Computing and Informatics, Vol. 31, 2012, 1193–1216

CONDITIONAL DYNAMIC MUTUAL INFORMATION-BASED FEATURE SELECTION

Huawen Liu

Department of Computer Science
Zhejiang Normal University
Jinhua 321004, China
NCMIS, Academy of Mathematics and Systems Science
CAS, Beijing 100190, China
e-mail: hwhiu@zjnu.cn

Yuchang Mo, Jianmin Zhao

Department of Computer Science
Zhejiang Normal University
Jinhua 321004, China
e-mail: {myc, zjm}@zjnu.cn

Communicated by Patrick Brézillon

Abstract. With emergence of new techniques, data in many fields are getting larger and larger, especially in dimensionality aspect. The high dimensionality of data may pose great challenges to traditional learning algorithms. In fact, many of features in large volume of data are redundant and noisy. Their presence not only degrades the performance of learning algorithms, but also confuses end-users in the post-analysis process. Thus, it is necessary to eliminate irrelevant features from data before being fed into learning algorithms. Currently, many endeavors have been attempted in this field and many outstanding feature selection methods have been developed. Among different evaluation criteria, mutual information has also been widely used in feature selection because of its good capability of quantifying uncertainty of features in classification tasks. However, the mutual information estimated on the whole dataset can not exactly represent the correlation between features. To cope with this issue, in this paper we firstly re-estimate mutual information on identified instances dynamically, and then introduce a new feature selection method based on

conditional mutual information. Performance evaluations on sixteen UCI datasets show that our proposed method achieves comparable performance to other well-established feature selection algorithms in most cases.

Keywords: Pattern classification, feature selection, mutual information, data mining, pattern recognition

Mathematics Subject Classification 2010: 68T99

1 INTRODUCTION

As one of core components of pattern recognition or data mining, whose purpose is to assist users to elicit potential useful and hidden information from the rapidly growing volumes of data, classification exploits currently available knowledge or information to map future data into one of several pre-defined classes [12]. Generally speaking, the process of pattern classification can be fulfilled within two phases, i.e., the learning and predicting procedures. The purpose of learning is to discover patterns from training data so as to build classifiers, whereas predicting is the process of tagging new or unknown data with pre-specified class labels in terms of the induced classifiers. Note that an algorithm is said to have learning capability, if it can improve its capability of predicting or classifying by the known data.

A good learning algorithm must take into consideration the nature of data (e.g., sample size and dimensionality) [16]. If data is compact and non-redundant, the task of pattern learning and discovering will be getting easier and more efficient. On the other hand, a great variety of data will slow down the learning process significantly, and noisy data may worsen this situation further. An effective solution is to adopt sampling techniques (e.g., active sampling [35] and boosting sampling [22]) to reduce the amount of training data by identifying representative data. However, in reality many datasets only contain less instances (or samples), which are often characterized by hundreds or even thousands of features.

Theoretically, having more features implies more discriminative power in classification [16]. However, this is not always true in practical experience because in high dimensional datasets, many features are usually relevant to each other and some of them are redundant or useless. At this point, not all features are important for understanding or representing the underlying phenomena of interest [37]. The presence of redundant features may degrade the classification performance significantly and pull the efficiency of learning algorithms down if they were not properly excluded. In addition, the high dimensionality of data will also lead to over-fitting situation and even raise the so-called problem of "curse of dimensionality" [12]. To untie this knot, feature reduction (or dimensionality reduction) has been put forward.

Dimensionality reduction refers to the process of removing useless or insignificant features for class discrimination and retaining as more salient features as possible.

The preserved features must be competent for characterizing the main property of the original ones. Dimensionality reduction can bring lots of potential benefits to learning algorithms [17], such as reducing computational cost, improving prediction performance, avoiding model over-fitting, freeing from noises and providing a better understanding of the generated models. According to the manner of disposing features, dimensionality reduction techniques can be roughly categorized into feature extraction [37] and feature selection [22]. The former mainly constructs new features by projecting the original high-dimensional space into a low-dimensional one [37]. Although those features in the new space have better discriminative capability, the physical interpretation of data may be lost.

Unlike feature extraction, feature selection seeks an optimal subset from the original feature space to improve the quality of data. Due to its high efficiency and easy interpretation of the final results, feature selection has been studied extensively, and numerous outstanding feature selection algorithms have been addressed (see good surveys, e.g., [7, 15, 29, 34]). Generally, the feature selection algorithms can be grouped into three major categories [15, 34]: embedded, wrapper and filter models. The embedded methods mean that feature selection is integrated into the process of training for a given learning algorithm. For example, C4.5 [33] is a typical illustration of this kind. The wrappers, however, choose features with high priorities estimated by using a specified learning algorithm itself as part of the evaluation function [15]. As an example, Neumann et al. [30] integrated support vector machines (SVM) into feature selection to improve the performance of classifiers. Since the wrapper models are highly coupled with specified learning algorithms, they require much more training or learning time and have less general capability.

Conversely, the filter model evaluates the goodness of feature or subset on the basis of given criteria. Due to its computational efficiency, the filter method is very popular in solving high-dimensional data. It is noticeable that the given criteria play a vital role in the filter model. Liu and Yu [29] summarized them into four groups: distance, information, dependency and consistency measures [2]. The distance one takes the discrimination or divergence metric (e.g., Euclidean distance) between features as the class discriminative capability. Relief [24] is a representative one of this kind. In addition, Liang et al. [27] adopted inter- and intra-classes distances to score feature weight, while Zhang et al. [41] employed pairwise constraints (i.e., must- and cannot-link constraints) to evaluate feature goodness. The dependence metric mainly resorts to statistical correlations (e.g., Pearson's correlation coefficient and t-statistic test) to scale the correlation between features. However, the consistency metric determines the discriminative capability of features by virtue of the distribution of data [11]. Since these measures are all sensitive to the concrete values of the training data, they are less robust and easily affected by noises or outlier data.

The information criterion exploits information entropy to represent non-linear correlation between features. Since entropy is capable of quantifying the uncertainty of feature, the information criteria have attracted much attention and seem

to be widely studied in practice [28]. Currently, many outstanding selection algorithms based on different information criteria have been developed. As an example, mRMR [32] measures the relevance between features by mutual information and at each time the feature which has maximal relevance with the classes and minimal redundance with selected features will be selected. Hall [16] took symmetrical uncertainty as the criterion function to measure the correlation between discrete features, and then addressed a correlation-based feature selection method. Bonev et al. [6] estimated approximately Renyi entropy of features with the aid of entropic graph and then presented a filter feature selection approach. Several extensive experiments (e.g. [14, 19]) have also demonstrated that the information criteria work well in many cases.

Besides, other techniques have also been adopted by several feature selection algorithms in the literature. For instance, Hu et al. [18] took the sizes of the neighboring lower and upper approximations of decisions as the discriminating capability of feature subsets, and then utilized it to evaluate the significance of a subset of heterogeneous features. Similarly, Cornelis et al. [9] proposed a new selection method under the context of fuzzy rough set theory.

Usually, the values of information criteria are estimated on the whole sampling space in the traditional selection algorithms. This implies that once training data has been given, the values of mutual information of features are fixed throughout the whole selection process. It has been noticed that during the training procedure of classifiers, instances will be recognized or tagged with the pre-defined target labels in terms of selected features. Thus, the quantity of unrecognized instances is getting smaller and smaller. In other words, the weights of instances will be dynamically changed during the selection process, and the unrecognized instances are more important than the recognized ones in evaluating the interestingness of features. However, the invariable mutual information can not exactly represent this kind of correlation between features as the training procedure continues to work. Under this context, the values of mutual information should be re-estimated on those unrecognized instances, rather than the whole sampling space, along with the selection process [28]. In this paper, a new feature selection algorithm called CDMI is proposed. The difference of our method to DMIFS is that the evaluation criterion used here is conditional dynamic mutual information. The reason of adopting this criterion is that its value will be dynamically changed so as to exactly represent the correlation between features.

The main contributions of this paper are summarized as below. Firstly, the mutual information is adopted in our method, because it is a nonlinear one and competent for quantifying the degree of uncertainty between features. Moreover, it is independent of the assumption of data distribution. Secondly, the significant features are identified by using conditional mutual information, which can effectively measure the information amount of features after other features have been selected during the whole selection process. In addition, the number of features finally selected will be self-adaptively determined by this criterion, rather than pre-specified by users.

The structure of the rest is organized as follows. Section 2 presents some basic concepts about feature selection and mutual information. In Section 3, a new feature selection algorithm using conditional dynamic mutual information is proposed. Experimental results conducted to evaluate the usefulness and effectiveness of our approach is shown in Section 4. Section 5 briefly provides the state of the art about the feature selection algorithms based on information criteria. Finally, conclusions and future works are given in the end.

2 BASIC CONCEPTS

In this section, we firstly recall several basic concepts in information theory, and then give the formalism of feature selection under this context. More details can be consulted to good literatures (e.g. [7, 15, 29, 10]).

2.1 Mutual Information

Information entropy is one of fundamental concepts in information theory [10]. Unlike other metrics, which only consider linear correlation between features or variables, information entropy is a nonparametric and nonlinear one. Since it is theoretically capable of quantifying the amount of information, and no assumption about the distribution of data is made, information entropy and mutual information have been now widely used in feature selection. Before we delve into the details of feature selection algorithms, let us focus our attention on several basic concepts about information entropy.

Let X, Y and Z be three discrete random variables. The information *entropy* of X is represented as $H(X)$,

$$H(X) = -\sum_{x \in X} p(x) \log p(x), \tag{1}$$

where $p(x)$ is the marginal probability distribution of X. It is worth noting that information entropy $H(X)$ does not depend on the actual values of variable X, but only on its probability distribution. Hence, if X is a continuous random variable, its information entropy $H(X)$ can be taken as integral form, i.e.,

$$H(X) = -\int_x p(x) \log p(x) \mathrm{d}x. \tag{2}$$

For the sake of simplification, hereafter we only deal with random variables with finite discrete values, notwithstanding there are various approaches, such as Parzen window, histogram and kernel-based methods, to estimate the probability density function of continuous variable.

The *conditional entropy* $H(X|Y)$ of X with respect to Y mainly quantifies the remaining uncertainty of X when Y is known. It is denoted as

$$H(X|Y) = -\sum_{y\in Y}\sum_{x\in X} p(x,y)\log p(x|y), \tag{3}$$

where $p(x|y)$ is the conditional probability of X given the observing values of Y. From this equation, it can be observed that $H(X|Y)$ is zero as X is fully dependent on Y. This means that no more other information is required to describe X when Y is known. On the contrary, if they are mutually independent, $H(X|Y) = H(X)$ holds.

Mutual information is mainly used to describe how much information is shared between variables. Given two random variables X and Y, their *mutual information* $I(X;Y)$ is defined as

$$I(X;Y) = \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\frac{p(x,y)}{p(x)p(y)}. \tag{4}$$

This definition releases a signal that the higher $I(X;Y)$ is, the more relevancy between X and Y, and $I(X;Y) = 0$ implies they are totally unrelated with each other. According to Equations (2) and (3), we have $I(X;Y) = H(X) - H(X|Y)$.

In a similar vein, *conditional mutual information* is used to describe the amount of common information between two variables when other variables are known. Specifically, if the random variable Z has been given, the conditional mutual information of X and Y is represented as

$$I(X;Y|Z) = H(X|Z) - H(X|Y,Z). \tag{5}$$

This definition indicates that X brings information about Y which is not already contained in Z, and the larger the value of $I(X;Y|Z)$ is, the more information X has. Furthermore, according to the definitions, the following equation holds:

$$I(X;Y|Z) = I(X,Z;Y) - I(Z;Y). \tag{6}$$

That is to say, the conditional mutual information $I(X;Y|Z)$ denotes how much incremental amount of information can X bring to the given variables Z with respect to Y. This is very important and especially useful in feature selection.

2.2 Feature Selection

Assume that C and $\mathcal{D}$ denote the class labels and instances represented by features $\mathcal{F} = \{f_1, \ldots, f_m\}$, respectively. Given a dataset $\mathcal{T} = (\mathcal{D}, \mathcal{F}, C)$, the process of learning aims to construct a hypothesis $h : dom(f_1) \times \ldots \times dom(f_m) \to C$ (i.e., classifier), where $dom(f_i)$ is the domain of $f_i \in \mathcal{F}$. Due to the limitation of data, the induced hypothesis on $\mathcal{D}$ may lead to a probability error $\epsilon_{\mathcal{F}}(h)$ in classifying

new data. Generally speaking, the task of feature selection is to select a minimum optimal feature subset S from original feature space $\mathcal{F}$ such that the representative power of S is as high as possible, i.e., $\epsilon_S(h) \approx \epsilon_{\mathcal{F}}(h)$.

From the view of information theory, the process of constructing classifier is to minimize the uncertainty of the known observations S regarding the class labels C. This, however, is inherently consistent with mutual information $I(C;S)$, which contains important dependence information between the selected subset S and the labels C. Intuitively, mutual information $I(C;S)$ can be utilized to evaluate the goodness of a feature subset S. Under this context, if the mutual information $I(C;S)$ of features S with C is very small, S are irrelevant to C and they contribute nothing to the distribution of the classes. Consequently, feature selection is to achieve the highest possible value of $J(S) = I(C;S)$ with the smallest possible size of S.

To obtain an optimal subset S, the brute force way is to systematically examine all feature subsets S of $\mathcal{F}$ and find out the optimal one which has the least features and the largest $J(S)$. Note that the computational cost of this method is relatively high. Even a moderate size of $\mathcal{F}$, it is still computationally intractable, because there are 2^m feature subsets. Meanwhile, estimating $J(S)$ is also a hard work and its value is incredible. To alleviate this cumbersome issue, many heuristic subset search or selection strategies, such as branch and bound search, beam search, probabilistic search and random search, have been addressed [36]. The commonly adopted way in practice is the sequential forward selection and monotone property [29], whereas only individual feature f must be calculated $J(f)$ at each iterative procedure of feature selection, rather than the whole subset S.

3 THE PROPOSED METHOD

As mentioned in the previous section, the common goal of feature selection algorithms is to maximize the relevance between selected features and the labels, and to minimize the redundancy among the already selected features at the same time. This indicates that the information loss incurred by the selecting process is as little as possible, and the classification performance will not deteriorated greatly after feature selection has been performed.

To describe the relevance of features, several definitions have been proposed in the literature. The most comprehensively-studied definition in machine learning is that introduced by John et al. [23]. The intuitional meaning of their definition is that given a feature subset S, feature $s \in S$ is strongly relevant to the labels C if the posterior distribution of C regarding to S is changed after s has been discarded from S. According to this definition, irrelevant features can never contribute to prediction performance. Besides, mutual information is also competent for scaling the relevance between features effectively, and this has been demonstrated by several simulation experiments (e.g. [14, 19]).

In order to obtain the relevance between feature f and the classes C, both the probability and mutual information methods must firstly estimate the probability distribution of f with respect to C on the training data. However, once the training dataset has been given, the probability distributions of features on this sampling space will not be altered any more. This may raise a problem that these invariable probabilities cannot exactly represent the information, which is provided by candidate features, to unrecognized instances when the number of the combined feature values increases. As a result, "false" relevant information between features will be obtained, which leads to feature selection procedure with many trivial details. To better understand this idea, let us consider an example as follows.

Example 1. Given a training dataset $\mathcal{T} = (\mathcal{D}, \mathcal{F}, C)$ (Table 1), where $\mathcal{D} = \{1, \ldots, 6\}$, $\mathcal{F} = \{f_1, f_2, f_3\}$. $\mathcal{F}$ and C divide $\mathcal{D}$ into four different partitions: $P/f_1 = \{\{1234\}, \{56\}\}$, $P/f_2 = \{\{234\}, \{15\}, \{6\}\}$, $P/f_3 = \{\{1246\}, \{35\}\}$ and $P/C = \{\{14\}, \{2\}, \{35\}, \{6\}\}$. According to the definition of mutual information, we have $I(C; f_1) = 0.58$, $I(C; f_2) = 0.79$ and $I(C; f_3) = 0.92$. Thus, the ranking of $\mathcal{F}$ is f_3, f_2, f_1 in the light of their values, and this is the final selection result of BIF [22]. However, it is noticeable that after f_3 has been chosen firstly, the 3^{th} and 5^{th} instances can be identified or recognized, that is, their information has already been embedded in f_3. If we remove them from the partitions P/f_1, P/f_2 and P/C, the new values of mutual information $I(C, f_1)$ and $I(C, f_2)$ are 0.81 and 0.50, respectively. In this case, f_1 has more information than f_2 with respect to the classes C. Hence, it is more preferable to f_2. Indeed, those recognized instances usually result in redundant interactions between features. Once they have been discarded from the dataset D, redundant information would also be cut down at the same time.

No	f_1	f_2	f_3	C
1	v_{11}	v_{21}	v_{31}	l_1
2	v_{11}	v_{22}	v_{31}	l_2
3	v_{11}	v_{22}	v_{32}	l_3
4	v_{11}	v_{22}	v_{31}	l_1
5	v_{12}	v_{21}	v_{32}	l_3
6	v_{12}	v_{23}	v_{31}	l_4

Table 1. A dataset $\mathcal{T} = (\mathcal{D}, \mathcal{F}, C)$ in the example

From the view of classification learning, instances in the sampling space $\mathcal{T} = (\mathcal{D}, \mathcal{F}, C)$ can be exclusively classified into two disjoint partitions: recognized D_l and unrecognized D_u, where $\mathcal{D} = D_u \cup D_l$ and $D_u \cap D_l \emptyset$. Along with the process of learning, unrecognized instances D_u will be continuously identified with a pre-defined class label in C in the light of available knowledge embodied by selected features. This learning procedure will not be terminated, unless unrecognized instances have been identified or recognized. That is to say, the stopping condition is that the amount of information owned by selected features is approximately equal to those of original features.

Assume S is the already selected subset of relevant features and $\mathcal{D}$ has been divided into D_l and D_u partitions. The next step of feature selection is to identify a good feature f out of the candidate features F. Usually, the feature with the largest mutual information estimated on $\mathcal{D}$ will be selected, where each instance has the same weight. However, not all instances have the same importance in evaluating the goodness of features. Indeed, the learning procedure will place more emphasis on unrecognized instances D_u, and for recognized instances D_l, we have the following property.

Property 1. Let F be the set of candidate features. For recognized instances D_l, any feature $f \in F$ is irrelevant to the labels C, namely, $I(f;C) = 0$.

This property can be proved in a straightforward way because D_l can be completely classified or recognized by classifier with the selected features S, that is, S contains the information describing D_l. It reveals an important fact that initial mutual information $I(C;f)$ or its conditional one, which estimated on the whole instances $\mathcal{D}$, can not exactly represent the relevance between f and C. Therefore, they are not appropriate to acting the role of accurate metric any more, unless they are re-estimated on D_u.

Based on this analysis, here we propose a new feature selection algorithm using conditional dynamic mutual information, where the mutual information is dynamically estimated on D_u, not in the whole space $\mathcal{D}$. Thus, the next selected feature f is capable of recognizing as many instances in D_u as possible. To obtain accurate value of conditional mutual information of candidate features, new recognized instances induced by already selected feature S must be kept down. These new recognized instances will be removed from D_u, before $I(C;f|S)$ is re-estimated. Except these two steps, others are the same as those of the common schema of feature selection methods. More specifically, the details of our algorithm (CDMI) are shown in the following.

Algorithm 1 (CDMI) Feature selection via conditional dynamic mutual information

Input: A training data set $\mathcal{T} = (\mathcal{D}, \mathcal{F}, C)$;
Output: A selected feature subset S;
1). Initialize relative parameters, e.g., $S = \emptyset$, $D_l = \emptyset$, $F = \mathcal{F}$ and $D_u = \mathcal{D}$;
2). **For** each feature $f \in F$ **do**
3). Calculate $I(C;f|S)$ on D_u;
4). **If** $I(C;f|S) = 0$ **then** Remove feature f from F;
5). Choose the feature f with the largest $I(C;f|S)$;
6). Remove it from F and insert it into the selected feature subset S;
7). Obtain new recognized instances D_l induced by f;
8). Remove instances in D_l from unrecognized instances D_u;
9). **If** $F \neq \emptyset$ and $D_u \neq \emptyset$ **then goto** 2);
10). Return S as the selected feature set;

As illustrated in Algorithm 1, our selection algorithm works in a straightforward way. At the beginning, it estimates the conditional mutual information $I(C;f|S)$ for

each candidate feature $f \in F$ with the labels C on the unrecognized instances D_u, and then eliminates those candidate features that do not contribute to the classification issue. After that, the candidate feature f with the highest priority is chosen according to its conditional mutual information with regard to selected features. Further, new recognized instances induced by the just selected feature f are obtained and saved to D_l. Since D_l will not bring benefits to the succeeding selection processes, they will be thrown away from D_u directly. The purpose of this step is to prevent them from being recalculated in estimating $I(C; f|S)$. After doing this, the algorithm will go to next iteration to pick up other candidate features if there still are unrecognized instances in D_u or available candidate features in F.

In real world, not all datasets are consistent. In this case, some constraints, such as $|D_u| \leq \delta$, can be directly added into the 9th step to achieve better results. Given a training dataset $\mathcal{T} = (\mathcal{D}, \mathcal{F}, C)$, the quantities of instances in $\mathcal{D}$ and features $\mathcal{F}$ are finite. Moreover, the number of features decreases by one at each repetition, notwithstanding new recognized instances D_l may be empty and D_u may not be reduced. Therefore, the proposed selection algorithm will be terminated at last. Since the most expensive overhead of computation procedure lies in Step 3), which takes $O(n)$ to estimate $I(C; f|S)$ for every feature f, the total computational cost of CDMI is $O(nm^2)$.

4 SIMULATION EXPERIMENTS

This section presents comparative experiments on 16 UCI datasets to validate the effectiveness and performance of our proposed method by comparing with other popular feature selection algorithms.

4.1 Benchmark Datasets

To better evaluate the performance of CDMI, several experiments have been carried out on 16 benchmark datasets with different types and sizes. All these datasets are widely used in the data mining community to evaluate the learning and selection algorithms, and available from the UCI Machine Learning Repository [3]. Full documentation of original datasets can be obtained from the UCI website. Table 2 gives some general information about these datasets. These datasets comprise a diverse mixture of feature types (continuous and nominal). Additionally, the quantities of data in these datasets vary from 76 to 8124, and the highest dimensionality of the original datasets is up to 1558. To some extent, they can provide a comprehensive test for feature selection methods under different conditions.

Since some features in datasets are too trivial to the suitable for classification, we omitted them during the preprocessing stage. For example, the *timestamp*, *cylinder-number* and *customer* features in the *Cylinder-bands* dataset were excluded in classification. Additionally, the *name* feature in the *Sponge* dataset is also too trivial. In a similar vein, for *Spectrometer*, we left two features, i.e., *LRS-name* and

No	*Datasets*	*#instances*	*#features*	*#classes*
1	Anneal	898	38	6
2	Cylinder-bands	540	36	2
3	Dermatology	366	34	6
4	Hypothyroid	3 772	29	4
5	Internet advertisers	3 279	1 558	2
6	Ionosphere	355	34	2
7	KDD synthetic control	600	60	6
8	Kr-vs-kp	3 196	36	2
9	Lymph	148	18	4
10	Mfeat-pixel	2 000	240	10
11	Mfeat-zernike	2 000	47	10
12	Mushroom	8124	22	2
13	Musk clean1	476	166	2
14	Musk clean2	6 598	166	2
15	Spectrometer	531	100	4
16	Sponge	76	44	3

Table 2. Descriptions of 16 datasets in our experiments

LRS-class, out of consideration in further experiments. Moreover, the last feature in each dataset was taken as the classification label, except the *Spectrometer* dataset, where the classification label is *ID-Type*.

Before evaluating the performance of feature selectors, preprocessing is necessary for the purpose of achieving fair classification results. For example, all missing values rooting from various aspects in these datasets were replaced with the most frequent values (or means) for nominal (or numeric) features, respectively. Additionally, it is always difficult to obtain the probability density function of continuous feature with limited number of data in estimating the conditional mutual information. Thus, continuous features were discretized into nominal ones by minimum description length technique for the convenience of simplification.

4.2 Experimental Setting

By now, many outstanding feature selectors based on information criteria have been proposed. For instance, MIFS [4] is a classical feature selection algorithm using mutual information, while MIFS-U [25] and mMIFS-U [31] developed recently are its modified versions. Here we made a comparison between them and our proposed algorithm. The parameter β in MIFS and MIFS-U was assigned to $1/|S|$. Additionally, we also took the evaluation criterion of symmetrical uncertainty (SU) [40] as the baseline, because it is a normalized form of mutual information. The search strategy in all feature selectors is the sequential forward selection and the initial selected subset is empty.

It can be observed that these four selectors are all filter ones and independent of learning algorithms. Thus, external learning algorithms are required to evaluate

their performances. Although many outstanding classifiers are available to fulfill experimental purpose, in our experiments, three popular classifiers were chosen to test the capability of selection methods. They are 1-NN [1], C4.5 [33] and Bagging [8], where the built-in classifier of Bagging is REPTree which is also a well known learning algorithm in the machine learning community. The reason of choosing them is that they represent three quite different types of learning approaches and they are relatively fast.

To achieve impartial results, the same quantity of features was chosen for each feature selector and the selected features were arranged in a descending order in terms of their priorities. Moreover, three ten-fold cross validations had been adopted for each classifier-dataset combination in verifying classification capability and its average values were the final results. For each classifier, statistic t-test between selectors and the original performance was carried out. Its purpose is to determine whether the corresponding feature selection method can significantly improve the performance of classifier or not. Throughout this paper, the difference is considered to be significantly different if its corresponding p-value is less than 0.05 (i.e., confidence level greater than 95 %). All experiments were conducted under the platform of Weka [39], which brings together many machine learning algorithms into a common framework.

4.3 Experimental Results

4.3.1 Individual Classifier

The results of classification performance of individual classifier induced by five feature selectors on the datasets are presented in Tables 3, 4 and 5, where the *Full* column in each table denotes the performance of the corresponding classifier over the original datasets without using any selectors. In these tables, '$\circ$' (or '$\bullet$') is used to illustrate that the accurate rate with corresponding selector is significantly better (or worse) than those without using feature selector (i.e., the *Full* column in the same classifier) in the statistical t-test. The bold value refers to the largest one among five feature selection methods in the same classifier. Average performances of classifier with different feature selectors are given in the *Average* row.

From the experimental results in Table 3, one can observe that the proposed method is superior to other four selectors in all aspects. For example, the number of the highest values of CDMI is ten, which is larger than others. Meanwhile, there is no dataset whose performance was significantly worse by our proposed method. However, for other four feature selectors, there is more than one dataset whose performance was degraded significantly. The situation is worse for the MIFS selector, where the number of worse cases is eight. From the view of the *Average* performance, the value induced by CDMI is also the largest one among these five feature selectors.

In the C4.5 classifier (Table 4), the predominance of CDMI is still distinct. There are ten over 16 datasets whose performances induced by CDMI are the highest ones among selectors, and only on the *Sponge* dataset (i.e., No. 16) CDMI significantly

No	*Full*	*CDMI*	*mMIFS-U*	*MIFS-U*	*MIFS*	*SU*
1	93.32	**93.39**	93.20	**93.39**	81.49•	**93.39**
2	76.30	**76.25**	72.85	73.72	59.24•	72.85
3	95.81	91.05	90.34	83.20•	**91.22**	78.35•
4	97.93	97.71	**98.13**	97.47	97.30	97.51
5	97.26	**97.17**	96.75	95.46	93.98•	95.55
6	92.88	**92.48**	91.52	89.33	85.43•	90.48
7	98.00	**94.16**	76.91•	78.14•	87.26	77.91•
8	90.37	92.34	**93.56**○	91.39	88.83	91.39
9	80.35	**81.89**	80.03	80.03	74.19•	80.76
10	96.22	73.74	82.84	57.96•	**83.54**	50.36•
11	70.58	62.25	63.85	56.56•	**63.55**	56.56•
12	100.00	**100.00**	99.35	99.35	98.62•	99.90
13	89.72	**86.26**	86.16	80.64	74.04•	81.55
14	94.49	94.33	94.35	**94.40**	92.01•	93.74
15	59.82	**63.77**	59.31	57.80	63.02	60.44
16	92.32	**94.35**	93.51	93.51	93.99	89.40
Average	89.60	**86.21**	85.39	82.46	82.34	81.44

Table 3. A comparison of performances of five feature selectors in the 1-NN classifier
[1]Notation '○' (or '•') indicates the performance induced by selector is significantly better (or worse) than the *Full*.
[2]The bold value is the highest one among five feature selection methods.

No	*Full*	*CDMI*	*mMIFS-U*	*MIFS-U*	*MIFS*	*SU*
1	92.28	92.49	**92.60**	92.49	86.69•	92.49
2	71.30	71.67	70.37	**71.85**	66.85•	70.37
3	94.19	**93.53**	92.89	79.74•	92.43	78.28•
4	99.27	99.37	99.19	99.14	97.99•	**99.40**
5	96.83	**97.20**	**97.20**	96.98	96.14	96.98
6	89.36	91.14	92.00○	91.62○	90.76	**92.57**○
7	93.00	**92.54**	77.19•	78.02•	87.15	77.30•
8	99.44	**99.16**	97.95	96.98•	96.57•	96.98•
9	78.59	**79.35**	73.51	73.51	73.29	72.59•
10	78.57	72.42	73.10	61.53•	**75.82**	57.35•
11	65.35	**64.78**	62.86	59.43•	64.00	59.43•
12	100.00	**100.00**	99.41	99.41	98.62	99.90
13	87.76	**85.12**	84.91	81.12•	78.38•	84.70
14	96.63	**95.51**	94.84	95.03	93.62•	94.85
15	66.04	67.23	69.62○	67.74	**70.94**○	67.61
16	92.50	**92.32**•	**92.32**•	**92.32**•	**92.32**•	**92.32**•
Average	86.44	**86.08**	84.80	82.78	84.11	82.34

Table 4. A comparison of performances of five feature selectors in the C4.5 classifier

degraded the performance of the classifier. Even so, its accuracy is equal to those of other selectors, and slightly lower than the original one. For the remaining selection algorithms, the significantly worse cases range from two to seven, and the number of the highest values varies from two to five. In a similar vein, the mean performance of CDMI is larger than others.

No	*Full*	*CDMI*	*mMIFS-U*	*MIFS-U*	*MIFS*	*SU*
1	93.28	**93.23**	**93.23**	**93.23**	86.28•	**93.23**
2	71.30	**72.04**	71.92	71.98	66.97•	71.92
3	92.71	91.23	**91.51**	81.83	90.68	78.00•
4	99.08	99.08	**99.09**	99.08	97.85•	**99.09**
5	96.68	96.84	**96.93**	96.90	95.91•	96.91
6	91.55	92.19	**92.57**	91.90	90.19	91.62
7	95.28	**93.61**	79.47•	79.80•	88.65	80.24•
8	99.04	**99.01**	97.76	97.01	96.34•	97.01
9	77.48	77.57	77.81	77.81	73.73•	**78.68**
10	84.45	73.07	76.20	62.28•	**78.31**	56.86•
11	66.82	64.85	**66.08**	61.16•	66.00	61.16•
12	100.0	**100.0**	99.43	99.43	98.62•	99.90
13	87.70	**87.22**	86.87	82.94	79.36•	84.20
14	96.01	**95.62**	95.31	95.21	93.41•	95.19
15	70.31	**72.39**	72.14	68.18	71.76	67.86
16	92.50	92.32	92.32	92.32	92.32	**92.74**
Average	87.19	**86.55**	86.02	83.80	84.58	83.26

Table 5. A comparison of performances of five feature selectors in the Bagging classifier

According to the experimental results given in Table 5, the capability of CDMI is slightly better than that of the mMIFS-U selector. CDMI achieved eight highest values, while the remaining selectors had no more than six highest values. Although there are eight cases whose performances induced by CDMI are not the highest ones, they are also not the worst ones among these selection algorithms. Similar to the C4.5 and 1-NN classifiers, other three selectors in the Bagging classifier are inferior to CDMI in all facets. Note that none of these five feature selectors have strengthened the performance of the classifier significantly.

Apart from comparing with the *Full* column, we also make a comparison between CDMI and other selectors at the aspect of Win/Tie/Loss. To serve for this purpose, we compared their classification accuracies between CDMI and others on these 16 datasets. The comparison results are given in Table 6. The data indicates that our proposed selector still outperforms other four selection methods in most cases. As an example, the entry '10/2/4' of the first column and the first row in Table 6 denotes that CDMI wins ten over sixteen cases, while loses four in comparison with mMIFS-U on the classification performance of C4.5. For the remaining two datasets (i.e., Nos. 5 and 16), both CDMI and mMIFS-U have the same accurate rates.

Classifier	mMIFS-U	MIFS-U	MIFS	SU
C4.5	10/2/4	11/2/3	13/1/2	11/2/3
1-NN	11/0/5	14/1/1	13/0/3	15/1/0
Bagging	7/2/7	11/3/2	13/1/2	11/1/4

Table 6. A comparison of Win/Tie/Loss between CDMI and other selectors

4.3.2 Average Performance

In order to depict performance on the whole, we averaged classification accuracies of three classifiers, and made a comparison between the mean performance induced by the feature selectors and those without using the feature selectors. The results are presented as a bar graph (Figure 1). A bar above the zero line implies that the corresponding selector has improved classification performance; Otherwise the classification capability has been deteriorated. Moreover, the result is significantly better or worse than the original performance, i.e., without using the feature selectors, at 95 % confidence level, if the length of bar is larger than two.

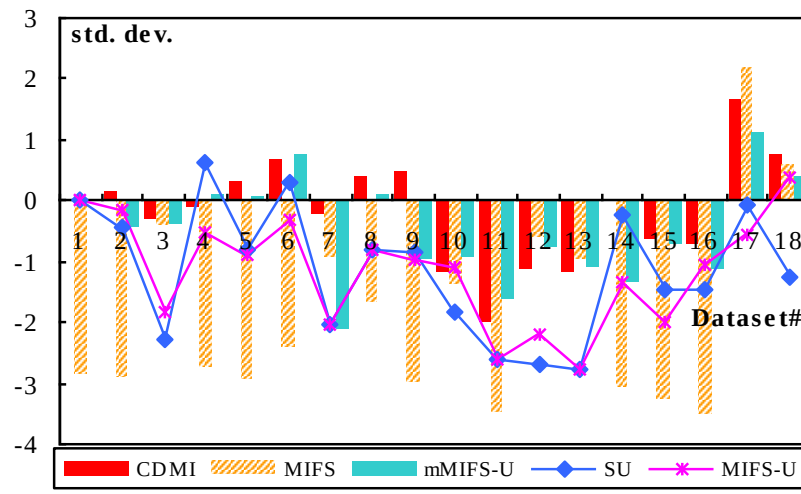


Fig. 1. p-value of mean performance between individual selector and original classifier. (Bars below the zero line in the figure indicate that the corresponding feature selector has degraded classification performance.)

Figure 1 also tells us that CDMI outperforms others on the mean performance. For instance, the numbers of datasets with better and worse performance are eight and seven, respectively. In addition, among these seven cases, there is none dataset whose performance had been significantly degraded. For other four feature selectors, however, the quantities of worse situations are all greater than seven. It may be

noticed that MIFS greatly improve classification performance on the *Spectrometer* dataset (No. 15 in Figure 1). Unfortunately, it significantly deteriorated performances of classifiers on nine over sixteen datasets.

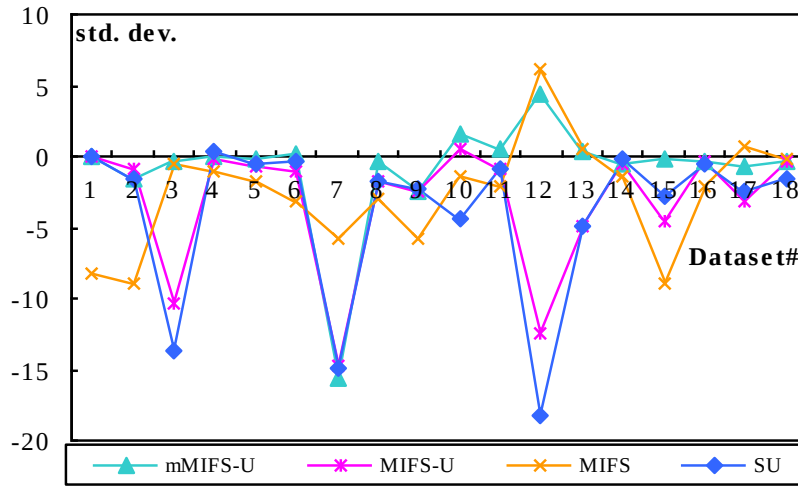


Fig. 2. p-value of mean performance between CDMI and other four feature selectors (bars below zero mean that the corresponding selector is inferior to the CDMI)

The comparison of the mean performance between CDMI and four feature selectors is also made and the results are provided in Figure 2. Similarly, bars above the zero line indicate that the corresponding selector surpasses our method on the current dataset. As shown in the graph, there are at most three over sixteen cases, where the mean performance by other selectors (e.g., mMIFS-U and MIFS) is better than CDMI. However, the mean performance induced by MIFS on nine datasets is significantly lower than that of CDMI. This implies that the CDMI selector is superior to other four selectors in most cases.

4.3.3 Performance of Selected Features

For the purpose of characterizing the impact of our method on individual selected features, another group experiment has been carried out on four datasets (i.e., *Internet advertisement*, *Kr-vs-kp*, *Musk clean1* and *Spectrometer*) by using the three classifiers with different feature selectors. The experimental mode is the same as before, i.e., three times of 10-fold cross validation. The experimental results are shown in Figures 3–6, where the classification accuracy is the average value of three classifiers over three times.

It can be observed in Figure 3 that in most cases CDMI is comparable to other four feature selectors. For example, all plots of CDMI are higher than those of

other selectors, except only two spots are lower than the SU selector in *Internet advertisement* (Figure 3). Analogous situations can be found in the *Kr-vs-kp* (Figure 4) and *Musk clearn1* datasets (Figure 5). This indicates that CDMI is capable of choosing informative features at each selection stage. For the case of *Spectrometer* (Figure 6); however, the performance achieved by CDMI is higher than that of the MIFS-U and SU selectors, while being lower than MIFS and mMIFS-U on the last several features. The main reason behind it perhaps is that this dataset contains so many noises, so as to the condition dynamic mutual information can not exactly represent the relevance between features, after most instances have been recognized and removed from the original dataset. As a matter of fact, the inconsistent rate of *Spectrometer* is very high.

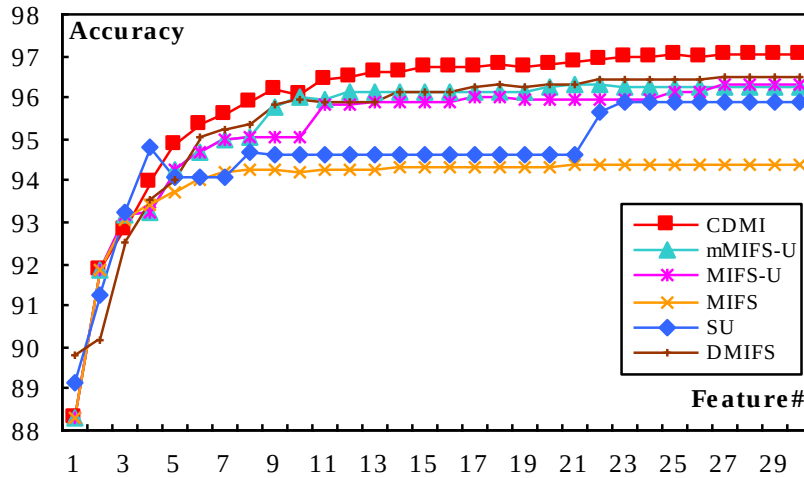


Fig. 3. Accuracy vs. different numbers of selected features on *Internet advertisement*

5 RELATED WORK

During past years, a modest number of feature selection algorithms based on mutual information (MIFSA) have been addressed. In this section, we will briefly review the state of the art about MIFSA. For convenience, $S \subseteq \mathcal{F}$ and $F \subseteq \mathcal{F}$ represent selected and candidate feature subsets, respectively. Correspondingly, $f \in F$ and $s \in S$ are candidate and selected features, respectively.

Currently, most MIFSAs adopt the monotone property in estimating evaluation criterion $J(S)$, that is, the subset search strategy is the sequential forward selection [28], which is incremental and greedy one. It begins with an empty set of selected features and each time the candidate feature f with the most positive

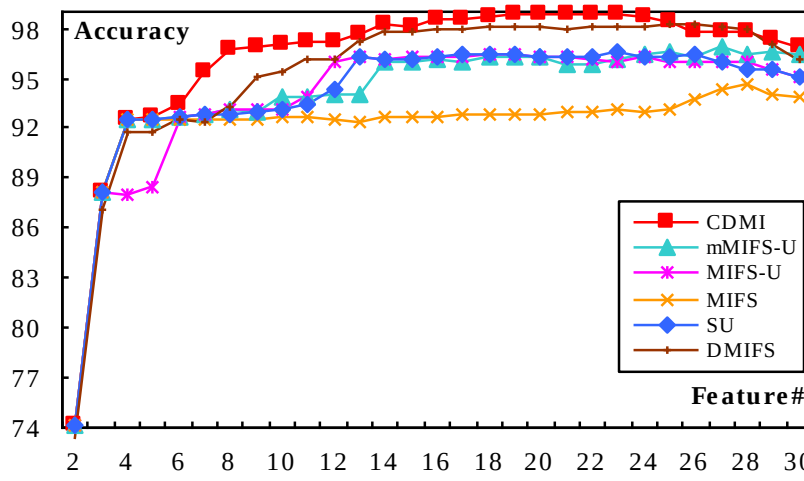


Fig. 4. Accuracy vs. different numbers of selected features on *Kr-vs-kp*

influence on the criterion, i.e., $J(f)$, will be chosen. This iteration procedure will be terminated when either the number of selected features is larger than a pre-specified threshold or information amount of S has not been improved by adding one more feature. The difference among such selection methods lies in their different criterion function $J(f)$.

The most naïve evaluation criterion of MIFSA is perhaps $J(f) = I(C; f)$, which is also known as best individual features (BIF) [22]. BIF evaluates features individually by virtue of this criterion, sorts them in descending order and then picks out the best k features. Despite BIF is highly efficient, it does not involve interactions among features. Moreover, selecting features individually may not lead to an optimal solution. To cope with this problem, Battiti summed the relevance (i.e., $I(f; s)$) in MIFS [4] between candidate feature f and selected feature s to penalize $I(C; f)$, that is, the criterion function of MIFS is $J(f) = I(C; f)\text{-}\beta \sum_{s \in S} I(s; f)$, where $0.5 \leq \beta \leq 1$. Since the parameter β in MIFS is hard to be regulated, Peng et al. [32] assigned it with a fixed value $1/|S|$, and then chose salient features by wrapping a learning algorithm.

Kwak and Choi [25] argued that the penalized operator in MIFS does not consider the aspect of the relevance between s and C. Thus they added $I(C; s)$ into the penalized operator in their evaluation criterion of MIFS-U. Like mRMR, Huang et al. [20] set the parameter β in MIFS-U with the same value and then trained the selector with genetic algorithm and support vector machine to achieve optimal results. Recently, Novovičová et al. [31] replaced the sum operation with a maximum one in mMIFS-U to further improve performance, that is, their criterion is

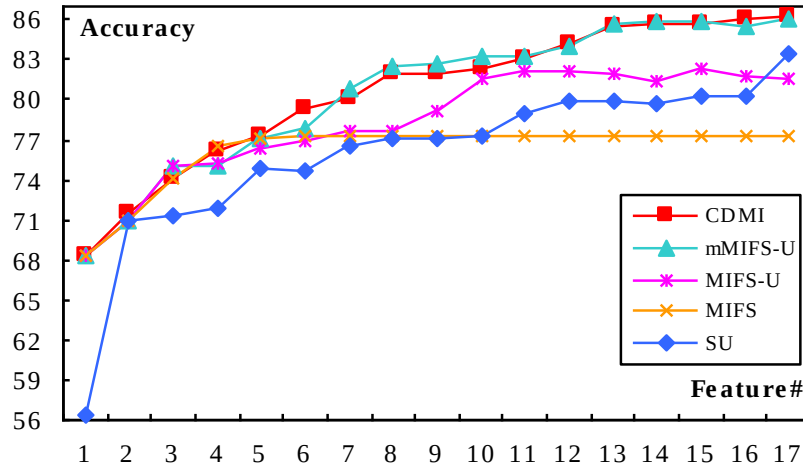


Fig. 5. Accuracy vs. different numbers of selected features on *Musk clean1*

$J(f) = I(C;f) - \max(I(f,s)I(C,s)/H(s))$. The advantage is that it is free from parameter harassment.

Beside the mutual information, other information metrics have also been adopted in the feature selection algorithms. As a typical example, Yu and Liu in FCBF [40] focused on *symmetrical uncertainty* (shortly, SU) to represent the information correlation between f and C, which is defined as $SU(C,f) = 2I(C;f)/[H(f) + H(C)]$. In addition, they also took use of approximate Markov blanket technique to eliminate redundant features. Bell and Wang [5] adopt *unconditional variable relevance* $r(C;f,S)$, where $r(C;f,S) = I(C;f,S)/H(S,f)$, as their criterion function in evaluating features. From its definition, one may observe that symmetrical uncertainty and unconditional variable relevance are all normalized forms of mutual information.

As mentioned above, the conditional mutual information $I(C;f|S)$ represents the relevant degree between f and C when the information of the selected features S is known. Naturally, it can also be used as the evaluation criterion of features. However, the estimation of $I(C;f|S)$ on the whole space S is intractable. To alleviate this dilemma, both Fleuret [13] and Wang et al. [38] substituted S with a single selected feature $s \in S$. They argued that the candidate feature f is good enough only if $I(C;f|s)$ is large for every selected feature $s \in S$. Thus, in their methods, the criterion function is $J(f) = \max_f I(C;f|s)$, where s is the selected feature whose $I(C;f|s)$ is minimal. Levi and Ullman in [26] did a similar work. They replaced the selected subset with a single selected feature, whose mutual information with the classes is minimal, in estimating $I(C;f,S)$. To estimate the value of $I(C;f,S)$, researchers resort to approximate or heuristic methods, e.g., histogram, Parzen win-

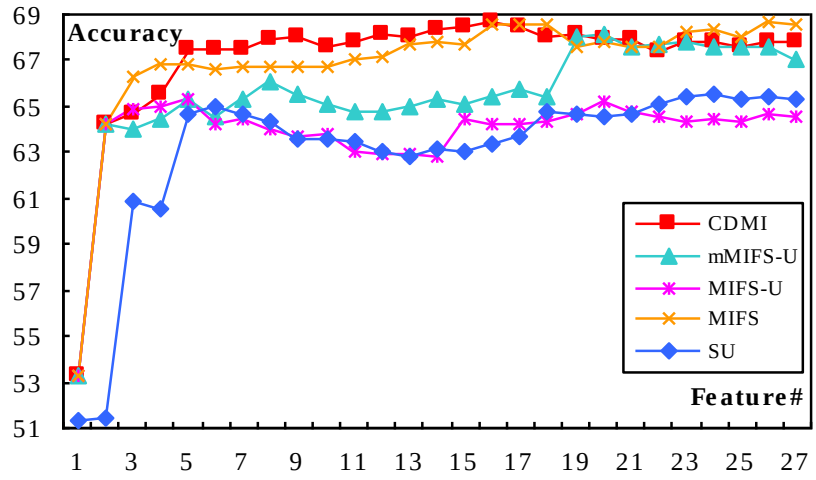


Fig. 6. Accuracy vs. different numbers of selected features on *Spectrometer*

dow [25], Gaussian kernel function [21] and entropic graph [6]. It is noticeable that dynamic mutual information has also been used to measure the interestingness of feature in DMIFS [28]. However, the difference between DMIFS and our method is that the metric here is a conditional one. Table 7 summaries most information evaluation functions adopted in the popular feature selection algorithms.

Name	Source	Information function $J(f)$
BIF	Jain et al. [22]	$I(f;C)$
MIFS	Battiti [4]	$I(C;f) - \beta \sum I(f;s)$
CR	Bell and Wang [5]	$r(S,f;C) = I(S,f;C)/H(S,f)$
mRMR	Peng et al. [32]	$I(C;f) - 1/\|S\| \cdot \sum I(f;s)$
MIFS-U	Kwak and Choi [25]	$I(C;f) - \beta \sum r(s;f) \cdot I(C;s)$
mMIFS-U	Novovičová et al. [31]	$I(C;f) - \max(r(s;f) \cdot I(C;s))$
FCBF	Yu and Liu [40]	$SU(f,C) = 2I(f;C)/(H(f) + H(C))$
CMIM	Fleuret [13]	$\max_f I(C;f\|s)$
MV	Levi and Ullman [26]	$\max_f I(C;f\|s)$

Table 7. Information criteria in most feature selection algorithms

6 CONCLUSIONS

Feature selection plays a unique role in pattern analysis and information processing. In this paper, we developed a new feature selection algorithm, whose basis is conditional dynamic mutual information. Unlike other selection methods based

on mutual information, our method estimates mutual information dynamically on unrecognized instances, not the whole sampling space. The rationale behind it is that candidate features are irrelevant or redundant for classification regarding to recognized instances. Thus, the dynamic mutual information can exactly measure the relevance between features along with selection procedure. Experimental results on 16 UCI datasets show that the proposed method works well and outperforms other classical feature selectors in most cases.

Since mutual information will be re-calculated after one feature has been chosen, it inevitably requires much more time than others. In addition, our method is sensitive to noise data, and the performance may be poor if dataset contains many noises. Therefore, our future work will be at handling these issues by using heuristic tactics.

Acknowledgement

The authors are grateful to the anonymous referees for their valuable comments and suggestions. This work is supported by the National NSF of China (61100119, 61170108, 61170109, 61272130, 61272468), Postdoctoral Science Foundation of China, Key Discipline of Computer Science and Theory of Zhejiang Province (ZSDZZZZ-XK05).

REFERENCES

[1] Aha, D.—Kibler, D.: Instance-Based Learning Algorithms. Machine Learning, Vol. 6, 1991, pp. 37–66.

[2] Arauzo-Azofra, A.—Benitez, J. M.—Castro, J. L.: Consistency Measures for Feature Selection. Journal of Intelligence Information System, Vol. 30, 2008, pp. 273–292.

[3] Asuncion, A.—Newman, D. J.: UCI Repository of Machine Learning Databases. Available on `http://www.ics.uci.edu/~mlearn/MLRepository.html`, Department of Information and Computer Science, University of California, Irvine, 2007.

[4] Battiti, R.: Using Mutual Information for Selecting Features in Supervised Neural Net Learning. IEEE Transactions on Neural Networks, Vol. 5, 1994, No. 4, pp. 537–550.

[5] Bell, D. A.—Wang, H.: A Formalism for Relevance and Its Application in Feature Subset Selection. Machine Learning, Vol. 41, 2000, pp. 175–195.

[6] Bonev, B.—Escolano, F.—Cazorla, M.: Feature Selection, Mutual Information, and the Classification of High-Dimensional Patterns. Pattern Analysis and Application, Vol. 11, 2008, pp. 309–319.

[7] Blum, A. L.—Langley, P.: Selection of Relevant Features and Examples in Machine Learning. Artificial Intelligence, Vol. 97, 1997, pp. 245–271.

[8] Breiman, L.: Bagging Predictors. Machine Learning, Vol. 24, 1996, No. 2, pp. 123–140.

[9] Cornelis, C.—Jensen, R.—Hurtado, G.—Ślęzak, D.: Attribute Selection With Fuzzy Decision Reducts. Information Sciences, Vol. 180, 2010, pp. 209–224.

[10] Cover, T. M.—Thomas, J. A.: Elements of Information Theory. New York: Wiley, 1991.

[11] Dash, M.—Liu, H.: Consistency-Based Search in Feature Selection. Artificial Intelligence, Vol. 151, 2003, pp. 155–176.

[12] Duda, R. O.—Hart, P. E.—Stork, D. G.: Pattern Classification. 2nd edition. New York: John Wiley & Sons, 2001.

[13] Fleuret, F.: Fast Binary Feature Selection with Conditional Mutual Information. Journal of Machine Learning Research, Vol. 5, 2004, pp. 1531–1555.

[14] Forman, G.: An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Journal of Machine Learning Research, Vol. 3, 2003, pp. 1289–1305.

[15] Guyon, I.—Elisseeff, A.: An Introduction to Variable and Feature Selection. Journal of Machine Learning Research, Vol. 3, 2003, pp. 1157–1182.

[16] Hall, M. A.: Correlation-Based Feature Subset Selection for Machine Learning. Ph. D. Thesis. Department of Computer Science, University of Waikato, Hamilton, New Zealand 1999.

[17] Hilario, M.—Kalousis, A.: Approaches to Dimensionality Reduction in Proteomic Biomarker Studies. Briefings in Bioinformatics, Vol. 9, 2008, No. 2, pp. 102–118.

[18] Hu, Q.—Yu, D.—Liu, J.—Wu, C.: Neighborhood Rough Set Based Heterogeneous Feature Subset Selection. Information Sciences, Vol. 178, 2008, pp. 3577–3594.

[19] Hua, J.—Tembeb, W. D.—Doughertya, E. R.: Performance of Feature-Selection Methods in the Classification of High-Dimension Data. Pattern Recognition, Vol. 42, 2009, No 3, pp. 409–424.

[20] Huang, J.—Cai, Y.—Xu, X.: A Hybrid Genetic Algorithm for Feature Selection Wrapper Based on Mutual Information. Pattern Recognition Letters, Vol. 28, 2007, pp. 1825–1844.

[21] Huang, D.—Chow, T. W. S.: Effective Feature Selection Scheme Using Mutual Information. Neurocomputing, Vol. 63, 2005, pp. 325–343.

[22] Jain, A. K.—Duin, R. P. W.—Mao, J.: Statistical Pattern Recognition: A Review. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, 2000, No. 1, pp. 4–37.

[23] John, G. H.—Kohavi, R.—Pfleger, K.: Irrelevant Feature and the Subset Selection Problem. In: Proceedings of the 11th Int'l Conf. Machine Learning, San Francisco, CA: Morgan Kaufmann, 1994, pp. 121–129.

[24] Kira, K.—Rendell, L.: A Practical Approach to Feature Selection. In: Proceedings of the 9th Int'l Conf. Machine Learning, Morgan Kaufmann, 1992, pp. 249–256.

[25] Kwak, N.—Choi, C. H.: Input Feature Selection by Mutual Information Based on Parzen Window. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, 2002, No. 12, pp. 1667–1671.

[26] Levi, D.—Ullman, S.: Learning to Classify by Ongoing Feature Selection. Image and Vision Computing, DOI:10.1016/j.imavis.2008.10.010, 2010.

[27] Liang, J.—Yang, S.—Winstanley, A.: Invariant Optimal Feature Selection: A Distance Discriminant and Feature Ranking Based Solution. Pattern Recognition, Vol. 41, 2008, No. 5, pp. 1429–1439.

[28] Liu, H.—Sun, J.—Liu, L.—Zhang, H.: Feature Selection with Dynamic Mutual Information. Pattern Recognition, Vol. 42, 2009, No. 7, pp. 1330–1339.

[29] Liu, H.—Yu, L.: Toward Integrating Feature Selection Algorithms for Classification and Clustering. IEEE Transactions on Knowledge and Data Engineering, Vol. 17, 2005, No. 4, pp. 491–502.

[30] Neumann, J.—Schnorr, C.—Steidl, G.: Combined SVM-Based Feature Selection and Classification. Machine Learning, Vol. 61, 2005, pp. 129–150.

[31] Novovičová, J.—Somol, P.—Haindl, M.—Pudil, P.: Conditional Mutual Information Based Feature Selection for Classification Task. In: Proceedings of the 12$^{\text{th}}$ Iberoamericann Congress on Pattern Recognition, Valparaiso, Chile: Springer, 2007, pp. 417–426.

[32] Peng, H.—Long, F.—Ding, C.: Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, 2005, No. 8, pp. 1226–1238.

[33] Quinlan, R.: C4.5: Programs for Machine Learning. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

[34] Saeys, Y.—Inza, I.—Larrañaga, L.: A Review of Feature Selection Techniques in Bioinformatics. Bioinformatics, Vol. 23, 2007, No. 19, pp. 2507–2517.

[35] Schein, A. I.—Ungar, L. H.: Active Learning for Logistic Regression: An Evaluation. Machine Learning, Vol. 68, 2009, pp. 235–265.

[36] Somol, P.—Novovičová, J.—Pudil, P.: Notes on The Evolution of Feature Selection Methodology. Kybernetika, Vol. 43, 2007, No. 5, 2007, pp. 713–730.

[37] van der Maaten, L. J. P.—Postma, E. O.—van den Herik, H. J.: Dimensionality Reduction: A Comparative Review. Tilburg University, Technical Report, TiCC-TR 2009-005, 2009.

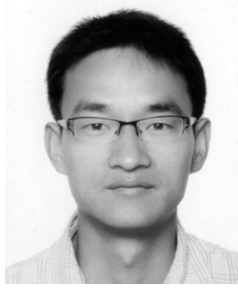
[38] Wang, G.—Lochovsky, F. H.—Yang, Q.: Feature Selection with Conditional Mutual Information MaxiMin in Text Categorization. In: Proceedings of the 13$^{\text{th}}$ ACM Int. Conf. Information and Knowledge Management, Washington D.C., USA: ACM Press, 2004, pp. 342–349.

[39] Witten, I. H.—Frank, E.: Data Mining-Pracitcal Machine Learning Tools and Techniques with JAVA Implementations. 2$^{\text{nd}}$ ed., Morgan Kaufmann Publishers 2005.

[40] Yu, L.—Liu, H.: Efficient Feature Selection via Analysis of Relevance and Redundancy. Journal of Machine Learning Research 5, 2004, pp. 1205–1224.

[41] Zhang, D.—Chen, S.—Zhou, Z.-H.: Constraint Score: A New Filter Method for Feature Selection With Pairwise Constraints. Pattern Recognition, Vol. 41, 2008, No. 5, pp. 1440–1451.

Huawen Liu works in Department of Computer Science at Zhejiang Normal University, P.R. China, as a Lecturer. He received his B. Sc. degree in computer science from Jiangxi Normal University in 1999, and M. Sc. and Ph. D. in computer science from Jilin University, P.R. China, in 2007 and 2009, respectively. His research interests include data mining, machine learning, pattern recognition and feature selection.



Yuchang Mo is Director of the Dependable Computing Lab and Associate Professor in the College of Mathematics, Physics and Information Engineering at the Zhejiang Normal University. He received his B. Sc., M. Sc. and Ph. D. degrees in computer science from Harbin Institute of Technology, Harbin, P.R. China. His research interests include dependable computing and networking, complex systems reliability engineering, fault-intrusion tolerant computing.

Jianmin Zhao joined Zhejiang Normal University in 1980 as a lecturer. Currently, he is a Professor and the Dean of College of Mathematics, Physics and Information Engineering at the Zhejiang Normal University, China. He has wide research interests, mainly pattern recognition, image processing, network engineering, agent, cloud computing, etc.