1

Discovery of the Latent Supportive Relevance in Medical Data Mining

Sam Chao[1], Fai Wong[1], Qiulin Ding[2], Yiping Li[1] and Mingchui Dong[1]
[1]*University of Macau, Macau*
[2]*Nanjing University of Aeronautics and Astronautics, Nanjing, China*

1. Introduction

The purpose of a classification learning algorithm is to accurately and efficiently map an input instance to an output class label, according to a set of labeled instances. Decision tree is such a method that most widely used and practical for inductive inference in the data mining and machine learning discipline (Han & Kamber, 2000). However, many decision tree learning algorithms degrade their learning performance due to irrelevant, unreliable or uncertain data are introduced; or some focus on univariate only, without taking the interdependent relationship among others into consideration; while some are limited in handling the attributes[1] with discrete values. All these cases may be caused by improper pre-processing methods, where feature selection (FS) and continuous feature discretization (CFD) are treated as the dominant issues. Even if a learning algorithm is able to deal with various cases, it is still better to carry out the pre-processing prior the learning algorithm, so as to minimize the information lost and increase the classification accuracy accordingly. FS and CFD have been the active and fruitful fields of research for decades in statistics, pattern recognition, machine learning and data mining (Yu & Liu, 2004). While FS may drastically reduce the computational cost, decrease the complexity and uncertainty (Liu & Motoda, 2000); and CFD may decrease the dimensionality of a specific attribute and thus increase the efficiency and accuracy of the learning algorithm.

As we believe that, among an attributes space, each attribute may have certain relevance with another attribute, therefore to take the attributes relevant correlation into consideration in data pre-processing is a vital factor for the ideal pre-processing methods. Nevertheless, many FS and CFD methods focus on univariate only by processing individual attribute independently, not considering the interaction between attributes; this may sometimes loose the significant useful hidden information for final classification. Especially in medical domain, a single symptom seems useless regarding diagnostic, may be potentially important when combined with other symptoms. An attribute that is completely useless by itself can provide a significant performance improvement when taken with others. Two attributes that are useless by themselves can be useful together (Guyon & Elisseeff, 2003; Caruana & Sa, 2003). For instance, when learning the medical data for disease diagnostic, if

[1] In this paper, attribute has the same meaning as feature, they are used exchangeable within the paper.

Open Access Database www.intechweb.org

Source: Data Mining in Medical and Biological Research, Book edited by: Eugenia G. Giannopoulou,
ISBN 978-953-7619-30-5, pp. 320, December 2008, I-Tech, Vienna, Austria

a dataset contains attributes like patient *age*, *gender*, *height*, *weight*, *blood pressure*, *pulse*, *ECG result* and *chest pain*, etc., during FS pre-processing, it is probably that attribute *age* or *height* alone will be treated as the least important attribute and discarded accordingly. However, in fact attribute *age* and *height* together with *weight* may express potential significant information: whether a patient is *overweight*? On the other hand, although attribute *blood pressure* may be treated as important regarding classifying a cardiovascular disease, while together with a useless attribute *age*, they may present more specific meaning: whether a patient is *hypertension*? Obviously, the compound features *overweight* and/or *hypertension* have more discriminative power regarding to disease diagnostic than the individual attributes stated above. It is also proven that a person is *overweight* or *hypertension* may have more probabilities to obtain a cardiovascular disease (Jia & Xu, 2001).

Moreover, when processing CFD, the discretized intervals should make sense to human expert (Bay, 2000; Bay, 2001). We know that a person's *blood pressure* is increasing as one's *age* increasing. Therefore it is improper to generate a cutting point such as 140mmHg and 90mmHg for systolic pressure and diastolic pressure, respectively. Since the standard for diagnosing hypertension is a little bit different from young people (orthoarteriotony is 120-130mmHg/80mmHg) to the old people (orthoarteriotony is 140mmHg/90mmHg) (Gu, 2006). If the blood pressure of a person aged 20 is 139mmHg/89mmHg, one might be considered as a potential hypertensive. In contrast, if a person aged 65 has the same blood pressure measurement, one is definitely considered as normotensive. Obviously, to discretize the continuous-valued attribute *blood pressure*, it must take at least the attribute *age* into consideration. While discretizing other continuous-valued attribute may not take *age* into consideration. This demonstrates again that a useless attribute *age* is likely to be a potentially useful attribute once combined with attribute *blood pressure*. The only solution to address the mentioned problem is to use multivariate interdependent discretization in place of univariate discretization.

In the next section, we show the importance of attributes interdependence by describing in detail the multivatiate interdependent discretization method – MIDCA. Then in section 3, we demonstrate the significance of attributes relevance by specifying the latent utility of irrelevant feature selection – LUIFS in detail. The evaluations of our proposed algorithms to some real-life datasets are performed in section 4. In final section, we summarize our paper and present the future directions of our research.

2. Multivariate discretization

Many discretization algorithms developed in data mining field focus on univariate only, which discretize each attribute with continuous values independently, without considering the interdependent relationship among other attributes, at most taking the interdependent relationship with class attribute into account, more detail can be found in(Dougherty et al., 1995; Fayyad & Irani, 1993; Liu & Setiono, 1997; Liu et al., 2002). This is unsatisfactory in handling the critical characteristics possessed by medical area. There are few literatures discussed about the multivariate interdependent discreitzation methods. The method developed in (Monti & Gooper, 1998) concentrates on learning Bayesian network structure; hence the discretization is relied on the Bayesian network structure being evaluated only, which is unable to be applied in other structures, such as decision tree learning structure.

Multivariate interdependent discretization concerns the correlation between the attribute being discretized and the other potential interdependent attributes. Our MIDCA – Multivariate Interdependent Discretization for Continuous Attribute is based on the normalized relief (Kira & Rendell, 1992a; Kira & Rendell, 1992b) and information

theory(Fayyed & Irani, 1993; Mitchell, 1997; Zhu, 2000), to look for the best correlated attribute for the continuous-valued attribute being discretized as the interdependent attribute to carry out the multivariate discretization. In order to obtain the good quality for a multivariate discretization, discovery of a best interdependent attribute against the continuous-valued attribute is considered as the primary essential task.

2.1 MIDCA method

MIDCA is interested mainly in discovering the best interdependent attribute relative to the continuous-valued attribute being discretized. As we believe that a good multivariate discretization scheme should highly rely on the corresponding perfect correlated attributes. If assume that a dataset $S = \{s_1, s_2, \ldots, s_N\}$ contains N instances, each instance $s \in S$ is defined over a set of M attributes (features) $A = \{a_1, a_2, \ldots, a_M\}$ and a class attribute $c \in C$. For each continuous-valued attribute $a_i \in A$, there exists at least one $a_j \in A$, such that a_j is the most correlated with a_i, or vice versa, since the correlation is measured symmetrically. For the purpose of finding out such a best interdependent attribute a_j for each continuous-valued attribute a_i, both gain information in equation (2) that derived from entropy information in equation (1) and relief measures in equation (3) depicted below are taken into account to capture the interaction among the attributes space.

$$Entropy(S) = -\sum_{i \in C} p(S_i) log(p(S_i)) \tag{1}$$

where $p(S_i)$ is the proportion of instances S belonging to class i; based on this measure, the most informative attribute A relative to a collection of instances S can be defined as:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{2}$$

where $Values(A)$ is the set of all distinct values of attribute A; S_v is the subset of S for which attribute A has value v, that is $S_v = \{s \in S \mid A(s) = v\}$. While the reformulated relief measure can be defined as to estimate the quality of an attribute over all training instances:

$$Relief_A = \frac{Gini'(A) \times \sum_{x \in X} p(x)^2}{(1 - \sum_{c \in C} p(c)^2) \sum_{c \in C} p(c)^2} \tag{3}$$

where C is the class attribute and $Gini'$ is a variance of another attribute quality measure algorithm Gini-index (Breiman, 1996).

Then we use the symmetric relief and entropy information algorithms to calculate the interdependent weight for each attribute pair $\{a_i, a_j\}$ where $i \neq j$. However, the measures output from two symmetric measures are in different standards, the only way to balance them is to normalize the output values by using proportions in place of real values. Finally averaging the two normalized proportions as in equation (4) and chooses the best interdependent weight amongst all potential interdependent attributes as our target.

$$InterdependentWeight(a_i, a_j) = \left[\frac{SymGain(a_i, a_j)}{\sqrt{\sum_{M \neq i}^{A} SymGain(a_i, a_M)^2}} + \frac{SymRelief(a_i, a_j)}{\sqrt{\sum_{M \neq i}^{A} SymRelief(a_i, a_M)^2}} \right] / 2 \tag{4}$$

where

$$SymGain(A,B) = \left[Gain(A,B) + Gain(B,A)\right]/2$$

and

$$SymRelief(A,B) = \left[\frac{Gini'(A)\times\sum_{x\in X} p(x)^2}{(1-\sum_{b\in B} p(b)^2)\sum_{b\in B} p(b)^2} + \frac{Gini'(B)\times\sum_{y\in Y} p(y)^2}{(1-\sum_{a\in A} p(a)^2)\sum_{a\in A} p(a)^2}\right]/2$$

The advantage of incorporating the measures of entropy information and relief in MIDCA algorithm is to minimize the uncertainty between the interdependent attribute and the continuous-valued attribute being discretized and at the same time to maximize their correlation by discovering the perfect interdependencies between them. However, if an interdependent attribute is a continuous-valued attribute too, it is first discretized with entropy-based discretization method (Dougherty et al., 1995; Fayyad & Irani, 1993). This is important and may reduce the bias of in favor of the attribute with more values. Furthermore, our method creates an interdependent attribute for each continuous-valued attribute in a dataset rather than using one for all continuous-valued attributes, this is also the main factor for improving the final classification accuracy.

MIDCA ensures at least binary discreatizaiton, which is different from other methods that sometimes the boundary of a continuous-valued attribute is [-∞, +∞]. We realized that if a continuous-valued attribute generates null cutting point means that the attribute is useless, hence increase the classification uncertainty. This may finally cause the higher error rate in the learning process. We believe that most continuous-valued attributes in medical domain have their special meanings, even though it alone seems unimportant, while it becomes useful when combining with other attributes. The before-mentioned examples are some typical ones. Furthermore, most figures express the degrees and seriousness of the specific illness, such as *blood pressure* may indicate the level of hypertension; higher *heart rate* may represent the existence of cardiovascular disease; while *plasma glucose* is an index for diabetes and so on, hence their discretization cannot be ignored.

The only drawback of MIDCA is its slightly heavy complexity. In the worst case, all N attributes in a dataset are numeric, but the interdependent weighting is calculated symmetrically, so there are $N/2$ attributes involved into calculations. For the first attribute taking into calculation, (N-1) times of executions are necessary; while for the second attribute taking into calculation, (N-3) times of executions should be carried out, and so on. Since such calculation is decreased gradually, so the total execution times are at most (N-1)$*N/2$, i.e., $O(N^2/2)$. However, such case is only a minority. Most real life datasets have low percentage of numeric attributes, then the execution time of the algorithm becomes less influential compared with classification accuracy.

2.2 MIDCA algorithm

Different from other discretization algorithms, MIDCA is carried out with respect to the best interdependent attribute that discovered from equation (4) in addition to the class attribute. Moreover, we assume that the interdependent attribute *INT* has T discrete values; as such each of its distinct value identifies a subset in the original data set S, the probability should

be generated relative to the subset in place of the original data set. Therefore, the combinational probability distribution over the attribute space $\{C\} \cup A$ can be redefined based on equation (2) as well as the information gain algorithm as following:

$$MIDCAInfoGain(A, P; INT_T, S) = Entropy(S \mid INT_T) - \sum_{v \in Values(A) \mid INT_T} \frac{|S_v|}{|S|} Entropy(S_v) \quad (5)$$

where P is a collection of candidate cutting points for attribute A under the projection of value T for the interdependent attribute INT, the algorithm defines the class information entropy of the partition induced by P. In order to emphasize the importance of the interaction respect to the interdependent attribute INT, $Entropy(S)$ is replaced by the conditional entropy $Entropy(S \mid INTT)$. Consequently, $v \in Values(A) \mid INT_T$ becomes the set of all distinct values of attribute A of the cluster induced by T of interdependent attribute INT; and S_v is the subset of S for which attribute A has value v and under the projection of T for INT, i.e., $S_v = \{s \in S \mid A(s) = v \wedge INT(s) = T\}$.

Now we compendiously present our MIDCA algorithm as below and the evaluation for such algorithm is illustrated in section 4:

1. Sort the continuous-valued attributes for the data set in ascending order;
2. Discovery the best interdependent attributes pair by algorithm INTDDiscovery;
3. Calculate the MIDCAInfoGain measure and select the best cutting point;
4. Evaluate whether stopping the calculation according to MDLP;
5. Repeat step 3 if the test failed. Else, order the best cutting points to generate the discretized data set.

INTDDiscovery algorithm

    If the interdependent attribute is a continuous-valued attribute
        Discretize using entropy-based method and select the best cutting point;
        Test to stop using MDLP;
    Calculate the symmetric entropy information measure for the pair of attributes;
    Calculate the symmetric relief measure for the pair of attribute;
    Normalize the symmetric entropy information and relief measure;
    Average the two normalized measures;
    Select the one with the highest average measure;

End algorithm;

3. Feature selection

More features no longer mean more discriminative power; contrarily, they may increase the complexity and uncertainty of an algorithm, thus burden with heavy computational cost. Therefore various feature selection algorithms have been introduced in (Liu & Motoda, 2000) as well as their evaluations and comparisons in (Hall & Holmes, 2003; Molina et al., 2002; Dash & Liu, 1997). Feature selection can be defined as a process of finding an optimal subset of features from the original set of features, according to some defined feature selection criterion (Cios et al., 1998).

Suppose a dataset D contains a set of M original attributes $OriginalA = \{a_1, a_2, \ldots, a_M\}$ and a class attribute $c \in C$, i.e., $D = OriginalA \cup C$. The task of feature selection is to find such a subset of N attributes among M that $OptimalA = \{a_1, a_2, \ldots, a_N\}$, where $N \leq M$ and $OptimalD =$

OptimalA ∪ *C*, hence the *ClassificationErrorRate*(*OptimalD*)≤*ClassificationErrorRate*(*D*). Features can be defined as relevant, irrelevant or redundant. A relevant feature is neither irrelevant nor redundant to the target concept; an irrelevant feature does not affect the target concept in any way, and a redundant feature does not add anything new to the target concept (John et al., 1994).

Many feature selection algorithms generate the optimal subset of relevant features by ranking the individual attribute or subset of attributes. Each time the best single attribute or attributes subset will be selected, the iteration will be stopped when some pre-defined filtering criteria has been matched. For instance, a forward selection method recursively adds a feature x_i to the current optimal feature subset *OptimalA*, among those have not been selected yet in *OriginalA*, until a stopping criterion is conformed to. In each step, the feature x_i that makes evaluation measure *W* be greater is added to the optimal set *OptimalA*. Starting with *OptimalA* = {}, the forward step consists of:

$$OptimalA := OptimalA \cup \{OriginalA \setminus OptimalA \mid W(OptimalA \cup \{x_i\}) \text{ is maximum}\} \tag{6}$$

Nevertheless, this method does not have in consideration certain basic interactions among features, i.e., if x_1 and x_2 are such interacted attributes, that $W(\{x_1, x_2\}) >> W(\{x_1\}), W(\{x_2\})$, neither x_1 nor x_2 could be selected, in spite of being very useful (Molina et al., 2002). Since most feature selection methods assume that the attributes are independent rather than interactive, hence their hidden correlations have been ignored during feature generation like the one above. As before-mentioned, in medical domain, sometimes a useless symptom by itself may become indispensable when combined with other symptom. To overcome such problem, our LUIFS (latent utility of irrelevant feature selection) method takes interdependence among attributes into account instead of considering them alone.

3.1 LUIFS

LUIFS mainly focuses on discovering the potential usefulness of supportive irrelevant features. It takes the inter-correlation between irrelevant attributes and other attributes into consideration to measure the latent importance of those irrelevant attributes. As we believe that in medical field an irrelevant attribute is the one that providing neither explicit information nor supportive or implicit information. In (Pazzani, 1996), Pazzani proposed a similar method to improve the Bayesian classifier by searching for dependencies among attributes. However, his method has several aspects that are different from ours: (1) it is restricted under the domains on which the naïve Bayesian classifier is significantly less accurate than a decision tree learner; while our method aims to be a preprocessing tool for most learning algorithms, no restrictions on the learner. (2) It used wrapper model to construct and evaluate a classifier at each step; while a simpler filter model is used in our method, which minimized computational complexity and cost. (3) His method created a new compound attribute replacing the original two attributes in the classifier after joining attributes. This may result in a less accurate classifier, because joined attributes have more values and hence there are fewer examples for each value, as a consequence, joined attributes are less reliable than the individual attributes. Contrarily, our method adds the potential useful irrelevant attributes to assist increasing the importance of the valuable attributes, instead of dynamically joining the attributes. Our experimental evidence in the

corresponding section will show that there are additional benefits by including the latent useful irrelevant attributes.

LUIFS generates an optimal feature subset in two phases: (1) Attribute sorting: a general feature ranking process by sorting the attributes according to the relevance regarding the target class, and filters the ones whose weight is below a pre-defined threshold σ; (2) Latent utility attribute searching: for each filtered irrelevant and unselected attribute, determine its supportive importance by performing a multivariate interdependence measure that combined with another attribute. There are two cases that an irrelevant attribute in the second phase becomes a potentially supportive relevant attribute and will be selected into the optimal feature subset: (1) if a combined attribute is a relevant attribute and already be selected, then the combinatorial measure should be greater than the measure of such combined attribute; (2) if a combined attribute is an irrelevant attribute too and filtered out in the first phase, then the combinatorial measure should be greater than the pre-defined threshold σ, such that both attributes become relevant and will be selected. Two hypotheses have been conducted from our method LUIFS as below:

Hypothesis1. If a combined attribute helps in increasing the importance of a yet important attribute; including such combined attribute into the optimal feature subset may most probably improve the final class discrimination.

Hypothesis2. If a combined attribute helps an ignored attribute in reaching the importance threshold; including both attributes into the optimal feature subset may most probably improve the class discrimination

3.2 Attribute sorting

In this phase, we first sort all N features according to the importance respect to the target class. Information gain theory in equation (2) is used as the measurement, which may find out the most informative (important) attribute A relative to a collection of examples S. Attributes are sorted in descending order, from the most important one (with the highest information gain) to the least useful one. In LUIFS, we introduced a threshold ϖ to distinguish the weightiness of an attribute. The value of a threshold either too high or too low may cause the attributes insufficient or surplus. Therefore it is defined as a mean value excluding the ones with maximum and minimum weights, in order to eliminate as much bias as possible. Equation (7) illustrates the threshold.

$$\varpi = mean(\sum_{i=1}^{N} InfoGain(S, A_i) - \max(InfoGain) - \min(InfoGain)) \tag{7}$$

Then, an attribute A will be selected into the optimal feature subset if $Gain(S, A) > \varpi$. In addition, if an attribute A is a numeric type, it is discretized first by using the method of (Fayyad & Irani, 1993). This phase requires only linear time in the number of given features N, i.e. $O(N)$.

3.3 Latent utility attribute searching

This phase is the key spirit in LUIFS, since our objective is to uncover the usefulness of the latent or supportive relevant attributes, particularly those specifically owned in medical domain. It is targeted at the irrelevant attributes that filtered out from the Attribute sorting phase, to look for their latent utilities in supporting other relevant attributes. To determine

whether an irrelevant attribute is potentially important or not, we measure the interdependence weight between it and another attribute regarding the class attribute. We use Relief theory (Kira & Rendell, 1992a; Kira & Rendell, 1992b), which is a feature weighting algorithm for estimating the quality of attributes such that it is able to discover the interdependencies between attributes. It randomly picks a sample, for each sample it finds *near hit* and *near miss* sample based on distance measure. Relief works for noisy and correlated features, and requires only linear time in the number of features and the number of samples (Dash & Liu, 1997). Our method adopts the combinatorial relief in equation (8), which measures the interdependence between a pair of attributes and the class attribute rather than a single attribute. It approximates the following probability difference:

$$\begin{aligned} W[a_i + a_j] = P(&\text{different value of } a_i + a_j \mid \\ &\text{nearest instances from different class } C) \\ - P(&\text{different value of } a_i + a_j \mid \\ &\text{nearest instances from same class } C) \end{aligned} \tag{8}$$

where a_i is an irrelevant attribute, whose information gain measure in equation (2) is less than the mean threshold ϖ in equation (7) and is filtered out in Attribute sorting phase; a_j is a combined attribute either relevant or irrelevant. $W[a_i+a_j]$ is the combinatorial weighting between attributes a_i and a_j regarding the class attribute C, and P is a probability function. Equation (8) measures the level of hidden supportive importance for an irrelevant attribute to another attribute, hence the higher the weighting, the more information it will provide, such that the better the diagnostic results. According to our hypotheses, an irrelevant attribute a_i may become latent relevant if there exists another attribute a_j, where $a_i \neq a_j$, so that the combinatorial measure $W[a_i+a_j] > W[a_j]$ if a_j is an explicit relevant attribute and already be selected into the optimal feature subset; or $W[a_i+a_j] > \varpi$ (a pre-defined threshold) if a_j is an irrelevant attribute also.

Unlike the Attribute sorting phase, the complexity of this searching phase is no longer simple linear in time. In the worst case, if there is only one important attribute was selected after Attribute sorting phase, that is, there are (N-1) irrelevant attributes were ignored and unselected. For each irrelevant attribute $a_i \in$ *UnselectedAttributes*, calculate its interdependent weight with another attribute. Again in the worst case, if a_i could not encounter an attribute that makes it becoming useful, then the searching algorithm should be repeated for (N-1) times. Whereas the algorithm is symmetric, i.e. $W[a_i+a_j] = W[a_j+a_i]$, so the total times of searching should be in half respect to the number of *UnselectedAttributes*, which equals to (N-1)/2. Therefore, the complexity of such Latent utility attribute searching for irrelevant attributes is (N-1)∗(N-1)/2 for the worst case, i.e. $O((N-1)^2/2)$. Nevertheless the feature selection is typically done in an off-line manner (Jain & Zongker, 1997), in the meantime, the capacity of the hardware components increase while the price of them decrease. Hence, the execution time of an algorithm becomes less important compared with its final class discriminating performance.

3.4 LUIFS algorithm

In this section, the pseudo code of LUIFS algorithm is illustrated as following. First, several variables are defined, and then the program body are sketched subsequently.

Mean – the threshold value for picking the relevant attributes
selectedlist – list of attributes that are treated as important and used in final classification
unselectedlist – list of attributes that are treated as useless and are ignored
N – the total number of attributes in a dataset
A_i – the i^{th} attribute in a dataset
M – the total number of attributes in *unselectedlist*
Map – an array of boolean values indicating whether an attribute is processed multivariate interdependent attributes mapping or not

```
MIFS algorithm
    selectedlist := {};
    unselectedlist := {};
    -- phase one processing
    for i := 1 to N do
  if InformGain(Ai) >= Mean then
            selectedlist := selectedlist + {Ai};
        else
            unselectedlist := unselectedlist + {Ai};
        end if;
    end for;
    -- phase two processing
    for i := 1 to M do
        for j := 1 to N do
          if {Aj} is in selectedlist and {Ai} <> {Aj} then
            if Measure[Ai+Aj] > Measure[Aj] then
                selectedlist := selectedlist + {Ai};
                unselectedlist:=unselectedlist-{Ai};
                Map[Ai] := true;
            end if;
          elseif {Aj} is in unselectedlist and
          {Ai} <> {Aj} then
            if Measure[Ai+Aj]>Mean and
            Map[Aj] = false then
                selectedlist := selectedlist + {Ai};
                selectedlist := selectedlist + {Aj};
                unselectedlist:=unselectedlist-{Ai};
                unselectedlist:=unselectedlist-{Aj};
                Map[Ai] := true;
                Map[Aj] := true;
            end if;
          end if;
        end for;
    end for;
end algorithm;
```

4. Experiments

In this section, we have verified the superiority of our before-mentioned theories by evaluating the effectiveness of proposed pre-processing methods: MIDCA and LUIFS. The solid evidences demonstrate our belief: by uncovering potential attributes relevance during pre-processing step (either FS or CFD) and taking them into the data mining task, the learning performance can have significant improvement. The experiments are carried out individually with learning algorithms into two subsections, and are performed on several real life datasets from UCI repository (Blake & Merz, 1998). Both experiments adopt ID3 (Quinlan, 1986) from (Blake & Merz, 1998) and C4.5 (Quinlan, 1993; Quinlan, 1996) as the learning algorithms for the comparisons to be performed. Table 1 depicts the detailed characteristics of various datasets, while the datasets beneath the double line separator are only involved in the experiment of LUIFS.

4.1 Experiment of MIDCA

The last four datasets in Table 1 are not considered in this experiment, since they do not have numeric attributes, so there are eight datasets to take performance in the experiment of MIDCA. In order to make comparisons between different discretization methods, we downloaded the discretization program – *Discretize* (a discretization program that downloaded from UCI repository, here it is used as a preprocessing tool for ID3 learning algorithm only). On the other hand, since C4.5 is embedded with the discretization function in the learning algorithm, *Discretize* is not applicable to this algorithm. Table 2 shows the results in classification error rate obtained by using ID3 learning algorithm with *Discretize* and MIDCA pre-processing methods respectively; while Table 3 shows the results in classification error rate obtained by using C4.5 learning algorithm without/with MIDCA pre-processing method respectively.

Dataset	Feature Type		Instance size	Class
	Numeric	Nominal		
Cleve	6	7	303	2
Hepatitis	6	13	155	2
Hypothyroid	7	18	3163	2
Heart	13	0	270	2
Sick-euthyroid	7	18	3163	2
Auto	15	11	205	7
Breast	10	0	699	2
Diabetes	8	0	768	2
Mushroom	0	22	8124	2
Parity5+5	0	10	1124	2
Corral	0	6	64	2
Led7	0	7	3200	10

Table 1. Bench-mark datasets from UCI repository

Dataset	Discretize (%)	MIDCA (%)
Cleve	26.733±4.426	17.822±3.827
Hepatitis	19.231±5.519	9.615±4.128
Hypothyroid	1.232±0.340	0.000±0.000
Heart	15.556±3.842	17.778±4.053
Sick-euthyroid	3.697±0.581	0.000±0.000
Auto	26.087±5.325	25.373±5.356
Breast	3.863±1.265	4.292±1.331
Diabetes	25.781±2.739	32.031±2.922
Average	15.27	13.36

Table 2. Results in classification error rate of ID3 algorithm with various discretization methods

The experiments reveal that our method MIDCA decreases the classification error rate for ID3 learning algorithm on all but three datasets among eight; similarly MIDCA decreases the classification error rate for C4.5 learning algorithm on all but two out of eight datasets. For the rest of the datasets, MIDCA provides a significant improvement in classification accuracy, especially on two data sets: *Hypothyroid* and *Sick-euthyroid*, which approached to zero error rates for both learning algorithms. As observed from Table 2 and Table 3, MIDCA slightly decreases the performance on two and three datasets out of eight for C4.5 and ID3 algorithms respectively, and all these datasets contain only continuous attributes. This is

Dataset	C4.5 (%)	MIDCA (%)
Cleve	24.8	17.8
Hepatitis	17.3	9.6
Hypothyroid	0.9	0.0
Heart	17.8	21.1
Sick-euthyroid	3.1	0.0
Auto	29.0	23.9
Breast	5.6	3.9
Diabetes	30.9	31.6
Average	16.18	13.49

Table 3. Results in classification error rate of C4.5 algorithm without/with MIDCA method

due to the MIDCA algorithm needs to perform a univariate discretization once prior the multivariate discretization if an interdependent attribute is a continuous-valued attribute also. Such step increases the uncertainty of the attribute being discretized, hence increases the error rate accordingly. Although it is inevitable that a continuous-valued attribute to be selected as a best interdependent attribute regarding the attribute being discretized, our theories DO work in most cases. From the average error rate results described in Table 2 and

Table 3, obviously our method MIDCA indeed decreases the classification error rate from 15.27% down to 13.36% for ID3 algorithm; and from 16.18% down to 13.49% for C4.5 algorithm. The improvements relative to both algorithms reach to approximately 12.5% and 16.6% respectively. The comparisons under various situations are clearly illustrated in Figure 1 and Figure 2 respectively.

4.2 Experiment of LUIFS

In this experiment, all twelve datasets in Table 1 are used. In order to make clear comparison, experiments on two learning algorithms without feature selection (NoFS) and with information gain attribute ranking method (ARFS) are performed, as well as LUIFS. Table 4 and Table 5 summarize the results in error rates of ID3 and C4.5 algorithms without/with feature selection respectively. The last column LRA in Table 4 indicates the number of irrelevant attributes becoming useful and is selected in LUIFS method, which further demonstrates the importance of supportive attributes.

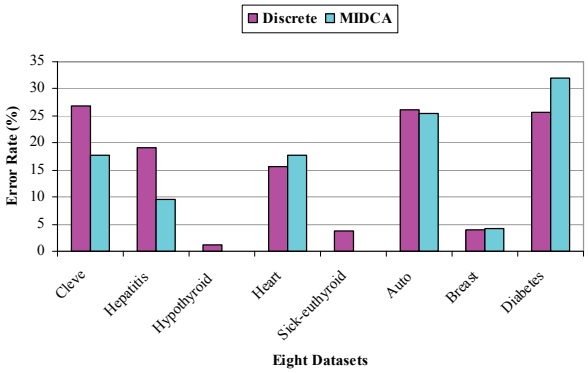

Fig. 1. Comparisons in classification error rate of ID3 algorithm with different discretization methods

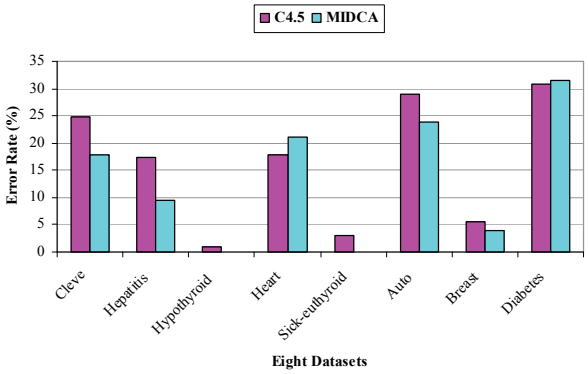

Fig. 2. Comparisons in classification error rate of C4.5 algorithm without/with MIDCA method

As observed from the results in Table 4 and Table 5, LUIFS slightly increases the classification error rate in compared with ARFS for both learning algorithms on only one dataset – *Heart*, amongst all twelve datasets; but its results are still better than the learning algorithms with NoFS. This is because the dataset *Heart* contains numeric attributes only, which needs prior discretized to the latent utility attribute searching being carried out. Such step increases the uncertainty to the attribute being correlated, hence increases the error rate accordingly.

Nevertheless, LUIFS may still help in obtaining quite good classification accuracy, even for the simplest learning algorithm. The average results from Table 4 and Table 5 reveal that LUIFS does improve on the final classification accuracy significantly. Especially with ID3 algorithm, which the accuracy growth rate relative to NoFS and ARFS are 10.79% and 11.09% respectively; while with C4.5 method, the corresponding growth rates are not as good as with ID3, they are only 8.51% and 7.86% respectively. Even so, the classification accuracy with LUIFS in Table 5 is still the highest. The vast improvements made by LUIFS in ID3 learning algorithm is due to ID3 is a pure decision tree algorithm, without any built-in pre-processing functions, such as FS or CFD, hence there is still certain rooms for improving; while C4.5 is embedded with its own feature selection and pruning functions already, it filters the useless features from its own point of view, no matter additional feature selection algorithm attached or not, hence limiting the great improvements. The comparisons under various situations are apparently portrayed in Figure 3 and Figure 4 respectively.

Dataset	**NoFS**	**ARFS**	**LUIFS**	**LRA**
Cleve	35.64	23.762	22.772	2
Hepatitis	21.154	15.385	13.462	7
Hypothyroid	0.948	0.190	0.190	8
Heart	23.333	13.333	18.889	2
Sick-euthyroid	3.791	0	0	7
Auto	26.087	18.841	18.841	3
Breast	5.579	6.438	5.579	2
Diabetes	Error[2]	35.156	32.131	3
Mushroom	0	0	0	1
Parity5+5	49.291	50	49.291	4
Corral	0	12.5	0	2
Led7	33.467	42.533	32.8	1
Average	18.12	18.18	16.16	3.5

Table 4. Results in classification error rate of ID3 algorithm without/with various feature selections

[2] Error refers to the program error during execution, hence no result at all. Therefore, the corresponding average error rate with NoFS is only approximated without taking such result into calculation.

5. Conclusion and future research

5.1. Conclusions

In this paper, we have proposed a novel concept of improving the classification accuracy for common learning algorithms by uncovering the potential attributes relevance through multivariate interdependent attribute pre-processing methods. However, many common pre-processing methods ignored the latent inter-correlation between attributes, this sometimes may loose the potential hidden valuable information, thus limit the performance of the learning algorithms. Especially in medical domain, diagnostic accuracy is treated as the most important issue, in order to approach to the highest accuracy, qualities of the pre-processing methods are quite essential, thereby finding out the most useful attributes relevance is a key factor for a successful method.

Dataset	NoFS	ARFS	LUIFS
Cleve	20	22.2	19.3
Hepatitis	15.6	7.9	7.9
Hypothyroid	1.1	0.4	0.4
Heart	17.8	14.4	15.6
Sick-euthyroid	2.5	2.5	2.4
Auto	27.1	21.9	21.8
Breast	5.7	5.6	5.4
Diabetes	30.9	30.1	30.1
Mushroom	0	0	0
Parity5+5	50	50	50
Corral	9.4	12.5	9.4
Led7	32.6	43.7	32.3
Average	17.72	17.6	16.21

Table 5. Results in classification error rate of C4.5 algorithm without/with various feature selections

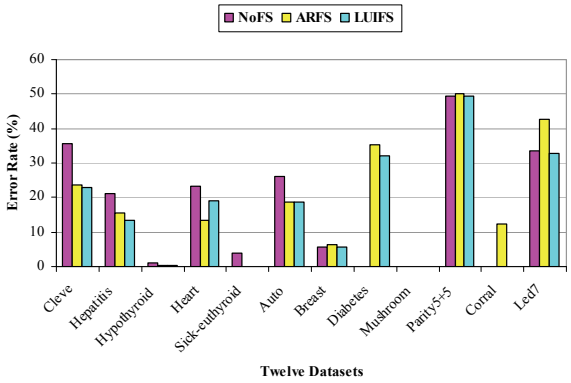

Fig. 3. Comparisons in classification error rate of ID3 algorithm without/with various feature selections

Two pre-processing methods MIDCA and LUIFS have been presented in detail, to verify the superiority of our theories. The empirical evaluation results from various experiments further demonstrated their effectiveness by applying two different classification algorithms on the datasets with and without MIA-Processing. The significant performance improvement by using MIDCA method indicates that it can accurately and meaningfully discretize a continuous-valued attribute by discovering its perfect matched interdependent attribute. In addition, the usage of LUIFS method on the other hand can minimize the classification error rate as well, even for the learning algorithm with built-in feature selection and discretization capabilities.

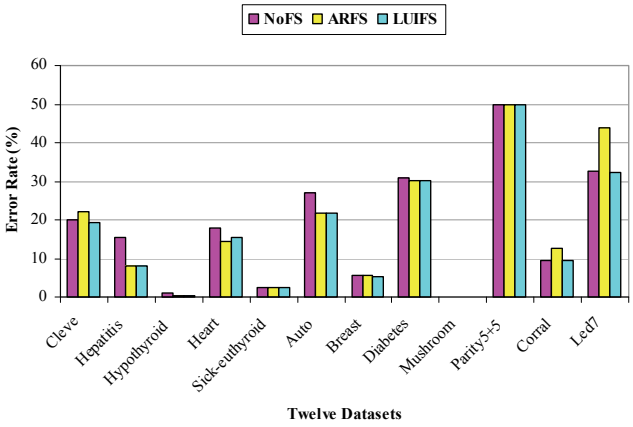

Fig. 4. Comparisons in classification error rate of C4.5 algorithm without/with various feature selections

Finally, the purpose of our methods to incorporate relief theory with information measure is to minimize the uncertainty and at the same time maximize the classification accuracy. Although the methods are designed for using in medical domain, they still can be applied to other domains, since the models do not rely on any background knowledge to be constructed.

5.2 Future research

The future work of our research can be substantially extended to include more factors into comparisons; they may be summarized into the following aspects:

1. Involve more learning algorithms in data mining, such as Naïve-Bayes (Langley et al., 1992), 1R (Holte, 1993), ANNs (Hand et al., 2001; Kasabov, 1998) or Gas (Melanie, 1999), etc. This may validate the practicability of our proposed MIA-Processing methods.
2. Implement the existing pre-processing methods, such as ReliefF (Kononenko, 1994), Focus (Almualim & Dietterich, 1992) or MLDM (Sheinvald et al., 1990), etc. This may allow us to discover the weakness of our current methods, and then improve accordingly.
3. Include different types of real-life or artificial datasets into the experiments, to enrich the usability of our MIA-Processing methods.

Furthermore, our next principal direction for the future research is focused on the optimization of both MIA-Processing methods. A feasible solution should be investigated to eliminate the uncertainties and to reduce the complexities as much as possible, in order to adapt to the future development trend in data mining. Besides, the existing weaknesses of our methods should be found out and reimplemented. They should not decrease the accuracy in learning on a dataset contains all numeric attributes; and should handle high dimensional or large datasets efficiently.

6. References

J. Han, and M. Kamber, *Data Mining - Concepts and Techniques*, Morgan Kaufmann Publishers, 2000.

L. Yu, and H. Liu, "Efficient Feature Selection via Analysis of Relevance and Redundancy", *Journal of Machine Learning Research*, *Vol. 5*, 2004, pp. 1205-1224.

H. Liu, and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*, Kluwer Academic Publisher, 2000.

I. Guyon, and A. Elisseeff, "An Introduction to Variable and Feature Selection", *Journal of Machine Learning Research, Vol. 3*, 2003, pp. 1157-1182.

R. Caruana, and V.R. Sa, "Benefitting from the Variables that Variable Selection Discards", *Journal of Machine Learning Research, Vol. 3*, 2003, pp. 1245-1264.

L. Jia, , and Y. Xu, *Guan Xing Bing De Zhen Duan Yu Zhi Liao*, Jun Shi Yi Xue Ke Xue Chu Ban She, 2001.

S.D. Bay, "Multivariate Discretization of Continuous Variables for Set Mining", In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2000, pp. 315–319.

S.D. Bay, "Multivariate Discretization for Set Mining", *Knowledge and Information Systems, Vol. 3(4)*, 2001, pp. 491-512.

W.Q. Gu, "Xin Xue Guan Ji Bing Jian Bie Zhen Duan Xue", Xue Yuan Chu Ban She, 2006

J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and Unsupervised Discretization of Continuous Features", *In Proceedings of the Twelfth International Conference on Machine Learning,* Morgan Kaufmann Publishers, San Francisco, CA., 1995.

U.M. Fayyad, and K.B. Irani, "Multi-interval Discretization of Continuous-valued Attributes for Classification Learning", *In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, 1993, pp. 1022–1027.

H. Liu, and R. Setiono, "Feature Selection via Discretization", *Technical report*, Dept. of Information Systems and Computer Science, Singapore, 1997.

H. Liu, F. Hussain, C. Tan, and M. Dash, "Discretization: An Enabling Technique", *Data Mining and Knowledge Discovery*, 2002, pp. 393–423.

S. Monti, and G.F. Cooper, "A Multivariate Discretization Method for Learning Bayesian Networks from Mixed Data", *In Proceedings of 14th Conference of Uncertainty in AI*, 1998, pp. 404-413.

K. Kira, and L. Rendell, "A Practical Approach to Feature Selection", *In Proceedings of International Conference on Machine Learning*, Morgan Kaufmann, Aberdeen, 1992a, pp. 249-256.

K. Kira, and L. Rendell, "The Feature Selection Problem: Traditional Methods and New Algorithm", *In Proceedings of AAAI'92*. San Jose, CA, 1992b.

T.M. Mitchell, *Machine Learning*, McGraw-Hill Companies, Inc., 1997.

X.L. Zhu, *Fundamentals of Applied Information Theory*, Tsinghua University Press, 2000.

L. Breiman, "Technical Note: Some Properties of Splitting Criteria", *Machine Learning Vol. 24*, 1996, pp. 41–47.

M.A. Hall, and G. Holmes, "Benchmarking Attributes Selection Techniques for Discrete Class Data Mining", *IEEE Transactions on Knowledge and Data Engineering Vol. 15(3)*, 2003, pp. 1-16.

L.C. Molina, L. Belanche, and A. Nebot, "Feature Selection Algorithms: A Survey and Experimental Evaluation", *In Proceedings of the IEEE International Conference on Data Mining*, 2002.

M. Dash, and H. Liu, "Feature Selection for Classification", *Intelligent Data Analysis, Vol. 1(3)*, 1997, pp. 131-156.

K.J. Cios, W. Pedrycz, and R. Swiniarski, *Data Mining Methods for Knowledge Discovery*, Kluwer Academic Publisher, 1998.

G.H. John, R. Kohavi, and K. Pfleger, "Irrelevant Features and the Subset Selection Problem", *In Proceedings of the Eleventh International Conference on Machine learning*, 1994, pp. 121-129.

M.J. Pazzani, "Searching for Dependencies in Bayesian Classifiers", *In Proceedings of the Fifth International Workshop on AI and Statistics*, Springer-Verlag, 1996, pp. 424-429.

A. Jain, and D. Zongker, "Feature Selection: Evaluation, Application and Small Sample Performance", *IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 19(2),* 1997, pp. 153–158.

C.L. Blake, and C.J. Merz, "UCI Repository of Machine Learning Databases", Irvine, CA: University of California, Department of Information and Computer Science, 1998. [http://www.ics.uci.edu/~mlearn/MLRepository.html]

J.R. Quinlan, "Induction of Decision Trees", *Machine Learning Vol. 1(1)*, 1986, pp. 81-106.

J.R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, CA, 1993.

Quinlan, J. R. Improved use of continuous attributes in C4.5. Journal of Artificial Intelligence Research 4, 1996, 77-90.

P. Langley, W. Iba, and K. Thompsom, "An Analysis of Bayesian Classifiers", *In Proceedings of the tenth national conference on artificial intelligence*, AAAI Press and MIT Press, 1992, pp. 223-228.

R.C. Holte, "Very Simple Classification Rules Perform Well on Most Commonly Used Datasets", *Machine Learning Vol. 11*, 1993.

D. Hand, H. Mannila, and P. Smyth, *Principles of Data Mining*, The MIT Press, 2001.

N.K. Kasabov, *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering*, The MIT Press, Cambridge, Massachusetts London, England, 1998.

M. Melanie, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, Massachusetts, London, England, 1999.

I. Kononenko, "Estimating Attributes: Analysis and Extension of RELIEF", *In Proceedings of the European Conference on Machine Learning*, Springer-Verlag, Catania, Italy, Berlin, 1994, pp. 171-182.

H. Almualim, and T.G. Dietterich, "Learning with Many Irrelevant Features", *In Proccedinggs of the Ninth National Conference on Artifical Intelligence*, AAAI Press/The MIT Press, Anaheim, California, 1992, pp. 547-552.

J. Sheinvald, B. Dom, and W. Niblack, "A Modeling Approach to Feature Selection", *In Proceedings of the Tenth International Conference on Pattern Recognition*, 1990, pp. 535-539.