real data set. Section 5 concludes the paper.

II RELATED WORKS

Data labeling is used to allocate an unlabeled data point into the corresponding appropriate cluster. The technique of data labeling is available in CURE [7]. However, CURE is a special numerical clustering algorithm to find non-spherical clusters. A specific data labeling algorithm is defined to assign each unlabeled data point into the cluster which contains the representative point closest to the unlabeled data point. CURE is robust to outliers and identifies clusters with non-spherical shapes, and wide variances in size. Each cluster is represented by a fixed number of well scattered points.

To allocate each categorical unclustered data point into the corresponding proper cluster MARDL [1], mechanism is proposed. It is a framework of clustering large categorical database with sampling and data labeling techniques. MARDL is independent of clustering algorithms, and any categorical clustering algorithm can be utilized in this framework.

Clusters are represented by several representative points. ROCK [8], is an adaptation of an agglomerative hierarchical categorical clustering algorithm. This algorithm assigns data point to a separated cluster, and then merges the clusters repeatedly according to the closeness between clusters. The closeness between clusters is defined as the sum of the number of "links" between all pairs of representative points in the clusters. However, this representative utilizes several representative points and moreover it does not provide a summary of cluster, and thus cannot be efficiently used for the post processing.

For example, in the data labeling, the similarity between unclustered data points and clusters is needed to be measured. It is time consuming to measure the similarity between unclustered data points and each representative point, especially when a large amount of representative points is needed for the better representability.

Squeezer algorithm [9], produces high-quality cluster in high-dimensional categorical datasets. This algorithm has been extended for the domains with mixed numeric and categorical attributes algorithm namely dsqueezer and usm-squeezer. Since the Squeezer algorithm has been demonstrated to be very effective for clustering categorical datasets, in the dsqueezer algorithm, we adopt a simple strategy of transforming the original dataset into categorical dataset by discretizing numeric attributes. Then, the Squeezer algorithm is used to cluster the transformed dataset. For the usm-squeezer algorithm, a unified similarity measure for mixed-type attributes, in which both numeric and categorical attributes could be handled equally in the framework of Squeezer algorithm.

III PROPOSED FRAMEWORK FOR CLUSTERING USING MODIFIED MARDL

In MARDL technique unlabeled data points would be allocated into cluster via two phases, namely, the Cluster Analysis phase and Data Labeling Phase. Fig.1 shows the entire framework on clustering a large database on sampling and MARDL. This MARDL technique is independent of any clustering algorithm. The problem and several notations are defined in Section 3.1. In Section 3.2, the n-nodeset importance is defined. Section 3.3 introduces a novel cluster representative which is named as NNIR. The insignificant n-nodeset pruning strategies are presented in Section 3.4. Finally, in Section 3.5, MARDL technique for mixed data is proposed.

Cluster Analysis phase

In the cluster analysis phase, a cluster representative is generated to characterize the clustering result. In this paper, a cluster representative, named NIR is devised. NIR represents clusters by the attribute values, and the importance of an attribute value is measured by

the following two concepts: *1) the attribute value is important in the cluster when the frequency of the attribute value is high in this cluster and 2) the attribute value is important in the cluster if the attribute value appears prevalently in this cluster rather than in other clusters.* To measure the importance of attribute values, NIR considers both the intracluster similarity and the intercluster similarity to represent the cluster. Moreover, we extend NIR to represent clusters by multivariate attribute values.

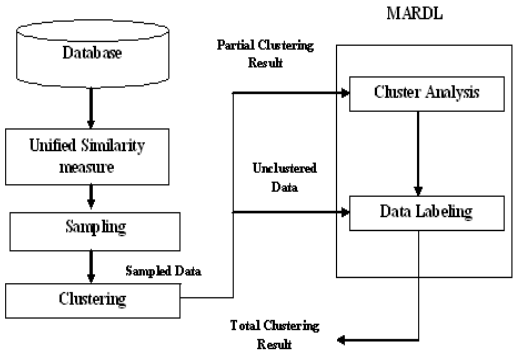

Fig. 1, The framework for sampling large database and clustering Mixed data set using MARDL

The generalized cluster representative is named NNIR (N-Nodeset Importance Representative). NNIR preserves the same concepts from NIR to measure the importance of the combination of attribute values and provides a more powerful representative than NIR where the combination of attribute values is also considered.

Data labeling phase

In the data labeling phase, each unlabeled data point is given a label of appropriate cluster according to NIR/NNIR. The similarity between the unlabeled data point and the cluster is designed based on the NIR/NNIR. Based on this similarity measurement, MARDL allocates each unlabeled data point into the cluster which possesses the maximal resemblance. Therefore, NNIR is able to be utilized in the clustering visualization, and tries to represent the clustering result in an effective way.

Unified similarity measure

Defining a unified similarity measure for mixed-type attributes, handled both numeric and categorical attributes equally. Let $A_1^n, A_2^n, \ldots A_n^p, A_{p+1}^c, \ldots A_m^c$ be a set of attributes with domains $D_1, \ldots, D_m$, assume that the first p elements are numeric attributes and the rest are categorical attributes without loss of generality.

3.1 Problem formulation

The problem of allocating unlabeled data points into appropriate clusters is formulated as follows: Suppose that a prior clustering result C= $\{c_1,c_{2,\ldots,}c_k\}$ is given, where c_i , $1 \le i \le k$, is the *i*th cluster. Cluster c_i, with a label c_i^*, is composed of m_i data points, i.e., $c_i = \{p_{(i,1)},p_{(i,2)},\ldots,p_{(i,mi)}\}$, where each data point is a vector of q attribute values, i.e., $p_{(i,j)} = \{p^1_{(i,j)},p^2_{(i,j),\ldots}p^q_{(i,j)}\}$. Let A= $\{A_1,A_2,\ldots A_q\}$, where A_a is the *a*th categorical attribute, $1 \le a \le q$. In addition, the unlabeled data set U = $\{p_{(U,1),\ldots,}p_{(U,j)}\}$ is given, where $p_{(U,j)}$ is the *j*th data point in data set U. Without loss of generality, U contains the same attribute set A. Based on the preceding, the objective of MARDL can be stated as "to decide the most appropriate cluster label c_i^* for each data point in U". Fig. 2 shows an example of this problem. There are three clusters c_1, c_2, and c_3, and the attribute set A has three attributes, A_1, A_2, and A_3. The task of data labeling is given each unlabeled data point in U the most appropriate cluster label, i.e., one of c_1^*, c_2^* or c_3^* .We first define node, n-nodeset, and independent nodesets as follows:

Definition 1 (node). *An node, d_i, is defined as attribute name with its corresponding attribute value.*

The term *node* is defined to represent attribute value and it avoids the ambiguity which might be caused by identical attribute values. For example, if there are two different attributes with the same attribute value, e.g., the age is in the range 50-59 and the weight is in the range 50-59. Here the attribute value 50-59 is confusing when we separate the attribute value from the attribute name. *Nodes* [*age* = 50-59] and [*weight* = 50-59] avoid this uncertainty. If the attribute name and the attribute value are both the same in nodes d_1 and d_2, d_1 and d_2 are said to be equal. For example, in Fig. 2 cluster c_1, [A_1 = 4] and [A_2 = a] are nodes.

Cluster C_1

A_1	A_2	A_3
4	a	c
4	b	b
5	c	c
4	a	a
4	a	c

Cluster C_2

A_1	A_2	A_3
5	c	a
4	c	a
5	c	a
5	a	b
4	b	a

Cluster C_3

A_1	A_2	A_3
4	c	c
5	c	b
4	c	b
4	b	c
5	a	a

Unlabeled dataset U

A_1	A_2	A_3
4	a	c
4	c	a
5	b	b
5	a	c
…	…	…

Fig. 2. An sample data set with three clusters and several unlabeled data points

Definition 2 (n-nodeset). *An n-nodeset, I_r^n, is defined as a set of n nodes, $\{d_1, d_2, \ldots, d_n\}$, in which every node is a member of the distinct attribute A_a.*

For example, in Fig. 2 cluster c_1, {[A_1 = 4], [A_2 = a]} is a 2-nodeset, but {[A_1 = 4],[A_1 =5]} is not a 2-

nodeset because those two nodes come from the same attribute A_1.

The importance of the combination of nodes is measured in order to represent clusters with multivariate attribute values. The properties that all of the nodes in each nodeset will be 1) frequently occur together in that cluster and 2) infrequently occur together in the other clusters if the nodeset is utilized to represent the cluster.

Definition 3 (Independent nodesets). *Two nodesets I_r^{n1} and $I_{r'}^{n2}$ in a represented cluster are said to be independent if 1) the nodeset $I_r^{n1} \cap I_{r'}^{n2}$ do not come in that cluster representative and 2) for all nodes in I_r^{n1} and $I_{r'}^{n2}$ do not come from the same attribute, i.e., for all d_t in I_r^{n1} and for all $d_{t'}$ in $I_{r'}^{n2}$ do not come from the same attribute.*

For example, suppose that two nodesets, {[A_1= 4], [A_2 = a]} and {[A_3 = c]}, are in a represented cluster, and 3-nodeset {[A_1= 4], [A_2 = a], [A_3 = c]}, does not utilize to represent that cluster. When we estimate the probability of the 3-nodeset {[A_1= 4], [A_2 = a],[A_3 = c]} in the cluster, we assume that those two nodesets, {[A_1= 4], [A_2 = a]} and {[A_3 = c]}, are independent. Therefore, the probability of the 3-nodeset in the cluster is calculated by multiplying the probabilities of those two nodesets in the cluster.

3.2 Node and N-Nodeset Importance

The basic idea of NIR is to represent a cluster as the distribution of the nodes. To measure the representability of each node in a cluster, the importance of node is evaluated based on the following two concepts:
1. The node is important in the cluster when the frequency of the node is high in this cluster.
2. The node is important in the cluster if the node appears commonly in this cluster rather than in other clusters.

A_m	The m-th attribute in the data set.
C	The clustering result.
c_i	The i-th cluster in C.
c_i^*	The label of cluster c_i.
d_t	The t-th node in an nodeset.
$f(I_{ir}^n)$	The weighting function of I_{ir}^n in calculate n-nodeset importance.
I_r^n	The n-nodeset.
I_{ir}^n	The n-nodeset which occurs in c_i.
$\lvert I_{ir}^n \rvert$	The frequency of the n-nodeset which occurs in c_i.
k	The number of clusters in C.
m_i	The number of data points in c_i.
$p(i, j)$	The j-th data point in c_i.
q	The number of data dimensionality.
U	The unlabeled data set.
$w(c_i, I_{ir}^n)$	The importance value of I_{ir}^n in c_i.
$R(x_{i,j}, x_{i,j'})$	The resemblance value between unlabel data point $x_{i,j}$ and $x_{i,j'}$.
θ	The threshold parameter in threshold pruning.

Table 1. Summary of the symbols utilized in this paper

Definition 4 (n-nodeset importance).

The importance value of the n-nodeset I_{ir}^n is calculated as follows:

$$w(C_i, I_{ir}^n) = \frac{|I_{ir}^n|}{m_i} * f(I_r^n) \quad (1)$$

$$f(I_r^n) = 1 - \frac{-1}{\log k} * \sum_{y=1}^{k} p(I_{yr}^n) \log(p(I_{yr}^n)) \quad (2)$$

$$\text{where } p(I_{yr}^n) = \frac{|I_{yr}^n|}{\sum_{z=1}^{k^t} |I_{zr}^n|} \quad (3)$$

If *n,* is the number of nodes in a nodeset, which equals to one, the 1-nodeset is able to be seen as a node. So that the definition of the node importance can be inferred from 1-nodeset importance. $w(C_i, I_{ir}^n)$ represents the importance of n-nodeset I_{ir}^n in cluster c_i with two factors, the probability and the weighting function. Based on the concepts of the n-nodeset importance, the probability of I_{ir}^n in c_i computes the frequency of I_{ir}^n cluster c_i, and the weighting function is designed to measure the distribution of the n-nodeset between clusters. The weighting function $f(I_{ir}^n)$ measures the entropy of the n-nodeset between clusters.

The importance of the n-nodeset I_{ir}^n in cluster c_i is measured by multiplying the first concept, i.e., the probability of I_{ir}^n in c_i, and the second concept, i.e., the weighting function $f(I_{ir}^n)$. Note that both the range of the probability of I_{ir}^n in c_i and the weighting function $f(I_{ir}^n)$ are [0, 1], implying that the range of the important value $w(C_i, I_{ir}^n)$ is also [0, 1].

Example 1. Consider the data set in Fig. 2. Cluster c_1 contains five data points. The 1-nodeset {[A_1=4]} occurs four times $|I^1_{1,\{[A_1=4]\}}|$ in c1, twice in c2, and three times in c3. The weight of the 1-nodeset

$$f(I^1_{\{[A_1=4]\}}) = 1 - \frac{-1}{\log 3}\left(\frac{4}{5}\log\frac{4}{5} + \frac{2}{5}\log\frac{2}{5} + \frac{3}{5}\log\frac{3}{5}\right) = 0.225$$

Therefore, the importance of the 1-nodeset {[A_1=4]} in cluster c_1 is

$$w(C_1, \{[A_1 = 4]\}) = 0.225 * \frac{4}{5} = 0.18$$

Note that in cluster c1, the 2-nodeset {[A_1 = 4], [A_2 =a]} also occurs three times. However, this nodeset does not occur in c2 and c3. Therefore, in cluster c1, the combination of these two items is more significant than 1-nodeset {[A_1=4]}. Corresponding to the n-nodeset importance,

$$w(C_1, \{[A_1 = 4], [A_2 = a]\}) = f(I^2_{\{[A_1=4],[A_2=a]\}}) * \frac{3}{5} = 1 * \frac{3}{5} = 0.43 > w(C_1, \{[A_1 = 4]\})$$

= 0.17

Although these two nodesets both occur three times in cluster c_1, the combination of nodes [A_1 =4] and [A_2 = a] provides more information on cluster c_1 than single node [A_1 = 4].

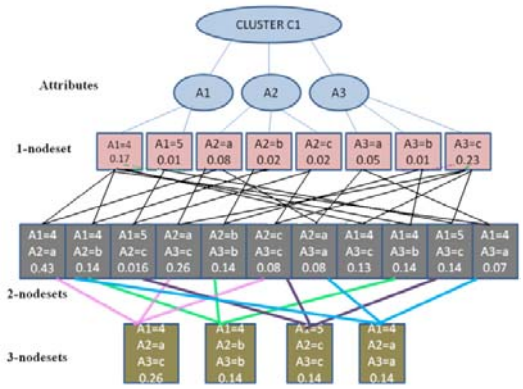

Fig. 3, An example of complete NNIR tree which represents cluster c1 in Fig. 2.

3.3 N-Nodeset Importance Representative

To represent the clustering result by the n-nodeset importance, a lattice tree structure is employed in NIR/ NNIR. Fig. 3 shows an example of the complete NNIR tree structure which represents cluster c1 in Fig. 2. The basic idea of NNIR is to represent a cluster by the NNIR tree. Each tree node in the NNIR tree records an n-nodeset and the importance value in the cluster. In a complete NNIR lattice tree, all the combinations of attribute values that occur in cluster will be found. In this case, the size of the tree seems to be extremely large, and it is very hard to generate such a huge tree by an efficient algorithm. Actually, some of the n-nodesets are infrequent in the cluster or occur in all clusters averagely. The importance of those n-nodesets is small, and representing clusters by those insignificant n-nodesets is ineffective. Therefore, to obtain an effective clustering representative, there is a need to preserve the significant n-nodesets and to delete the insignificant n-nodesets. For that purpose three greedy methods, which prune the lattice tree dynamically are applied in our method. Based on the pruning method, the significant n-nodesets without respect to the tree level n are kept in this tree.

The NNIR tree constructing and pruning algorithm is composed of three phases, which are *initialization phase, computing candidate nodeset importance and pruning phase, and generating candidate nodesets phase*. The works done in each phase are described as follows:

In the initialization phase, clustered data points are decomposed into 1- and 2-nodesets, and the frequencies of those nodesets in each cluster are recorded. For example, in Fig. 2, the data point p(1,1), which is the first row of cluster c1, it can be decomposed into three 1-nodesets {[A_1= 4]}, {[A_2 = a]},{[A_3 = c]} and three 2-nodesets ({[A_1= 4], [A_2 = a]}, {[A_2= a], [A_3 = c]}, {[A_1= 4], [A_3 = c]}). All of the 2-nodesets are treated as the initial candidate nodesets (c-nodesets) and are fed as an input for second phase. In the computing candidate nodeset importance and pruning phase, the importance of each candidate nodeset is calculated by using the n-nodeset importance formula. After that, the decision is taken whether the c-nodeset is dropped or not according to the pruning algorithm. It is important to note that in our pruning strategies, 1-nodesets will not be pruned because we consider that all the 1-nodesets which occur in the cluster provide basic representative.
The cluster analysis algorithm for mixed dataset which inputs a clustering result C and returns the NNIR tree of each cluster is shown in Algorithm 1.

Algorithm 1 Cluster Analysis(C)
Input: the clustered sample data set C
1. Read the data point $p_{(i,j)}$ from the cluster C.
2. Divide data points into 1-nodesets N^1 and 2-nodesets N^2
3. Add N^1 and N^2 into candidate nodesets CN
4. Check CN value
5. IF CN $\neq \emptyset$ then perform the following steps
6. Calculate the frequency of the n-nodesets
7. Compute the weight of the nodesets by using the importance formula
8. Perform pruning method for 2 and 3 nodesets to obtain the result.

3.4 NNIR Tree Pruning Algorithms

The main objective of the tree pruning algorithms is to delete the insignificant nodesets and to preserve the nodesets which truly possess representability to the represented cluster. In this work, we design three pruning algorithms, which are threshold, relative maximum, and hybrid pruning. The pruning algorithms are presented in the following sections.

3.4.1 Threshold Pruning

The idea of threshold pruning algorithm is *to prune a nodeset if the nodeset is insignificant in the cluster*. In the threshold pruning, a user-specified parameter θ, which set up the threshold on the value of the n-nodeset importance, is required. The parameter θ is in the range [0, 1], and the nodeset I^n_{ir} remaining criterion in the threshold pruning algorithm is shown as the following equation: "I^n_{ir} remains in NNIR tree if $w(C_i, I^n_{ir}) \geq \theta$".
For example, suppose that θ= 0.1. The 2-nodesets {[A_1 = 5], [A_2 = c]}, {[A_2 = c], [A_3 =c]}, {[A_2 = a], [A_3 =a]}, {[A_1 = 4], [A_3 =a]} in Fig. 4 are pruned.

3.4.2 Relative Maximum Pruning

In the relative maximum pruning, consider the idea that the n-nodeset which is composed of n n-1-

nodesets is a positive combination, i.e., combining n n-1-item sets provides more information for the cluster. The nodeset I_{ir}^{n} remaining criterion in the relative maximum pruning algorithm is shown as the following equation: "I_{ir}^{n} remains in NNIR tree if $\forall r', I_{ir'}^{n-1} \subset I_{ir'}^{n}, w(C_i, I_{ir}^{n}) > w(C_i, I_{ir'}^{n-1})$". In the above equation, the nodeset I_{ir}^{n} remains in the NNIR tree if the importance value of nodeset I_{ir}^{n} is larger than each importance value of $I_{ir'}^{n-1}$ which is decomposed from I_{ir}^{n}. In other words, the nodeset I_{ir}^{n} remains in the NNIR tree if $w(C_i, I_{ir}^{n})$ is the maximum value relative to all the $w(C_i, I_{ir'}^{n-1})$.

For example, the 2-nodeset {[A_1 = 4], [A_2 = a]} Fig. 4 remained because the importance value of {[A_1 = 4], [A_2 = a]} is larger than the importance value of 1-nodesets {[A_1 = 4]} and {[A_2 = a]}. However, the 2-nodeset {[A_1 = 5], [A_2 = c]} is dropped because the importance value of {[A_1 = 5], [A_2 = c]} is smaller than the importance value of {[A_1 = 5]} and{ [A_2 = c]}. This can be explained by the reason that combining items [A_1 = 5] and [A_2 = c]} does not provide more representability on cluster c_1.

3.4.3 Hybrid Pruning

It is clear that the ideas of the threshold pruning and relative maximum pruning are not conflict and can be applied simultaneously. Therefore, both criteria shown above are applied to prune the NNIR tree in the hybrid pruning algorithm. Fig. 4 shows the result of the pruned NNIR tree after applying the hybrid pruning in Fig. 3. The threshold θ is set to 0.1.

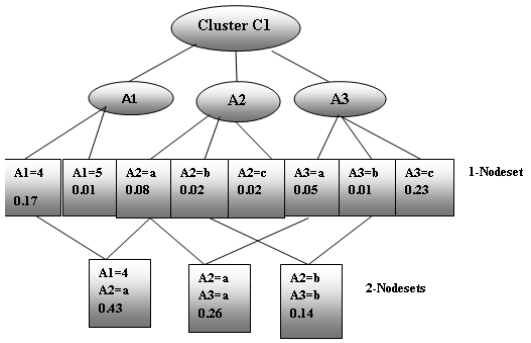

Fig. 4. The pruned NNIR tree which applies hybrid pruning to represent cluster c_1 in Fig. 2. The threshold θ is set to 0.1

3.5 Maximal Resemblance Data Labeling

The goal of MARDL is to decide the most appropriate cluster label c_i^* for the unlabeled data point. Specifically, when an unlabeled data point $p_{(U,j)}$ is given MARDL computes the similarity $S(c_i, p_{(U,j)})$ between $p_{(U,j)}$ and cluster c_i, $1 \leq i \leq n$, and finds the cluster which has Max $(S(c_i, p_{(U,j)}))$. In order to calculate the similarity between $p_{(U,j)}$ and c_i, referred to as resemblance in this paper, we define the nodeset combination.

Definition 5 (nodeset combination).

Given an unlabeled data point $p_{(U,j)}$ and an NNIR tree which represents cluster c_i , the nodeset combination is defined by a set of nodesets which are composed of $p_{(U,j)}$, and are independent to each other, and also, are able to be found in the NNIR tree.

For example, considering the first unlabeled data point in Fig. 2 and the NNIR tree in Fig. 6, three nodeset combinations composing of $p_{(U,1)}$, which are {[A_1= 4], [A_2 = a]} and {[A3= c]}, {[A_1= 4],[A_3=c]} and {[A_2= a]},and {[A_1= 4], [A_2 = a]} and {[A3= c]}, can be found in that NNIR tree. Based on those nodeset combinations, the resemblance between $p_{(U,j)}$ and c_i is defined as follows:

Definition 6 (resemblance).

Given an unlabeled data point $p_{(U,j)}$ and an NNIR tree which represents cluster c_i, the resemblance between $p_{(U,j)}$ and c_i is calculated by the following equation:

$$R(C_i, p(U,j)) = \max \prod_{u} \frac{\left|I_{ir_u}^{n_u}\right|}{m_i} * E\left(f(I_{ir_u}^{n_u})\right) \quad (4)$$

where $0 < n_u \leq n$, $n_u = n$, $\forall I_{ir_u}^{n_u}$ are independent, and $UI_{ir_u}^{n_u} = p_{(U,j)}$.

The resemblance between $p_{(U,j)}$ and c_i is measured by the n_u-nodesets combinations. The first part estimate the probability of the combination and the second part estimate the weight of the combination. Since all n_u-nodesets are independent with each other, the probability of the combination in cluster can be measured by the product of the probabilities of $I_{ir_u}^{n_u}$ in cluster c_i. In addition, the weight of the combination is estimated by the expected value of the weights of $I_{ir_u}^{n_u}$, i.e., $E(f(I_{ir_u}^{n_u}))$. Therefore, the weight of the combination is estimated by the expected value of the weight of each $I_{ir_u}^{n_u}$, which averages the contributions of each n_u-nodeset component on the second concept of nodeset importance. In addition, we may find many nodeset combinations from $p_{(U,j)}$ in the NNIR tree. The combination of $I_{ir_u}^{n_u}$ which contains the maximum resemblance is selected to be the resemblance, $R(C_i, p(u,j))$, between $p_{(U,j)}$ and c_i.

Example 2. Consider the data set in Fig. 2. We want to calculate the resemblance between the first unlabeled data point {[A_1= 4], [A_2 = a], [A_3= c]} and cluster c_1. The pruned NNIR tree of cluster c_1 is shown in Fig. 6. Three combinations of independent nodesets which is decomposed from the unlabeled data point are able to be found in Fig. 6. One combination is the 2-nodeset {[A_1= 4], [A_2 = a]} and the 1-nodeset {[A_3= c]}, another combination is the 2-nodeset {[A_1= 4], [A_3 = c]} and the 1-nodeset {[A_2 = a]}, and the other one includes three 1-nodeset, which are {[A_1= 4]}, {[A_2 = a]}, and {[A_3= c]}. The resemblance computed by the first combination is

$$\left(\frac{\{[A_1 = 4], [A_2 = a]\}|}{m_1} * \frac{\{[A_3 = c]\}|}{m_1}\right)$$

$$* \left(\frac{2}{3}(f\{[A_1 = 4], [A_2 = a]\}) + \frac{1}{3}(f\{[A_3 = c]\})\right)$$

$$= \left(\frac{3}{5} * \frac{3}{5}\right) * \left(\frac{2}{3} * 1 + \frac{1}{3} * 0.387\right) = 0.286$$

Based on the same measurement, the resemblance computed by the second combination is 0.217, and the resemblance computed by the third combination is 0.053. Therefore, the resemblance between the unlabeled data point $p_{(U,1)}$and cluster c_1, $R(C_i, p(U, j))$= 0.286, which is the maximal resemblance computed from the above three nodeset combinations decomposed from $p_{(U,1)}$.

According to Definition 6, the resemblance between an unlabeled data point and a cluster is measured by the maximal value of all the nodeset combinations decomposed from the unlabeled data point. Given an NNIR tree and an unlabeled data point, the approximate algorithm to calculate resemblance is described below.

The combination of n_u-nodesets contained larger importance with higher priority probably close to the maximal estimation. For example, consider the problem in Example 2. The sorted queue is listed as follows:

{[A_1=4], [A_2 = a]}, {[A_1=4], [A_3 = c]},
{[A_3=c]}, {[A_1 = 4]}, {[A_2= a]}.

At first, {[A_1=4], [A_2 = a]} is selected, and the tree nodes {[A_1= 4], [A_3 = c]}, {[A_1= 4]}, {[A_2= a]} are removed from the queue. Then, node {[A_3 = c]} is selected and the queue is empty. Therefore, the combination, {[A_1= 4], [A_2 = a]}, {[A_3 = c]} is utilized to compute the resemblance between the unlabeled data point and cluster c1, and the result is 0.286.

Definition 7 (maximal resemblance). *An unlabeled data point $p_{(U,j)}$ is labeled to the cluster according to the following equation:*

$$\text{Label} = \arg\max_{c_i^*} R(p(U, j), C_i) \qquad (5)$$

Since we measure the similarity between the unlabeled data point $p_{(U,j)}$ and cluster c_i as $R(p_{(U,j)}, c_i)$, the cluster with the maximal resemblance is the most appropriate cluster for the unlabeled data point. Note that after executing the data labeling phase, the labeled data point just obtains a cluster label but is not really added to the cluster. Therefore, NNIR trees will not be modified in the data labeling phase. This can be explained by the reason that the MARDL framework does not cluster data but rather presents the original clustering characteristics to the incoming unlabeled data points. The data labeling algorithm which gives a cluster label for each unlabeled data point based on the NNIR trees of clusters is shown in Algorithm 2.

Algorithm 2 DataLabeling(NNIR_TREEs, U)

Input: the clusters C_i represented by NNIR_TREEs and the unlabeled data set U

1. If there is next tuple in U read the data point $P_{(U,j)}$ from the unlabeled data point.
2. For all clusters, perform the traversal from the root node.
3. Add the tree node from cluster table into queue Q.
4. Sort Q by the importance of the nodeset.
5. Compute resemblance value by using the formula $R(C_i, p(U,j))$
6. Find the maximal resemblance and allocate label to the unlabeled dataset.

Output: the unlabeled data set U with cluster label

IV EXPERIMENTAL RESULTS

This section gives the details about the simulation environment and the datasets utilized in this paper**.** The experiments are conducted on a PC with an Intel Pentium 4 3.2-GHz processor and 1-Gbyte memory running the Windows XP SP2 operating system. The data sets which are chosen to test our technique is eucalyptus data set which comes under the agridata and the German data set.

We have chosen those datasets because of its public availability and also it contains attributes of both numeric as well as categorical attributes. The eucalyptus dataset has 736 instances, each being described by 14 numeric and 6 categorical attributes, totally 20 attributes and the German dataset has 999 instances, each being described by 13 categorical and 8 numerical attributes, totally 21 attributes. In our experiments, the random sampling technique is applied for data sampling, and K-means extension [12], EM Expectation Maximization [11], clustering algorithms which is implemented by WEKA tool [13] are considered to perform clustering on the sampled data sets. We compare Modified MARDL technique with the k-means extension algorithm and EM. The cluster tab in WEKA is used to identify the commonalities or clusters of occurrences within the data set. The option class, the cluster evaluation tab which is available in WEKA is used to compare how well the data compares with a pre-assigned class within the data.

The dataset is grouped into clusters of varying size from 2 to 8. The clustered data errors of these existing and proposed algorithms are observed. Fig 5 shows that, from the experimental result, clustering error rate is less compared to extension k-means and EM algorithm. Similarly Fig 6 shows that, from the experimental result, clustering error rate is less compared to extension k-means and EM algorithm. That is, compared to the extension simple *k*-means and EM algorithm our method MARDL_NNIR performs best in all cases. Furthermore the average clustering errors of our MARDL _NNIR are smaller than that of the other algorithms. The summarization on the relative performance of the 3 algorithms is given in Table 2.

Algorithm	Average Clustering Error
Simple K-Means	0.378
EM	0.350
MARDL_NNIR	0.252

Table 2. Relative Performance of different clustering algorithms (Eucalyptus data set).

The accuracy measure called the clustering accuracy (exactness) which is defined in the given equation below is also considered as another performance evaluation metric.

Fig. 5, Clustered Data Error Vs Different Numbers of Cluster (Eucalyptus data set)

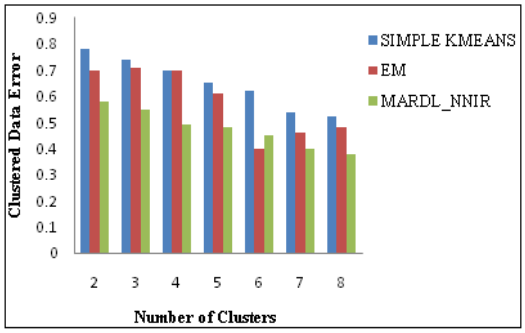

Fig. 6, Clustered Data Error Vs Different Number of Clusters(German data set)

$$r = \frac{1}{n}\sum_{i=1}^{k} a_i \qquad (6)$$

where a_i is the maximum number of data objects of cluster i belonging to the same original classes in the test data (correct answer) and n is the number of data objects in the database.

The graph in Fig 7 states that the accuracy obtained by Modified MARDL algorithm for mixed data set is maximum when compared to Extension K-means and EM clustering algorithms. Similarly Fig 8 shows the accuracy obtained by Modified MARDL algorithm for German dataset.

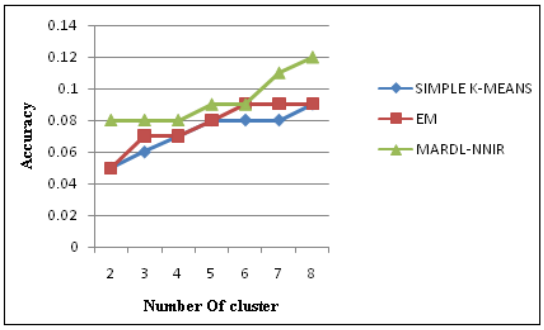

Fig. 7, Accuracy of clustering algorithms on eucalyptus dataset

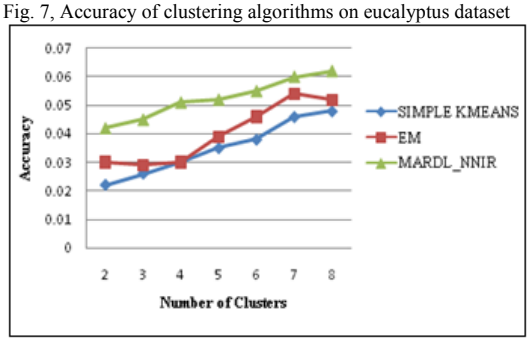

Fig. 8, Accuracy of Clustering algorithms on German data set

V CONCLUSION

Classical clustering algorithms are not able to cluster mixed data set. Only few algorithms are available for clustering numeric, categorical and mixed data set**.** In this paper, we have proposed Modified MARDL technique to cluster mixed data set, which allocates each unlabeled data point into the appropriate cluster. In addition, we have also developed a cluster representative technique, named NIR, to represent clusters which are obtained from the sampled data set by the distribution of the attribute values. The experimental evaluation validates our claim that MARDL is of linear time complexity with respect to the data size, and MARDL preserves clustering characteristics, high intracluster similarity, and low intercluster similarity. Consequently, MARDL is significantly more efficient than prior works which attains the result of high quality.

REFERENCES

[1] Hung-Leng Chen, Kun-Ta Chuang, Member, IEEE, and Ming-Syan Chen, Fellow, IEEE, "On Data Labeling for Clustering Categorical Data" IEEE Trans. Knowledge and Data Eng.,2008

[2] M.-S. Chen, J. Han, and P.S. Yu, "Data Mining: An Overview from a Database Perspective," IEEE Trans. Knowledge and Data Eng.,1996.

[3] A.K. Jain, R.P. Duin, and J. Mao, "Statistical Pattern Recognition: A Review," IEEE Trans. Pattern Analysis and Machine Intelligence, 2000.

[4] A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A Review," ACM Computing Surveys, 1999.
[5] M. Charikar, C. Chekuri, T. Feder, and R. Motwani, "Incremental Clustering and Dynamic Information Retrieval," Proc. 29th Ann. Symp. Theory of Computing, 1997.
[6] D.S.J.M.R. Garey and H.S. Witsenhausen, "The Complexity of the Generalized Lloyd-Max Problem", IEEE Trans. Information Theory, 1982.
[7] S. Guha, R. Rastogi, and K. Shim, "CURE: An Efficient Clustering Algorithm for Large Databases", Proc. ACM SIGMOD IEEE Trans. Knowledge and Data Eng., 1998.
[8] S. Guha, R. Rastogi, and K. Shim, "ROCK: A Robust Clustering Algorithm for Categorical Attributes," Proc. 15th Int'l Conf. Data Eng. (ICDE), 1999.
[9] Zengyou He, Xiaofei Xu, Scengchun Deng, "Scalable Algorithms for Clustering Large Datasets with Mixed Type Atributes," International journal of intelligent systems, vol. 20, 1077–1089 (2005).
[10] C. Shannon, "A Mathematical Theory of Communication," Bell System Technical J., 1948.
[11] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum Likelihood from Incomplete Data via the Em Algorithm," J. Royal Statistical Soc., 1977.
[12] Z. Huang, "Extensions to the K-Means Algorithm for Clustering Large Data Sets with Categorical Values," Data Mining and Knowledge Discovery, Vol. 2, No. 2, pp. 283–304, 1998.
[13] I.H. Witten and E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques" Morgan Kaufmann, 2005.
[14] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. ACM SIGMOD, 1993.
[15] N. Mishra, D. Oblinger, and L. Pitt, "Sublinear Time Approximate Clustering," Proc. 12th Ann. ACM-SIAM Symp. Discrete Algorithms (SODA), 2001.