Pattern Recognition 46 (2013) 199–214

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Selecting feature subset for high dimensional data via the propositional FOIL rules

Guangtao Wang [a], Qinbao Song [a,*], Baowen Xu [b], Yuming Zhou [b]

[a] Department of Computer Science & Technology, Xi'an Jiaotong University, 710049, China
[b] Department of Computer Science & Technology, Nanjing University, 210093, China

ARTICLE INFO

Article history:
Received 28 February 2012
Received in revised form
9 May 2012
Accepted 31 July 2012
Available online 17 August 2012

Keywords:
Feature subset selection
Feature interaction
Propositional FOIL rule
Filter method

ABSTRACT

Feature interaction is an important issue in feature subset selection. However, most of the existing algorithms only focus on dealing with irrelevant and redundant features. In this paper, a propositional FOIL rule based algorithm FRFS, which not only retains relevant features and excludes irrelevant and redundant ones but also considers feature interaction, is proposed for selecting feature subset for high dimensional data. FRFS first merges the features appeared in the antecedents of all FOIL rules, achieving a candidate feature subset which excludes redundant features and reserves interactive ones. Then, it identifies and removes irrelevant features by evaluating features in the candidate feature subset with a new metric *CoverRatio*, and obtains the final feature subset. The efficiency and effectiveness of FRFS are extensively tested upon both synthetic and real world data sets, and it is compared with other six representative feature subset selection algorithms, including CFS, FCBF, Consistency, Relief-F, INTERACT, and the rule-based FSBAR, in terms of the number of selected features, runtime and the classification accuracies of the four well-known classifiers including Naive Bayes, C4.5, PART and IB1 before and after feature selection. The results on the five synthetic data sets show that FRFS can effectively identify irrelevant and redundant features while reserving interactive ones. The results on the 35 real world high dimensional data sets demonstrate that compared with other six feature selection algorithms, FRFS cannot only efficiently reduce the feature space, but also can significantly improve the performance of the four well-known classifiers.

1. Introduction

Feature subset selection, which is the technique of identifying the most salient features for learning, is an important research issue in the domains of machine learning and data mining. Its purpose is to focus a learning algorithm on those aspects of the data most useful for analysis and future prediction. It can reduce the dimensionality of the data and may improve a learner either in terms of learning performance, generalization capacity or model simplicity. It further helps understand and better interpret the results obtained by the learner, lessen its volume of storage, reduce the noise generated by irrelevant or redundant features and eliminate these useless features [1].

Feature subset selection achieves its intended purpose by identifying and removing as many irrelevant and redundant features as possible. As irrelevant features are useless or even negative to the predictive accuracy [2], and redundant features, even if being relevant to the learning target, since most of the information they carry is already present in other feature(s), they do not help for getting a better predictor. Therefore, many feature subset selection algorithms have been proposed to handle the irrelevant features [2] or/and redundant features [3]. Of these algorithms, some of them can effectively remove irrelevant features but can't identify redundant ones [4–8], some of them can remove the irrelevant features while taking into the redundant ones into account [9–14].

Apart from the identification of irrelevant and redundant features, an important but usually being neglected issue is feature interaction [15], which means although a single feature is irrelevant to target concept, when combined with other feature(s), it becomes very relevant. For example, suppose $F_1 \oplus F_2 = Y$, where F_1 and F_2 are two boolean variables, Y represents the target concept, and $\oplus$ represents the XOR operation. From the definition of the target concept Y we know that the discrimination ability of F_1/F_2 to Y is zero. Hence, F_1 and F_2 are irrelevant when we consider their discrimination abilities to Y separately. However, it is obvious that they become very relevant when we combine them together. This means removing interactive features will lead to poor predictive accuracy. Therefore, a feature subset selection algorithm should have the ability of eliminating the irrelevant features or/and redundant features while taking into consideration the feature interaction.

* Corresponding author. Tel.: +86 29 82668645; fax: +86 29 82668971.
E-mail addresses: gt.wang@stu.xjtu.edu.cn (G. Wang), qbsong@mail.xjtu.edu.cn (Q. Song).

0031-3203/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2012.07.028

Unfortunately, to the best of our knowledge, only a few algorithms can deal with this situation [15–17].

Let $\{f_1,f_2,\ldots,f_k\} \Rightarrow c$ be a rule. If we view the antecedent $f_1,f_2,\ldots,f_k$ as feature(s) and the consequent c as target concept, this rule can reveal the dependencies between feature(s) and target concept. Therefore, this kind of rules can be used for feature selection. Actually, Xie et al. [18] have employed association rules to select feature subset. However, their algorithm generates rules based on the Apriori algorithm [19], its time complexity is very high, thus is not applicable for high dimensional data. Moreover, it only focuses on relevant features, and does not consider redundant and interactive ones.

FOIL (First Order Inductive Learner) [20] is a rule-based learning algorithm, and the FOIL rules can be used as classification rules [21]. Compared with the Apriori algorithm, it is quite efficient especially on high dimensional data [22]. Moreover, the FOIL algorithm uses a special performance measure (*FoilGain*) that takes into account the different feature bindings. This means it is reasonable that using the FOIL rules to select features for high dimensional data while taking into consideration feature interaction.

Thus, in this paper, we propose a FOIL Rule based Feature subset Selection algorithm (FRFS).[1] FRFS firstly generates the FOIL classification rules using a modified propositional implementation of the FOIL algorithm. Then, it combines the features that appeared in the antecedents of all rules together, and achieves a candidate feature subset that excludes redundant features and reserves the interactive ones. Lastly, it measures the relevance of the features in the candidate feature subset by our proposed new metric *CoverRatio* (see Section 4.2.2 for details), and identifies and removes the irrelevant features. The experimental results on the 35 real world high dimensional data sets demonstrate that it cannot only efficiently reduce the feature space, but also can significantly improve the performance of the four well-known classifiers.

The rest of the paper is organized as follows. In Section 2, we describe the related work. In Section 3, we introduce the classification rule mining method based on restricted FOIL algorithm. In Section 4, we present the new feature subset selection algorithm FRFS. In Section 5, we report the experimental results. Finally, in Section 6, we summarize our work and draw some conclusions.

2. Related work

Feature subset selection has been an active research topic since 1970s, and a great deal of research work has been published.

Of the existing research work, most feature subset selection algorithms can effectively identify the irrelevant features based on different evaluation functions. But not all of them can eliminate the redundant features and take the feature interaction into consideration [23,24]. Thus, according to whether they can deal with irrelevant features, redundant features and the feature interaction, the existing feature subset selection algorithms can be generally grouped into three categories: (i) the algorithms that can only handle irrelevant features; (ii) the algorithms that can handle both irrelevant and redundant features; and (iii) the algorithms that can handle irrelevant and redundant features while considering feature interaction. Next, we give a brief review of the three categories respectively.

Traditionally, the research work on feature subset selection has focused on searching for relevant features. Feature weighting/ranking algorithms [25,26] weigh features individually and rank them based on their relevance to the target concept. The most prominent algorithm is Relief [5], which is proposed based on the assumption that a good feature should differentiate between instances with different target values and should have the same value for instances with the same target values, and weighs each feature based on a Euclidian distance measure. However, one of its major limitations is that it cannot detect the redundant features since two highly correlated features will be both highly weighted [27]. Relief-F [6] is an extended version of Relief, which enable the method to work for multiple classes and be robust on noisy and incomplete data sets, but still fails to handle redundant features. ElAlami [28] proposed to select feature subset from trained neural network using the Genetic Algorithm (GA). The GA is used to find the optimal relevant features, which maximize the output functions of trained neural network. Unfortunately, it is also incapable of removing redundant features.

However, redundant features should be eliminated as well since they also affect the speed and accuracy of classification [27,29]. For this purpose, many algorithms have been proposed. Among these algorithms, CFS [30] evaluates the feature subset rather than individual ones based on the assumption that a good feature subset is one that includes features highly correlated with the target concept, yet uncorrelated with each other. Consistency [11] searches for the minimal subset that separates different target concepts as consistently as the full set under certain strategy, which can effectively exclude the redundant features. FCBF [13,31] is a correlation-based method for efficient feature selection via relevance and redundancy analysis. The redundant peer between features is defined based on the symmetric uncertainty. MIFS [9] uses the mutual information as the evaluation function to identify the relevant features; and a weight coefficient is introduced in the calculation of mutual information to handle the redundant features. CMIM [32] iteratively picks features on the condition of the features already selected. It does not select a feature similar to already picked ones, even if it is individually powerful, as it does not carry additional information about the class to predict. Song et al. [14] propose a clustering-based feature selection algorithm which divides the features into several clusters, and selects the most representative feature from each cluster to form a subset of features. All these algorithms can effectively remove irrelevant and redundant features but do not take into consideration the feature interaction.

Feature interaction is drawing more attention in recent years. There can be two-way, three-way or complex multi-way interactions among features [33]. Jakulin and Bratko [15,16] use interaction gain as a heuristic to detect feature interaction. Their algorithms can detect two-way (one feature and the class) and three-way (two features and the class) interactions. Zhao and Liu [17] demonstrate that feature interactions can be implicitly handled by a carefully designed feature evaluation metric and a search strategy with a specially designed data structure. Chanda et al. [34] believe statistical interactions can capture the multivariate inter-dependencies among features, so they employ this interaction information to improve feature subset selection.

Recently, the association embodied in classification rules has been employed to select features. Xie et al. [18] proposed a rule-based feature subset selection algorithm FSBAR. This algorithm adopts the Apriori algorithm mining association rules and selects the feature subset by calculating the antecedent union of the corresponding association rules whose consequences are the same target concept, with predefined parameters being used to control the size of the selected feature subset. Unfortunately, it just detects relevant features and does not handle redundant and interactive features. Moreover, FSBAR is not suitable for high dimensional data due to its huge time consumption.

[1] The corresponding software can be obtained via qbsong@mail.xjtu.edu.cn.

In contrast, our algorithm employs the FOIL algorithm with a restriction to generate classification rules and is quite effective on high dimensional data. It eliminates irrelevant and redundant features while considering the multi-way feature interactions. Hence, it is quite different from these algorithms above.

3. Mining rules for feature selection with the restricted FOIL algorithm

3.1. Classification rule

Let $D=\{d_1,d_2,\ldots,d_n\}$ be a data set consisting of n instances, and $F=\{F_1,F_2,\ldots,F_m\}$ and Y be the feature space of D with m features and the domain of target concept (i.e., class label), respectively. Where F_i is the domain of the ith feature. The instance d_i can be denoted as a tuple (X_i,y_i), where X_i is an element of the set $F_1\times F_2\times\cdots\times F_m$, and $y_i\in Y$.

Suppose that the feature value itemset $\mathrm{FVIS}=\bigcup_{i=1}^{m}F_i$ contains all possible feature values, and the target value itemset $\mathrm{TVIS}=Y$, then the classification rule is defined as follows.

Definition 1 (*Classification Rule (CR)*). Let feature value set $\mathrm{FVSet}\subseteq\mathrm{FVIS}$ and target concept value $c\in\mathrm{TVIS}$, then the implication $\mathrm{FVSet}\Rightarrow\{c\}$ is defined as a classification rule.

From this definition we know that a propositional FOIL rule is a CR. In order to enhance the capabilities of the classification rules, and more conveniently use them to select features, for a classification rule r: $\mathrm{FVSet}\Rightarrow\{c\}$, two metrics *Support* and *Confidence* are defined as follows.

1. *Support*(r): The proportion of instances that contain both the feature value set FVSet and the target concept c of rule r. It reflects the frequency of rule r appearing in the data set.
2. *Confidence*(r): The proportion of the consequent c of rule r appearing in instances that contain the antecedent FVSet of rule r as well. It reflects the prediction power of rule r, i.e., the ability to interpret how well feature value set FVSet predicts the occurrence of target concept value c. The higher the confidence is, the more likely the target concept value c will be present with the feature value set FVSet.

3.2. Mining classification rules with restricted FOIL algorithm

3.2.1. Basic FOIL algorithm

FOIL is a first-order rule learner that generates rules to predict classification membership of instances. Rules are induced for each class (i.e. target concept value), one class at a time. For a given class, at first the instances are divided into two groups: those from the given class are regarded as positive instances, while those from all other classes are viewed as negative instances; then, FOIL tries to find a set of rules that covers[2] all positive instances but no negative instances. Finally, by merging the rules generated for each class, a set of rules of the given data set is obtained.

When learning an individual rule, FOIL first considers all possible rules consisting of a single test (i.e. feature value). It selects the best of these according to *FoilGain* [20], which favors a test that is true for many positive instances and few negative instances. Next, FOIL specializes the rule using the same search procedure as well as *FoilGain*, and the test with the maximum *FoilGain*, which would improve the rule by excluding many negative instances and few positive instances, is selected and added into the rule's antecedent. This specializing process continues until the rule covers no negative instances, resulting in a single rule whose antecedent is a conjunction of tests.

FoilGain is a measure used to evaluate the utility of adding a new test, and it is based on the numbers of positive and negative instances covered before and after adding the new test. Specifically, consider a given rule r, and a candidate test (feature value) v that might be added to the antecedent of r. Suppose r' is the rule created by adding test v to rule r. The value $\mathit{FoilGain}(v,r)$ by adding v to r is defined as

$$\mathit{FoilGain}(v,r)=P^*\cdot\left(\log\frac{P^*}{P^*+N^*}-\log\frac{P}{P+N}\right),$$

where P and N are the numbers of positive and negative instances covered by r, respectively. P^* and N^* are the numbers of the positive and negative instances covered by r', respectively.

We implement a propositional version of the FOIL algorithm, and use it to generate classification rules for feature selection. As introduced in Section 3.1, two metrics *Support* and *Confidence* are used to describe a rule. For the propositional FOIL rule r, its *Support*(r) is computed by $P/|D|$, and the *Confidence*(r) is calculated through $P/(P+N)$, where D represents the data set from which we generate the FOIL rules.

3.2.2. Restriction to FOIL algorithm

The rules generated by the FOIL algorithm can effectively predict classification membership of instances. However, these rules cannot be directly used for feature selection due to that redundant feature values may be added into rule's antecedent if the current best feature value (i.e. test) is identified only based on the *FoilGain*.

This can be illustrated by the following example. Suppose that there is a data set D (see Table 1 for details) with four boolean features F_1, F_2, F_3 and $F_{red}=F_1$, and the target concept $Y=F_1\oplus F_2\oplus F_3$. Since F_{red} is a copy of F_1, it is a redundant feature.

FOIL algorithm firstly learns a rule r for a specified target concept value (e.g. $Y=0$). Before the generation of the rule r, for every value f_v of feature F_i ($i\in\{1,2,3,red\}$), the number of positive instances with f_v is equal to that of negative ones, so any one is likely to be selected into r's antecedent. Suppose $F_1=0$ is the first one selected into r's antecedent, then the number of positive instances covered by r is 2 and the number of negative instances covered by r is 2 as well. This indicates only $F_1=0$ is not able to distinguish the different target concept values. Therefore, more feature values are needed to be selected into r's antecedent to further specialize the rule based on the *FoilGain*.

For the remaining three features F_{red}, F_2 and F_3, $\mathit{FoilGain}(F_{red}=0,r)=2\cdot(\log 2/(2+2)-\log 2/(2+2))=0$, and $\mathit{FoilGain}(F_2=0,r)=\mathit{FoilGain}(F_2=1,r)=\mathit{FoilGain}(F_3=0,r)=\mathit{FoilGain}(F_3=1,r)=1\cdot(\log 1/(1+1)-\log 2/(2+2))=0$. This means that it is impossible to distinguish the value of F_{red} from that of F_2 or F_3 by *FoilGain*. Therefore, the value of F_{red} may be identified as the current best value and selected into the rule's antecedent by mistake.

To overcome this shortcoming and ensure that the generated rules can be used for feature selection, we add a restriction to the basic FOIL algorithm. The detailed description of this restriction is given as follows.

Restriction. *The current best feature value should not only have the maximum FoilGain, but also be able to distinguish negative instances from positive ones as to the instances covered by the current rule, i.e.* $N^*<N$.

[2] A rule covers an instance if and only if the antecedent of the rule is a subset of the instance excluding the target concept.

Table 1
Example data set D.

F_1	F_2	F_3	F_{red}	Y
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	0
1	0	0	1	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

From the above example we know that the *FoilGain* of a relevant feature value is not always greater than that of a redundant feature value. Therefore, if the current best feature value is identified only based on the *FoilGain*, redundant feature values may be selected.

However, compared with relevant features, redundant features have no contribution to getting better prediction accuracy [2]. This reveals that the values of redundant features have no ability to further distinguish different target concept values. Therefore, after adding a redundant value to rule r's antecedent, the number of negative instances covered by r will remain unchanged. That is, $N^* = N$. On the contrary, the relevant feature values always contribute to distinguishing the different target concept values. And the addition of these relevant feature values will always lessen the number of negative instances covered by r. Therefore, restriction $N^* < N$ can be used to filter out redundant feature values when selecting the current best feature value.

Moreover, for the above given example, this restriction works well since the value of the redundant feature F_{red} will not be selected under the constraint of $N^* < N$.

4. Proposed feature subset selection algorithm

In this section, we first provide the definitions of relevant and redundant features as well as interactive feature based on classification rules. Then, we present the proposed feature subset selection algorithm.

4.1. Definitions of relevant, redundant and interactive features

As we know that there is an intrinsic relationship between a feature and its values, the properties of the feature can be studied through its values. Thus, for the sake of defining relevant, redundant and interactive features based on classification rules, we first provide the definitions of relevant, irrelevant and redundant feature values, and feature value interaction as follows.

Definition 2 (*Relevant Feature Value*). Let r: FVSet $\Rightarrow \{Y = y\}$ be a CR, $f \in$ FVSet be a specific feature value and FVSet$' =$ (FVSet$-\{f\}) \neq \phi$, and suppose $Conf(r)$ is the confidence of the classification rule r. Then f is a relevant feature value to the target concept Y if and only if

$$Conf(r) > Conf(FVSet' \Rightarrow \{y\}).$$

Otherwise, f is an *Irrelevant Feature Value*.

According to Definition 2, a feature value is relevant when (i) it appears in the antecedent of a CR, and (ii) the absence of this value has negative influence on the confidence of the rule. On the contrary, the feature values that never appear in the antecedent of any rule, or whose absence has no or even positive influence on confidence of the rule, are irrelevant feature values.

Definition 3 (*Redundant Feature Value*). Let FVSet be a non-empty feature value set, $f \in$ FVSet be a specific feature value, and FVSet$' =$ (FVSet$-\{f\}) \neq \phi$. Then f is a redundant feature value in FVSet if and only if

$$\forall y \in \text{TVIS}, \quad Conf(\text{FVSet} \Rightarrow \{y\}) = Conf(\text{FVSet}' \Rightarrow \{y\}).$$

According to Definition 3, a feature value in a given value set is redundant when its absence has no influence on the interpreting ability of the given value set to any target concept value. In other words, the information carried by this feature value is already present in other feature value(s). As a result, the redundant feature value does not contribute to getting better interpreting ability.

The confidence of a CR reflects the correlation between the feature value subset and the target concept value. The higher confidence indicates the stronger correlation. Therefore, the confidence can be used as a metric to evaluate the relevance of the target concept value with a feature value set. And based on the confidence, the feature value interaction is defined as follows.

Definition 4 (*k-th Feature Value Interaction*). Let FVSet $= \{f_1, f_2, \ldots, f_k\}$ be a feature value set with k values, $(A \subset \text{FVSet}) \neq \phi$, $B =$ FVSet$-A$, and $y \in Y$; and suppose r_F, r_A and r_B are the CRs of FVSet $\Rightarrow \{y\}$, $A \Rightarrow \{y\}$, and $B \Rightarrow \{y\}$, respectively. The feature values in FVSet are said to interact with each other if and only if

$$Conf(r_F) > Conf(r_A) \wedge Conf(r_F) > Conf(r_B).$$

According to Definition 4, the confidence of rule r_F is greater than those of rules r_A and r_B. This means that, comparing to either feature value set A or B, FVSet $= A \cup B$ has stronger interpreting ability to the target concept value. Moreover, the interpreting ability of A/B will become weaker with the absence of B/A. In this case, feature value sets A and B are said to interact with each other.

With the definitions of Relevant Feature Value, Irrelevant Feature Value, Redundant Feature Value and Feature Value Interaction above, we can define Relevant Feature, Redundant Feature, and Interactive Feature as follows.

Definition 5 (*Relevant Feature*). Let FVSet(F_i) be the value set of the feature F_i. Feature F_i is relevant to the target concept Y if and only if

$$\exists f_{ij} \in \text{FVSet}(F_i), \quad \{f_{ij} | f_{ij} \text{ is a Relevant Feature Value}\} \neq \phi.$$

Otherwise, F_i is an *Irrelevant Feature*, where f_{ij} denotes the jth value of FVSet(F_i).

According to Definition 5, a feature is relevant to the target concept as long as one of its values is relevant; and a feature is irrelevant when all of its values are irrelevant. This is consistent with but much stricter than the classical definition of relevant feature [2]. It not only meets the inequality in the classical definition, but also requires that at least one of its feature values helps to improve the interpreting ability of a rule, i.e. the confidence of the rule.

Definition 6 (*Redundant Feature*). Let FVSet(F_i) be the value set of the feature F_i. Feature F_i is redundant if and only if

> All values in FVSet(F_i) are Redundant Feature Values, or at least one is Redundant Feature Value and the rest are Irrelevant Feature Values.

According to Definition 6, a feature being redundant is due to the fact that either each value of this feature is redundant or some values of this feature are redundant while others are irrelevant.

Definitions 3 and 2 reveal that these redundant and irrelevant feature values hold unnecessary information to interpret the target concept. Moreover, the values of the redundant feature under the classical definition [3] have the same characteristics (i.e. carry unnecessary information). This indicates that the new definition of redundant feature agrees well with the classical one.

Definition 7 (*Interactive Feature*). Let $FSet = \{F_1, F_2, \ldots, F_k\}$ be a feature subset with k features, and *VASet* be its value-assignment sets. Features F_1, F_2, ..., F_k are said to interact with each other if and only if

$$\exists\ fvset \in VAset,\ \{fvset \mid fvset \text{ is a feature value set with } k\text{-th Feature Value Interaction}\} \neq \phi.$$

It is noticed that the intrinsic characteristic of feature interaction is its irreducibility, i.e., the absence of any feature will cause the interactive feature subset to lose its interpreting ability to the target concept. And the interpreting ability of the feature subset can be studied by its value-assignment. According to Definition 4 of feature value interaction, for a feature value set (i.e. a value-assignment) with value interaction, the interpreting ability of this value set to the target concept will be lost with the absence of any of its elements. This characteristic of the value interaction is consistent with that of the feature interaction. Therefore, Definition 7 that defines interactive feature based on feature value interaction is reasonable.

4.2. Feature subset selection algorithm

In this section, we present our proposed feature subset selection algorithm, FRFS, which not only retains relevant features and excludes irrelevant and redundant features, but also takes feature interaction into consideration.

The algorithm consists of two connected steps of the (i) *redundant feature exclusion and interactive feature reservation* and the (ii) *irrelevant feature identification*.

In the first step, classification rule set (CRSet) is mined from a given data set by the restricted-FOIL algorithm. Then, a candidate feature subset is achieved by joining the features whose values appeared in the antecedents of the classification rules in CRSet. Although the candidate subset excludes redundant features and reserves interactive features, there might be some irrelevant features introduced at the same time. Section 4.2.1 contains the detailed explanation.

The second step is used to identify and remove irrelevant features. For this purpose, a new rule-support-based metric, which is referred to as feature *CoverRatio* (see Section 4.2.2 for details), is defined to measure the relevance of a feature to the target concept. And for a specific feature, the smaller the *CoverRatio* is, the more likely it is an irrelevant feature. Thus, in this step, we firstly compute the *CoverRatio* for each feature in the candidate feature subset obtained in Step 1, then, the features whose *CoverRatio* is smaller than a predefined threshold are considered as irrelevant features and moved away from the candidate subset. Section 4.2.2 provides the details.

In the following sections, we first explain why these two steps can effectively address corresponding issues in feature selection, then present the pseudo-code of our proposed algorithm and analyze its time complexity.

4.2.1. Redundant feature exclusion and interactive feature reservation

In order to guarantee the first step of our proposed algorithm is able to exclude redundant features and reserve interactive ones, we first give the following Lemma 1.

Lemma 1. *Let RuleSet be a set of classification rules, if it satisfies the following two conditions*:

Condition 1. *Given* $(r: A \Rightarrow \{y\}) \in RuleSet : \forall\ (A' \subset A) \neq \phi$ *and* $\forall \hat{A} \supset A$, $Conf(A' \Rightarrow \{y\}) < Conf(r) \wedge Conf(\hat{A} \Rightarrow \{y\}) \leq Conf(r)$;

Condition 2. *All instances of a given data set are covered by RuleSet, and each instance is only covered by one rule*;

then the feature subset FSet, which is achieved by joining the features whose values appeared in the antecedents of rules in RuleSet, not only includes all relevant features, but also excludes redundant features and reserves interactive features.

Proof. The proof of Lemma 1 consists of the following two parts.

(1) FSet includes all relevant features.

According to Condition 1, for a given rule $r \in$ RuleSet, the absence of any feature value of r's antecedent will result in the attenuation of its confidence. Therefore, based on Definition 2, all the values of r's antecedent are necessary, and they are all relevant feature values to the target concept.

Meanwhile, Condition 2 requests the rules in RuleSet must cover the whole data set, i.e. the given data set can be represented by these rules. This states that RuleSet contains all the representative rules. Thus, all relevant feature values are included in the antecedents of these rules.

Finally, based on Definition 5, the feature with a relevant value is a relevant feature. Since the antecedents of the rules in RuleSet include all relevant values, then the feature subset FSet, which is obtained by mapping the values appeared in these rules to corresponding features, includes all relevant features.

(2) FSet excludes redundant features and reserves interactive features.

Definition 3 of redundant feature value shows that the elimination of the redundant feature value from the rule's antecedent does not change its confidence. However, for any rule $r \in$ RuleSet, according to Condition 1, the absence of any feature value of r's antecedent will change r's confidence. This means there is no redundant feature value in r's antecedent. Moreover, Definition 6 reveals that if none value of a feature is redundant, then the feature is not redundant as well. Therefore, feature set FSet excludes redundant features.

For a rule $r \in$ RuleSet, Condition 1 reveals that the elimination of any non-empty subset of r's antecedent will lessen r's confidence. This is equivalent to Definition 4 of feature value interaction. Meanwhile, Condition 2 guarantees RuleSet contains all the representative rules. Thus, all the feature value interactions are reserved by the rules in RuleSet. From Definition 7 of interactive feature we know that feature set FSet reserves all interactive features. □

From Lemma 1 we know that in order to exclude redundant features and reserve interactive ones, the rules in RuleSet have to meet both *Condition* 1 and *Condition* 2. Recall that in our proposed algorithm FRFS, the classification rule set (CRSet) is generated by the restricted FOIL algorithm. Since FOIL is originally proposed to generate rules covering the whole data set, *Condition* 2 is naturally satisfied. Then a question is raised: whether these rules in CRSet also meet *Condition* 1 or not? We give the analysis as follows.

In the restricted FOIL, each rule $r: A \Rightarrow \{y\}$ is generated by repeatedly adding these feature values into its antecedent until no negative instances are covered by this rule. That is, all the instances containing r's antecedent A are positive (i.e. with target concept value y). Moreover, from the definition of rule-confidence

we know that the confidence of rule r is always 1, which is the maximum value of the confidence of a rule. Therefore, rule r generated by the restricted FOIL has the following FOIL *rule property*:

$$(\forall A' \subset A,\ Conf\ (A' \Rightarrow \{y\}) \le Conf\ (r)) \wedge (\forall \hat{A} \supset A,$$
$$Conf\ (\hat{A} \Rightarrow \{y\}) \le Conf\ (r)).$$

However, it is noticed that this *rule property* is slightly different from *Condition* 1 of Lemma 1. The difference is that there possibly exists a feature value set $(A' \subset A) \neq \phi$ with $Conf\ (A' \Rightarrow \{y\}) = Conf\ (r)$. This means rules generated by the restricted FOIL meet a condition that is weaker than *Condition* 1. As a result, the feature subset selected directly based on these rules in CRSet will be a superset of that achieved based on the rules in Lemma 1. Thus, some uninteresting features (such as irrelevant or redundant features) can be introduced.

According to the analysis above, these uninteresting features are derived from the feature values in $(A-A')$, where $(A' \subset A) \neq \phi$ and $Conf\ (A' \Rightarrow \{y\}) = Conf\ (r)$. In this case, since the absence of any feature value in $(A-A')$ has no influence on the rule-confidence, according to the Definitions 2 and 3 of relevant and redundant feature values, the feature values in $(A-A')$ are either irrelevant or redundant. However, a redundant feature value provides the information already presents in other feature value(s), and does not contribute to distinguishing the negative instances from positive ones. Therefore, according to the restriction of FOIL introduced in Section 3.2.2, the redundant feature values will not be selected, and never appear in the antecedent of any rule in CRSet. So the feature values in $(A-A')$ can only be irrelevant feature values; and the uninteresting features derived from these feature values are irrelevant features.

Therefore, by collecting the features whose values appeared in the antecedents of the rules generated by the restricted FOIL, we can obtain a feature subset that excludes redundant features and reserves interactive ones. Yet, unfortunately, irrelevant features may be introduced as well.

4.2.2. Irrelevant feature identification

Since the irrelevant features usually show weak correlation to the target concept, their values have no or quite weak ability to distinguish different target concept values. This distinguishing ability is measured by the *FoilGain* in FOIL. As long as the size (i.e. the number of instances) of a data set is large enough to guarantee the distribution of feature values approximating the true distribution, the *FoilGain* of a relevant feature value is usually greater than that of an irrelevant feature value.

Unfortunately, in the FOIL algorithm, the size of the available data set (i.e. the number of instances covered the current rule) constantly decreases with the rule generation process. With the decrease of the size of the available data set, it will be more difficult to guarantee the distribution of feature values approximating the true distribution, and become more confusing to distinguish irrelevant features from relevant ones. In this case, the values of irrelevant feature might show "ability" to distinguish different target concept values as well. And their *FoilGain* could be equal to or even greater than that of a relevant feature value. As a result, these irrelevant feature values would be selected into the rules' antecedents by mistake. Moreover, all of these rules with irrelevant values have relatively small *Support* since these rules cover only a small amount of instances.

Once the values of irrelevant features are selected into the rule's antecedent, the feature subset, which is achieved by mapping the feature values appeared in the rules' antecedents to the corresponding features, will include the irrelevant features.

In order to address this problem, a rule-support-based new metric, which is referred to as *CoverRatio*, is proposed to evaluate the relevance of a feature to the target concept and used to identify irrelevant features.

Since the property of a feature can be measured through the properties of its values, we first define a metric *vCoverRatio* to measure the relevance of a feature value $f_{i,j}$ to a target concept, which is the consequent of the rule r, as

$$vCoverRatio(f_{i,j}) = \frac{\sum_{r \in R_{f_{i,j}}} Supp(r)}{freq(\{f_{i,j}\})}, \quad (1)$$

where $Supp(r)$ denotes the support of rule r, $f_{i,j}$ is the j-th value of feature F_i and a member of rule r's antecedent as well, $R_{f_{i,j}}$ is the set of rules whose antecedents contain $f_{i,j}$, and $freq(\mathrm{VSet})$ is the frequency of a value set VSet over the whole data set.

In Eq. (1), the numerator, which is the summation of the *Support* of all rules whose antecedents include the feature value $f_{i,j}$, represents the total frequency of $f_{i,j}$ selected into the rules. The denominator is the frequency of $f_{i,j}$ over the whole data set. Therefore, the $vCoverRatio(f_{i,j})$ represents the proportion of $f_{i,j}$ selected into the rules, and its maximum value is 1.

The *vCoverRatio* can be used to measure the relevance of a feature value to a target concept and identify the irrelevant values, since that (i) the *vCoverRatio* of an irrelevant value is usually relatively small, and (ii) the *vCoverRatio* of a relevant value is usually greater than that of an irrelevant one. This is illustrated as follows:

(1) *For an irrelevant feature value $f_{i,j}$, its vCoverRatio is relatively small.*

   (a) *The number of rules in $R_{f_{i,j}}$ is relatively small.*
   We know that the irrelevant feature values are useless for predicting classification membership while FOIL rules can be used to effectively explore this membership. This means only a very small amount of rules include irrelevant feature values.
   Moreover, in FOIL, the irrelevant feature values might be selected into the rule's antecedent if and only if the size of the current available data set is small. When the size of available data set is small, the distribution of feature values would deviate from the true distribution. As a result, it becomes much more difficult to distinguish irrelevant feature values from relevant ones. In this case, all irrelevant values have the same possibility of being selected into a rule's antecedent. And the selection of a specific irrelevant value into the rule's antecedent will be of great randomness. That is, for a specific irrelevant value, it is much less likely to simultaneously appear in multiple rules.
   Therefore, for the given irrelevant feature value $f_{i,j}$, the number of rules in $R_{f_{i,j}}$ is relatively small.
   (b) *The FOIL rule $r: A \Rightarrow \{y\}$ including $f_{i,j}$ always has relatively small $Supp(r)/freq(\{f_{i,j}\})$.*

   - Let $\alpha = Supp(r)/freq(\{f_{i,j}\})$. According to the definition of rule's support, we have $Supp(r) = freq(A \cup \{y\})$. From the definitions of α and $Supp(r)$ we obtain $freq(A \cup \{y\}) = \alpha \cdot freq(\{f_{i,j}\})$. Since $f_{i,j}$ is included in r (i.e., $f_{i,j} \in A$), $\{f_{i,j}, y\} \subseteq (A \cup \{y\})$, and the frequency of a value set is usually greater than/equal to that of its superset, Therefore, $freq(\{f_{i,j}, y\}) \ge freq(A \cup \{y\}) = \alpha \cdot freq(\{f_{i,j}\})$, and $freq(\{f_{i,j}, y\})/freq(\{f_{i,j}\}) \ge \alpha$. From the definition of rule's confidence, we know that

$freq(\{f_{i,j},y\})/freq(\{f_{i,j}\})$ just is the confidence of rule $r': \{f_{i,j}\} \Rightarrow \{y\}$.

- Assume α is a large value, then the confidence of rule r' will be large as well since $freq(\{f_{i,j},y\})/freq(\{f_{i,j}\}) \geq \alpha$. This means that the feature value $f_{i,j}$ itself can be used to effectively describe the target concept value y, and $f_{i,j}$ is very relevant to y. However, this is contradictory with that $f_{i,j}$ is an irrelevant value. Therefore, this states that $\alpha = Supp(r)/freq(\{f_{i,j}\})$ will only be a relatively small value.

(c) Since $vCoverRatio(f_{i,j}) = \sum_{r \in R_{f_{i,j}}} Supp(r)/freq(\{f_{i,j}\}) = \sum_{r \in R_{f_{i,j}}} Supp(r)/freq(\{f_{i,j}\})$, according to the analyses above, for an irrelevant feature value $f_{i,j}$, the number of rules in $R_{f_{i,j}}$ and $Supp(r)/freq(\{f_{i,j}\})$ for each rule $r \in R_{f_{i,j}}$ are both relatively small. Therefore, the *vCoverRatio* of $f_{i,j}$ is usually relatively small as well.

(2) *The vCoverRatio of a relevant value is greater than that of an irrelevant one.*

For a relevant feature value, the relevance to the target concept implies that it has the ability to distinguish different target concept values. The FOIL algorithm utilizes this distinguishing ability to generated rules for a given data set. Thus, the relevant feature value will be more likely to be selected into the rules' antecedents. That is, the possibility of a relevant feature value being selected into the rules' antecedents is relatively larger than that of an irrelevant feature value. This possibility can be measured by *vCoverRatio* since it denotes the proportion of a feature value being selected as a member of the antecedent of a FOIL rule. Therefore, compared with irrelevant feature value, the *vCoverRatio* of relevant feature value is usually relatively larger.

Given *vCoverRatio* of a feature value, we can identify whether the corresponding feature is irrelevant or not with a new metric *CoverRatio*. It is defined as

$$CoverRatio(F_i) = \frac{1}{K} \sum_{f \in \mathrm{VSet}(F_i)} vCoverRatio(f), \quad (2)$$

where $\mathrm{VSet}(F_i)$ is a feature value set consisting of the values from F_i and being selected into rules' antecedents at the same time, and K denotes the number of feature values in $\mathrm{VSet}(F_i)$.

According to Definition 5, a feature is irrelevant if and only if all of its values are irrelevant. Moreover, the above analysis about *vCoverRatio* indicates that the *vCoverRatio* of an irrelevant feature value is usually relatively smaller than that of a relevant feature value. Thus, the *CoverRatio* of a irrelevant feature, which is the arithmetic mean of some smaller *vCoverRatio*, is still relatively smaller than that of a relevant feature.

Therefore, *CoverRatio* is able to detect irrelevant features.

4.2.3. FRFS algorithm

Given a data set D and a predefined *CoverRatio* threshold δ, our proposed FRFS algorithm selects a feature subset S from D. The corresponding descriptive pseudo-code is shown in Algorithm 1.

Algorithm 1. FRFS

Inputs:
D: the given data set;
δ: a predefined threshold
Output:
S: the selected feature subset

```
1   S ← φ, FSet ← φ;
2   CRSet = restricted_FOIL (D);
3   for each r ∈ CRSet do
4   |  FSet = FSet ∪ {features whose values appeared
    |  in r's antecedent};
5   end
6   for each F ∈ FSet do
7   |  if FSet ≠ φ then
8   |  |  ratio = ComputeCoverRatio (F,CRSet);
9   |  |  if ratio > δ then
10  |  |  |  S = S ∪ {F};
11  |  |  end
12  |  |  FSet = FSet−{F};
13  |  end
14  end
15  return S;
```

The pseudo-code consists of two parts. In the first part (lines 1–5), classification rule set CRSet is generated from the data set D by function `restricted_FOIL()`. The function is implemented based on the basic FOIL algorithm with the restriction introduced in Section 3.2.2. That is, when choosing the best feature value being added into the antecedent of the current rule, the one picked up should not only has the maximum *FoilGain*, but also be helpful in distinguishing the negative instances from the positive ones. Afterward, the candidate feature subset FSet is obtained by combining the features whose values appeared in the antecedents of the rules in CRSet.

In the second part (lines 6–14), for each feature F in candidate feature subset FSet, the *CoverRatio* of F is computed by function `ComputeCoverRatio()` according to Eq. (2) based on F and CRSet. If the *CoverRatio* of F is greater than the predefined threshold δ, F is added into the optimal feature subset S, and removed from FSet at the same time. Otherwise, F is simply removed from FSet. Repeat this process until FSet is empty. Then, S is returned as the optimal feature subset.

In this algorithm, a larger value of a feature's *CoverRatio* indicates that the feature is more relevant to the target concept. Thus, a large δ is associated with a high probability of removing relevant features. The threshold δ is heuristically set to be $0.1 \times CoverRatio_{max}$ if not otherwise mentioned, where $CoverRatio_{max}$ is the maximum *CoverRatio* of the feature of candidate feature subset FSet. Of course this threshold also can be adjusted via the standard cross-validation.

Time complexity analysis. Suppose that N is the number of instances and K is the number of features in the given data set D. In the first part, CRSet is generated based on function `restricted_FOIL()`, and the time complexity is $O(M \cdot K \cdot N \cdot |R|)$ [22], where M is the average number of values of each feature and $|R|$ is the number of rules in CRSet. The time complexity of obtaining the candidate feature set FSet is $O(|R|)$. In the second part, all the features in FSet are evaluated by feature's *CoverRatio*, which is computed by scanning the rules in CRSet only one pass. The time complexity is $O(|\mathrm{FSet}| \cdot |R|)$, where $|\mathrm{FSet}|$ denotes the size of FSet, which is equal to K in the worst case. Consequently, the time complexity of FRFS is $O(M \cdot K \cdot N \cdot |R|)$. Usually $M \ll N \wedge |R| \leq N$, and in the worst case $|R| = N$. Therefore, there is a linear relationship between the time complexity of FRFS and the number of features K. This indicates that the new algorithm FRFS is efficient on high dimensional data sets, especially when $K \gg N$.

Table 2
Relevant, irrelevant, redundant, and interactive features of the five synthetic data sets.

Data name	Relevant features	Irrelevant features	Redundant features	Interactive features
synData1	a_0,a_1,a_5,a_6,a_8	a_2,a_3,a_4,a_7,a_9	–	(a_0,a_1,a_5), (a_0,a_1,a_6,a_8), (a_0,a_1,a_5,a_8), (a_5,a_6,a_8)
synData2	a_1,a_5,a_6,a_8	a_0,a_2,a_3,a_4,a_7,a_9	r	(a_1,a_6,a_8)
MONK1	a_1,a_2,a_5	a_3,a_4,a_6	–	(a_1,a_2)
MONK2	$a_1,a_2,\ldots,a_6$	–	–	All (a_i,a_j) pairs $(1\le i<j\le 6)$
MONK3	a_2,a_4,a_5	a_1,a_3,a_6	–	(a_4,a_5), (a_2,a_5)

5. Experimental results and analysis

In this section, we empirically evaluate the performance of FRFS, and present the experimental results compared with the other six different types of feature subset selection algorithms upon both synthetic and real world data sets.

5.1. Data sets

5.1.1. Synthetic data sets

In order to directly evaluate how well FRFS deals with irrelevant, redundant and interactive features, five synthetic data sets with all the irrelevant, redundant and interactive features being known are employed.

The first two data sets synData1 and synData2 are generated by the data generation tool RDG1 of the data mining toolkit WEKA.[3] The other three data sets about MONK's problems are available from UCI Machine Learning Repository [35].

The five data sets are described as follows:

1. synData1
   There are 100 instances and ten boolean features $a_0,a_1,\ldots,a_9$. Of the 10 features, five are irrelevant. The target concept c is defined by $c=(a_0\wedge a_1\wedge\overline{a_5})\vee(a_0\wedge\overline{a_1}\wedge a_6\wedge a_8)\vee(a_0\wedge a_1\wedge a_5\wedge a_8)\vee(\overline{a_0}\wedge a_1\wedge a_5\wedge\overline{a_8})\vee(a_5\wedge a_6\wedge a_8)\vee(a_0\wedge\overline{a_1})$.
2. synData2
   There are 100 instances, eleven boolean features denoted as $a_0,a_1,\ldots,a_9$ and a redundant feature r that is a copy of a_5. Of the ten non-redundant features, six are irrelevant. The target concept c is defined by $c=\overline{a_5}\vee(\overline{a_1}\wedge\overline{a_6}\wedge\overline{a_8})$.
3. MONK1
   There are 432 instances and six features $a_1,a_2,\ldots,a_6$. The target concept c is defined by $c=(a_1=a_2)\vee(a_5=1)$.
4. MONK2
   There are 432 instances and six features $a_1,a_2,\ldots,a_6$. The target concept c is defined by exactly two of $\{a_1=1,a_2=1,\ldots,a_6=1\}$.
5. MONK3
   There are 432 instances and six features $a_1,a_2,\ldots,a_6$. The target concept c is defined by $c=(a_5=3\wedge a_4=1)\vee(a_5\neq 4\wedge a_2\neq 3)$. 5% class noise was added to the training set.

For each data set, the features appearing in the definition of the target concept are all relevant, while the absent features are either redundant or irrelevant. The conjunctive terms in the definition of the target concept imply interactive features. Table 2 shows the relevant, irrelevant, redundant and interactive features of each synthetic data set.

5.1.2. Real world data sets

35 extensively used real world data sets, which come from different areas such as Computer, Image, Life, Microarray, Physical and Text, are employed. The sizes of these data sets vary from 34 to 4601 instances, and the numbers of the features are between 57 and 19 993. There are 22 (62.9%) data sets whose numbers of features are greater than 2000, 15 (42.9%) data sets whose numbers of features are greater than 5000, and 6 (17.1%) data sets whose numbers of features are greater than 10 000.

For data sets containing features with continuous values, the well-known MDL discretization method [36] is conducted to discrete these features.

Table 3 summarizes the 35 data sets in terms of number of features (denoted as F), the number of instances (denoted as I), the number of target concept values (denoted as T), area of the data set (denoted as Area), and the source of the data set (denoted as Source).

5.2. Experiment setup

1. Six representative feature selection algorithms were selected to be compared with FRFS.
   The algorithms include four well-known and frequently used CFS [30], Consistency [11], FCBF [13] and Relief-F [6]. These algorithms can effectively identify irrelevant features, and some of them (such as CFS, Consistency and FCBF) can detect redundant features.
   To further evaluate the performance of FRFS in terms of handling feature interaction, an algorithm INTERACT [17], which is specifically proposed to address the feature interaction, is selected as one benchmark algorithm. Its source codes are available online.[4]
   Moreover, since our proposed FRFS is a rule-based feature selection algorithm, a latest rule-based feature selection algorithm FSBAR [18] is selected as well.
   Of these algorithms, both CFS and Consistency exploit best-first strategy to search the space of feature subsets.
   For the rest algorithms, there are parameters to be preassigned by user. Such as, for FCBF, a relevance threshold needs to be set to identify all predominant features to the target concept and remove the rest; for Relief-F, both the number of nearest neighbors and the size of the sample affect the weights of the features; for INTERACT, there is a parameter, *c-contribution* threshold, used to identify the irrelevant features; and for FSBAR, a parameter, cycle number, is used to achieve considerable results of feature selection. The performance of these algorithm is closely related to the setting of these parameters. To make a fair comparison to the new proposed algorithm FRFS, the parameters of these algorithms were tuned via the standard cross-validation strategy.

[3] http://www.cs.waikato.ac.nz/ml/weka/.

[4] http://www.public.asu.edu/huanliu/INTERACT/INTERACTsoftware.html.

Table 3
Summary of the 35 real world data sets.

Data ID	Data name	I	F	T	Area	Source
1	Internet advertisements	3279	1558	2	Computer	http://archive.ics.uci.edu/ml/datasets.html
2	isol5	1559	617	26	Computer	http://archive.ics.uci.edu/ml/datasets.html
3	mfeat-fac	2000	216	10	Computer	http://archive.ics.uci.edu/ml/datasets.html
4	mfeat-fou	2000	76	10	Computer	http://archive.ics.uci.edu/ml/datasets.html
5	multiple-features	2000	649	10	Computer	http://archive.ics.uci.edu/ml/datasets.html
6	spambase	4601	57	2	Computer	http://archive.ics.uci.edu/ml/datasets.html
7	movement_libras	300	90	15	Image, Face	http://archive.ics.uci.edu/ml/datasets.html
8	ORL10P	100	10 304	10	Image, Face	http://featureselection.asu.edu/datasets.php
9	AR10P	130	2400	10	Image, Face	http://featureselection.asu.edu/datasets.php
10	PIE10P	210	2420	10	Image, Face	http://featureselection.asu.edu/datasets.php
11	Arrhythmia	452	279	16	Life	http://archive.ics.uci.edu/ml/datasets.html
12	Audiology (standardized)	226	69	24	Life	http://archive.ics.uci.edu/ml/datasets.html
13	splice	3190	60	3	Life	http://archive.ics.uci.edu/ml/datasets.html
14	tiger	1220	231	2	Life	http://sci2s.ugr.es/keel/datasets.php
15	CLL-SUB-111	111	11 340	3	Microarray, Bio	http://featureselection.asu.edu/datasets.php
16	TOX-171	171	5748	4	Microarray, Bio	http://featureselection.asu.edu/datasets.php
17	SMK-CAN-187	187	19 993	2	Microarray, Bio	http://featureselection.asu.edu/datasets.php
18	Gloabel cancer map	144	16 063	14	Microarray, Bio	http://www.upo.es/eps/bigs/datasets.html
19	leukemia_test	34	7129	2	Microarray, Bio	http://www.upo.es/eps/bigs/datasets.html
20	leukemia_train	38	7129	2	Microarray, Bio	http://www.upo.es/eps/bigs/datasets.html
21	Musk(Version1)	476	167	2	Physical	http://archive.ics.uci.edu/ml/datasets.html
22	spectrometer	531	102	48	Physical	http://archive.ics.uci.edu/ml/datasets.html
23	BASEHOCK	1993	4862	2	Text	http://featureselection.asu.edu/datasets.php
24	fibs.wc	2463	2000	17	Text	http://tunedit.org/repo/Data/Text-wc
25	la2s.wc	3075	12 433	6	Text	http://tunedit.org/repo/Data/Text-wc
26	oh0.wc	1003	3182	10	Text	http://tunedit.org/repo/Data/Text-wc
27	oh5.wc	918	3012	10	Text	http://tunedit.org/repo/Data/Text-wc
28	re0.wc	1504	2886	13	Text	http://tunedit.org/repo/Data/Text-wc
29	tr12.wc	313	5804	8	Text	http://tunedit.org/repo/Data/Text-wc
30	tr21.wc	336	7903	6	Text	http://tunedit.org/repo/Data/Text-wc
31	tr23.wc	204	5832	6	Text	http://tunedit.org/repo/Data/Text-wc
32	tr31.wc	927	10 128	7	Text	http://tunedit.org/repo/Data/Text-wc
33	tr41.wc	878	7454	10	Text	http://tunedit.org/repo/Data/Text-wc
34	tr45.wc	690	8261	10	Text	http://tunedit.org/repo/Data/Text-wc
35	wap.wc	1560	8461	20	Text	http://tunedit.org/repo/Data/Text-wc

2. Classification accuracy over selected feature subset is extensively used as a measure to evaluate the performance of the feature selection algorithm in feature selection literature. This is due to the fact that the relevant features of real world data sets are usually not known in advance, and we cannot directly evaluate how good a feature selection algorithm is by the features selected. However, different classification algorithms have different biases, and a feature subset selection algorithm may be more suitable for some classification algorithms than others. With this in mind, four different types of well-known classification algorithms including probability-based Naive Bayes [37], decision tree-based C4.5 [38], rule-based PART [39] and instance-based IB1 [40] were selected. Note that the classifier Naive Bayes assumes conditional independence between features [37]. Thus, picking up the interactive features or not should have limited influence on it. We employ this classifier in our experiment to confirm this phenomenon. Moreover, since the new proposed algorithm FRFS is rule-based, in order to explore whether FRFS is superior to the other algorithms for rule-based classifier, the classifier PART is employed.
In order to achieve a relatively accurate and stable estimation of the classification accuracy, *stratified* 5×10-*fold cross-validation* [41] procedure is performed. That is, for a given data set, each feature selection algorithm and each classifier are repeatedly performed on the data set with 10-fold cross-validation by five times, with each time the instances of the data set are randomly reordered. Afterward, for each classifier, we obtain $5\times 10=50$ classification accuracies for each feature selection algorithm over each data set. By averaging these accuracies, we get the estimation of the classification accuracy of each classifier under each feature selection algorithm for each data set.

5.3. Results on the synthetic data sets

Table 4 shows the feature subsets selected by the seven feature subset selection algorithms on the five synthetic data sets. In this table, '_' indicates a missing relevant feature, and the boldface letter indicates an irrelevant or a redundant feature selected by mistake. The last row "Relevant features" reports the actual relevant features of each data set.

From Tables 2 and 4, we observe that:

1. Only algorithm FRFS removes all irrelevant features while reserving all relevant features for all the five data sets. Algorithms Consistency and Relief-F can remove all irrelevant features as well, but they remove some relevant features on these data sets by mistake at the same time. Algorithms CFS, FCBF, INTERACT and FSBAR identify the irrelevant features for some but not all data sets.
2. Except algorithms CFS and Relief-F, all other five algorithms can identify and remove the redundant feature r in the data set "synData2".
3. Only algorithm FRFS reserves all the interactive features on all the five data sets. INTERACT works well on all the data sets except for "synData2". The other algorithms identify all the interactive features on some but not all the data sets.

5.4. Results on the real world data sets

In this section, we present and discuss the comparison results between FRFS and each of the other six feature subset selection

Table 4
Features selected by the seven algorithms on the five synthetic data sets.

FSS algorithm	synData1	synData2	MONK1	MONK2	MONK3
CFS	$a_0,\_,a_5,a_6,a_8$	$\boldsymbol{a_0},a_1,a_5,\_,\boldsymbol{a_7},\_,\boldsymbol{r}$	$\_,\_,a_5$	$\_,\_,\_,\_,a_5,\_$	$a_2,\_,\_$
FCBF	$a_0,\_,a_5,a_6,a_8$	$\boldsymbol{a_0},a_1,a_5,\_,\boldsymbol{a_7},\_$	$\_,\_,a_5$	a_1,a_2,a_3,a_4,a_5,a_6	a_2,a_4,a_5
Consistency	a_0,a_1,a_5,a_6,a_8	a_1,a_5,a_6,a_8	a_1,a_2,a_5	$\_,\_,\_,\_,\_,\_$	a_2,a_4,a_5
Relief-F	a_0,a_1,a_5,a_6,a_8	$a_1,a_5,a_6,\_,\boldsymbol{r}$	a_1,a_2,a_5	$\_,a_2,a_3,a_4,\_,a_6$	$a_2,\_,a_5$
FSBAR	$a_0,a_1,\boldsymbol{a_3},a_5,a_6,a_8$	$\boldsymbol{a_0},a_1,a_5,a_6,a_8$	$\_,\_,a_5$	$a_1,\_,\_,\_,\_,\_$	$a_2,\_,a_5$
INTERACT	a_0,a_1,a_5,a_6,a_8	$a_1,\boldsymbol{a_3},\boldsymbol{a_4},a_5,a_6,\boldsymbol{a_7},\_$	a_1,a_2,a_5	a_1,a_2,a_3,a_4,a_5,a_6	a_2,a_4,a_5
FRFS	a_0,a_1,a_5,a_6,a_8	a_1,a_5,a_6,a_8	a_1,a_2,a_5	a_1,a_2,a_3,a_4,a_5,a_6	a_2,a_4,a_5
Relevant features	a_0,a_1,a_5,a_6,a_8	a_1,a_5,a_6,a_8	a_1,a_2,a_5	a_1,a_2,a_3,a_4,a_5,a_6	a_2,a_4,a_5

Table 5
Number of selected features for the six feature selection algorithms.

Data name	FullSet	CFS	FCBF	Consistency	Relief-F	INTERACT	FRFS
Internet advertisements	1558	–	75	38	148	51	35
isol5	617	134	32	12	503	24	64
mfeat-fac	216	73	39	8	215	12	45
mfeat-fou	76	38	37	11	38	14	34
multiple-features	649	144	129	7	577	8	23
spambase	57	15	14	25	15	27	29
movement_libras	90	26	10	19	70	18	39
ORL10P	10 304	–	261	3	8906	5	14
AR10P	2400	46	25	6	654	7	27
PIE10P	2420	–	48	5	1350	7	22
Arrhythmia	279	22	12	27	24	22	48
Audiology (standardized)	69	16	16	13	16	14	27
splice	60	22	22	10	28	11	13
tiger	231	28	9	20	None	19	35
CLL-SUB-111	11 340	–	87	8	1967	17	14
TOX-171	5748	–	80	10	804	14	29
SMK-CAN-187	19 993	–	49	10	113	16	24
Gloabel cancer map	16 063	–	67	9	5571	13	25
leukemia_test	7129	23	1	1	567	1	1
leukemia_train	7129	34	1	1	864	1	1
Musk(Version1)	167	38	11	20	59	18	37
spectrometer	102	20	7	15	94	16	23
BASEHOCK	4862	50	44	64	61	65	26
fibs.wc	2000	–	26	36	402	50	69
la2s.wc	12 433	–	37	45	119	54	87
oh0.wc	3182	37	20	67	89	61	75
oh5.wc	3012	21	20	52	73	48	56
re0.wc	2886	28	18	45	103	42	72
tr12.wc	5804	16	14	18	105	17	27
tr21.wc	7903	27	12	15	214	15	23
tr23.wc	5832	13	12	10	334	11	13
tr31.wc	10 128	30	30	18	414	19	25
tr41.wc	7454	34	28	17	385	20	31
tr45.wc	8261	26	20	17	554	23	27
wap.wc	8461	55	41	33	271	38	84
Average	4826.14	39.08	38.69	20.43	756.09	22.80	34.97

algorithms in terms of (i) the number of selected features; (ii) the runtime of feature selection; and (iii) the classification accuracy over the selected feature subset, respectively.

It should be noted that FSBAR was not available on up to 30 out of 35 data sets since its high time consumption for high dimensional data. Therefore, we can only compare FSBAR with FRFS over the remaining five data sets in a separate Section 5.4.4.

5.4.1. Number of selected features

Table 5 reports the number of features selected by the six algorithms CFS, FCBF, Consistency, Relief-F, INTERACT and FRFS on the 35 data sets. The number of full features for each data set is recorded as well. In this table, '–' denotes that the feature subset selection algorithm is not available on the corresponding data set.

From Table 5 we observe that (i) the number of features selected by each of the six algorithms is much smaller than that of the full features. This means all the six feature subset selection algorithms could significantly reduce the number of features. (ii) The average number of selected features obtained by FRFS is less than those obtained by CFS, FCBF and Relief-F. (iii) Only four algorithms FCBF, Consistency, INTERACT, and FRFS are available for all the 35 data sets.

In Table 5, note that for data sets "leukemia test" and "leukemia train", only one feature is returned for algorithms FCBF,, Consistency, INTERACT and FRFS. The selected feature is not the label vector. For data set "leukemia test", the selected feature is "M31523_at" whose two values just correspond to the two target concept values and can be used to clearly distinguish the target concept values. The similar for the selected feature "X95735_at" of data set "leukemia train". This is the reason why only one feature returned on these two data sets but with 100% classification accuracy for different classifiers (see Tables 7, 8, 9 and 10).

5.4.2. Runtime

Table 6 records the runtime of the six feature subset selection algorithms CFS, FCBF, Consistency, Relief-F, INTERACT and FRFS on the 35 data sets. From it we observe that our proposed FRFS is much faster than any of the other five algorithms, its average runtime is only 7.93% of that of CFS, 12.25% of that of FCBF, 0.56% of that of Consistency, 1.05% of that of Relief-F and 2.68% of that of INTERACT.

According to the time complexity analysis of FRFS, the time complexity of FRFS is between $O(K \cdot N)$ and $O(K \cdot N^2)$, where K is the number of features, and N is the number of instances. In contrast, the time complexities of the other feature subset selection algorithms can be uniformly described as $O(N \cdot f(K))$, where $f(K)$ is a function of K. And the order of $f(K)$ is higher than linear order for any other feature subset selection algorithm [10,6,11,18,13]. Moreover, for most of the data sets in Table 3, $N \ll K$. These are the reasons why FRFS is faster than the other algorithms in theory. This agrees well with the practical results.

To sum up, both the time complexity analysis and the empirical research results show that our proposed FRFS algorithm is very efficient, especially on high-dimensional data sets with $N \ll K$.

5.4.3. Classification accuracy

In this section we present the classification accuracies of Naive Bayes, C4.5, PART and IB1 with the six feature subset selection algorithms, respectively. This result can be used to further compare our proposed algorithm with other algorithms.

Table 6
Runtime (in second) for the six feature selection algorithms.

Data name	CFS	FCBF	Consistency	Relief-F	INTERACT	FRFS
Internet advertisements	–	0.05	706.63	1210.67	117.25	2.42
isol5	73.81	0.14	52.86	171.18	18.03	1.17
mfeat-fac	5.12	0.91	13.24	71.47	3.64	0.60
mfeat-fou	0.48	2.12	4.09	26.44	0.78	0.84
multiple-features	45.48	0.06	29.69	216.49	24.68	0.68
spambase	0.95	0.05	11.00	92.81	1.14	1.80
movement_libras	0.27	22.79	0.69	1.18	0.21	0.05
ORL10P	–	0.59	33.35	18.55	198.79	0.22
AR10P	10.06	0.74	1.00	1.94	1.75	0.04
PIE10P	–	0.06	4.02	9.01	13.73	0.09
Arrhythmia	0.33	0.02	2.84	2.61	0.43	0.11
Audiology (standardized)	0.12	0.09	0.27	0.41	0.18	0.03
splice	0.33	0.28	4.71	49.40	1.14	0.72
tiger	0.26	0.35	2.03	–	0.36	0.16
CLL-SUB-111	–	0.16	8.98	4.04	25.50	0.07
TOX-171	–	0.90	5.64	4.96	11.77	0.08
SMK-CAN-187	–	6.04	6.97	5.93	17.91	0.07
Gloabel cancer map	–	3.91	44.03	18.81	132.21	0.30
leukemia_test	0.44	0.03	0.20	0.09	0.53	0.01
leukemia_train	0.82	0.00	0.38	0.16	0.99	0.01
Musk(Version1)	0.54	0.98	1.72	2.52	0.42	0.03
spectrometer	0.24	0.31	1.37	3.50	0.39	0.15
BASEHOCK	15.28	1.65	294.36	196.86	24.29	0.71
fibs.wc	–	0.01	814.54	99.19	80.18	4.45
la2s.wc	–	0.02	1968.71	107.10	188.30	6.91
oh0.wc	4.80	0.81	98.18	2.54	3.44	0.41
oh5.wc	1.18	153.61	38.07	1.87	1.73	0.31
re0.wc	3.42	1.10	67.18	7.82	2.46	0.68
tr12.wc	0.49	0.22	2.40	0.46	0.43	0.03
tr21.wc	1.52	0.14	4.11	1.01	0.80	0.03
tr23.wc	0.81	4.19	2.07	0.36	0.88	0.02
tr31.wc	17.97	3.56	79.84	11.95	27.53	0.23
tr41.wc	12.63	2.44	52.06	9.07	14.09	0.20
tr45.wc	13.21	0.09	48.82	7.58	16.71	0.19
wap.wc	28.54	0.00	194.72	16.51	19.36	1.72
Average	9.20	5.96	131.45	69.84	27.20	0.73

Tables 7, 8, 9, and 10 record the classification accuracies of Naive Bayes, C4.5, PART and IB1 on the 35 data sets, respectively. In these tables, "AverageAcc" and "AverageRank" show the average accuracies and the average ranks of accuracies, respectively.

The average ranks are computed based on the accuracies except those with the full feature set, and a lower rank value stands for a better feature selection algorithm. That is, for a given data set, the algorithm with the highest accuracy getting the rank of 1, the second highest rank 2,.... In case of ties (such as for "Relief-F" and "INTERACT" corresponding to data set "oh5.wc" in Table 7), average ranks are assigned.

From Table 7 we observe that compared with the full set, the average classification accuracy of Naive Bayes is improved only by FRFS, CFS, FCBF and Relief-F. FRFS is the best in terms of the average accuracy improvement on Naive Bayes. It outperforms CFS by 0.84%, FCBF by 1.74%, Consistency by 6.28%, Relief-F by 4.85% and INTERACT by 6.29%. Meanwhile, the average rank AverageRank shows that FRFS ranks 1.

From Table 8 we observe that compared with the full set, the average classification accuracy of C4.5 is improved by all feature selection algorithms except for INTERACT. FRFS is the best in terms of the average accuracy improvement on C4.5. It outperforms CFS by 2.46%, FCBF by 3.18%, Consistency by 3.45%, Relief-F by 2.22% and INTERACT by 3.85%. Moreover, the average rank AverageRank shows that FRFS outperforms all the other algorithms.

From Table 9 we observe that compared with the full set, the average classification accuracy of PART is improved all feature selection algorithms except for Consistency and INTERACT. FRFS is the best in terms of the average accuracy improvement on PART. It outperforms CFS by 2.69%, FCBF by 3.48%, Consistency by 4.91%, Relief-F by 3.19% and INTERACT by 4.63%. Meanwhile, the average rank AverageRank shows that FRFS ranks 1.

From Table 10 we observe that compared with the full set, the average classification accuracy of IB1 is improved only by FRFS, CFS and Relief-F. FRFS is the best in terms of the average accuracy improvement on IB1. It outperforms CFS by 2.16%, FCBF by 5.13%, Consistency by 7.6%, Relief-F by 1.83% and INTERACT by 5.05%. Once again, FRFS is the number 1 in terms of average rank.

From the above results we can conclude that FRFS is the best one among the six feature subset selection algorithms in terms of both average classification accuracy and average rank for each of the four classification algorithms.

In order to further explore whether or not the improvements of the classification accuracies are statistically significant, the nonparametric Friedman test [42] followed by Holm's sequential Bonferroni test[5] [43] was performed as suggested in [44,45] to statistically compare algorithms on multiple data sets.

Firstly, four Friedman tests were performed individually. And the null hypotheses of these tests are that the six feature subset selection algorithms perform equivalently at the significance level $\alpha = 0.05$ for each of the four classification algorithms. The test results show that all the $p\text{-}value = 0$ being smaller than α. This means that all the four null hypotheses are rejected and the six feature selection algorithms are not equivalent. Next, in order to further explore whether the differences are produced by which FRFS is statistically better than the other algorithms, four sequential Bonferroni tests were conducted for each of the four classification algorithms. To better understand the test results, we briefly introduce the sequential Bonferroni test as follows.

[5] Holm's sequential Bonferroni test is also referred to as Bonferroni–Holm procedure.

Table 7
Classification accuracy of Naive Bayes with the six feature selection algorithms.

Data name	FullSet	CFS	FCBF	Consistency	Relief-F	INTERACT	FRFS
Internet advertisements	96.77	–	96.85	96.78	96.16	96.54	96.24
isol5	90.29	91.08	84.70	71.88	89.96	68.12	89.18
mfeat-fac	93.87	96.25	95.25	86.28	93.90	86.69	94.85
mfeat-fou	79.00	80.17	79.98	76.62	79.66	79.47	80.35
multiple-features	96.72	98.50	97.97	93.08	96.52	92.09	95.83
spambase	90.22	92.68	92.08	89.09	92.65	89.76	90.11
movement_libras	67.28	68.67	66.17	62.33	65.78	64.72	69.78
ORL10P	100.00	–	100.00	84.20	100.00	89.60	99.00
AR10P	90.46	95.08	90.62	74.15	90.46	80.31	92.62
PIE10P	96.48	–	99.90	87.33	96.76	84.76	97.14
Arrhythmia	75.22	78.32	75.93	77.39	74.78	75.75	77.43
Audiology (standardized)	72.65	74.07	73.01	68.76	73.89	68.98	72.74
splice	95.39	96.17	96.17	94.42	94.17	94.60	95.21
tiger	77.93	79.11	78.41	76.54	–	75.05	77.74
CLL-SUB-111	87.57	–	98.20	93.87	83.96	88.23	94.23
TOX-171	89.36	–	99.30	80.00	89.36	71.67	93.57
SMK-CAN-187	71.02	–	91.66	78.93	86.42	81.32	83.74
Gloabel cancer map	64.44	–	90.00	69.44	64.44	63.70	87.78
leukemia_test	100.00	100.00	100.00	100.00	100.00	100.00	100.00
leukemia_train	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Musk(Version1)	86.22	90.76	90.29	86.05	88.24	88.91	86.55
spectrometer	53.48	55.86	48.93	53.37	53.48	54.16	56.01
BASEHOCK	96.85	96.00	94.52	96.42	94.33	96.69	92.03
fibs.wc	70.66	–	74.35	72.59	73.41	71.74	83.39
la2s.wc	86.95	–	74.34	76.61	81.85	80.01	83.92
oh0.wc	90.87	85.74	76.39	88.63	87.54	86.06	88.97
oh5.wc	86.08	80.57	79.37	86.71	84.97	84.97	88.37
re0.wc	75.98	75.97	74.76	77.78	77.39	74.53	80.40
tr12.wc	91.95	88.88	86.20	88.95	91.37	87.20	91.63
tr21.wc	68.33	95.95	92.32	89.64	68.75	91.24	94.40
tr23.wc	56.18	96.27	93.04	94.02	56.37	97.07	95.59
tr31.wc	92.77	97.67	96.55	95.69	93.53	97.30	97.41
tr41.wc	92.07	96.63	94.56	88.95	90.89	92.23	96.04
tr45.wc	79.65	94.32	91.07	86.70	77.39	87.28	95.51
wap.wc	82.15	77.14	71.65	71.22	81.60	73.33	79.82
AverageACC	84.14	87.76	86.99	83.27	84.41	83.26	88.50
AverageRank	–	3.10	3.14	4.34	3.80	4.19	2.43

Table 8
Classification accuracy of C4.5 with the six feature selection algorithms.

Data name	FullSet	CFS	FCBF	Consistency	Relief-F	INTERACT	FRFS
Internet advertisements	96.73	–	96.47	96.99	96.89	96.46	96.96
isol5	70.59	74.62	68.41	58.00	72.26	56.76	76.31
mfeat-fac	82.26	82.36	81.65	78.43	82.51	79.62	81.97
mfeat-fou	69.40	70.75	70.64	69.16	70.96	70.61	71.11
multiple-features	91.18	89.13	88.72	87.42	91.45	89.80	90.09
spambase	93.02	92.59	92.85	92.57	92.76	92.43	92.77
movement_libras	65.61	63.28	64.00	64.94	64.44	65.83	66.89
ORL10P	73.20	–	86.60	81.80	83.20	86.00	92.80
AR10P	82.77	77.85	85.69	70.62	86.00	80.31	84.62
PIE10P	82.19	–	80.86	85.43	82.67	83.05	83.33
Arrhythmia	72.88	72.96	74.47	73.72	72.30	71.37	72.88
Audiology (standardized)	77.35	77.88	77.70	76.11	77.88	74.20	78.94
splice	94.14	94.34	94.31	93.91	94.18	94.33	94.51
tiger	78.36	78.46	77.59	78.10	–	77.69	79.16
CLL-SUB-111	79.46	–	87.75	90.63	80.72	74.20	89.91
TOX-171	76.26	–	81.75	76.02	70.41	74.12	78.01
SMK-CAN-187	78.82	–	81.71	82.14	78.18	78.92	82.67
Gloabel cancer map	67.64	–	66.53	61.67	69.58	59.13	75.69
leukemia_test	88.82	89.41	100.00	100.00	100.00	100.00	100.00
leukemia_train	91.58	91.58	100.00	100.00	100.00	100.00	100.00
Musk(Version1)	87.23	89.33	88.19	86.81	86.55	91.22	90.42
spectrometer	50.51	51.64	49.27	56.31	51.07	54.40	55.52
BASEHOCK	94.81	93.49	93.08	94.41	92.17	93.44	91.14
fibs.wc	74.43	–	71.25	69.94	75.80	71.98	79.15
la2s.wc	79.10	–	72.85	77.24	77.85	77.61	81.00
oh0.wc	84.21	84.29	76.21	84.97	84.95	84.76	85.50
oh5.wc	85.27	82.88	80.17	86.21	85.95	85.97	85.95
re0.wc	77.29	75.88	74.18	78.47	77.46	76.24	78.88
tr12.wc	85.37	81.09	81.41	85.43	86.90	80.43	84.79
tr21.wc	89.88	91.25	92.08	90.95	88.10	90.37	91.19
tr23.wc	92.65	94.51	92.16	97.06	93.63	97.06	95.10
tr31.wc	94.43	95.51	96.55	96.20	95.25	95.90	96.33
tr41.wc	93.05	93.62	91.78	88.54	92.48	92.33	94.85
tr45.wc	92.81	91.36	88.90	89.13	91.45	88.17	93.30
wap.wc	68.36	66.95	64.23	63.29	68.08	66.77	69.77
AverageAcc	81.76	82.58	82.00	81.79	82.77	81.47	84.61
AverageRank	–	4.27	3.83	3.53	3.46	3.96	1.96

Table 9
Classification accuracy of PART with the six feature selection algorithms.

Data name	FullSet	CFS	FCBF	Consistency	Relief-F	INTERACT	FRFS
Internet advertisements	96.58	–	96.39	97.33	96.80	96.99	97.43
isol5	73.20	73.07	68.54	57.69	71.39	56.92	75.28
mfeat-fac	85.20	85.76	84.32	79.18	88.29	80.54	84.64
mfeat-fou	70.70	72.58	72.03	69.15	75.22	71.46	72.54
multiple-features	92.33	93.24	91.98	86.37	87.47	87.80	93.10
spambase	93.65	93.31	93.43	92.91	93.43	92.74	93.36
movement_libras	66.72	63.39	65.89	63.17	68.40	64.50	67.61
ORL10P	76.20	–	85.00	79.60	81.20	86.20	91.40
AR10P	76.15	78.77	81.08	69.08	80.58	76.77	82.31
PIE10P	83.24	–	86.67	84.67	78.03	82.67	91.33
Arrhythmia	71.99	72.43	74.20	72.12	75.95	70.57	72.08
Audiology (standardized)	79.38	77.52	77.52	72.39	77.43	74.38	79.82
splice	92.55	93.13	93.17	93.39	84.73	92.90	93.04
tiger	78.87	77.13	78.64	77.87	–	77.72	79.38
CLL-SUB-111	78.92	–	86.85	89.19	83.27	80.89	91.71
TOX-171	78.13	–	83.27	78.36	73.27	73.06	78.25
SMK-CAN-187	80.32	–	85.13	82.14	76.04	79.45	84.92
Gloabel cancer map	65.69	–	64.31	61.25	66.44	58.03	76.67
leukemia_test	88.82	89.41	100.00	100.00	89.72	100.00	100.00
leukemia_train	91.58	91.58	100.00	100.00	100.00	100.00	100.00
Musk(Version1)	89.33	89.24	87.94	89.12	87.82	91.14	89.41
spectrometer	50.51	52.54	48.78	52.05	58.66	53.97	53.26
BASEHOCK	95.66	94.82	93.64	96.40	93.33	96.22	91.83
fibs.wc	–	–	69.73	67.41	76.00	68.90	79.50
la2s.wc	–	–	74.67	76.62	80.98	78.88	83.76
oh0.wc	82.33	83.65	75.27	84.93	83.75	83.73	85.38
oh5.wc	84.27	82.85	80.41	85.71	85.19	85.16	85.66
re0.wc	77.79	75.78	73.95	77.18	79.06	76.49	78.78
tr12.wc	84.79	83.00	83.39	85.62	82.11	83.43	86.39
tr21.wc	90.54	90.71	92.50	91.55	91.07	92.03	91.90
tr23.wc	90.98	93.53	93.24	97.25	93.14	95.47	95.69
tr31.wc	94.63	96.63	96.09	97.13	96.33	95.69	96.81
tr41.wc	92.69	93.58	90.89	88.61	92.14	91.64	94.33
tr45.wc	92.29	90.84	89.19	86.49	91.45	88.06	93.19
wap.wc	70.46	70.59	65.78	62.77	70.64	68.17	73.69
AverageAcc	82.32	83.04	82.40	81.28	82.63	81.50	85.27
AverageRank	–	4.35	3.49	3.76	3.46	3.96	1.99

* In the column "FullSet", '–' denotes that the classifier is so time consumption and not available on the corresponding data set due to the large number of features.

The sequential Bonferroni test performs more than one hypothesis test simultaneously. Suppose that there are k algorithms being compared with a control algorithm (e.g. FRFS in the experiment) at an initial significance level α, and let $p_1,p_2,\ldots,p_k$ be ordered (from the smallest to the largest) *p-values* and $H_1,H_2,\ldots,H_k$ be the corresponding null hypotheses, where each hypothesis supposes that there is no difference between the corresponding algorithm and the control algorithm. Holm's sequential Bonferroni test starts with the most significant (the smallest) p-value. If $p_1 < \alpha/k$, H_1 is rejected and we are allowed to compare p_2 with $\alpha/(k-1)$. This process continues until a certain hypothesis H_i $(1 \le i \le k)$ cannot be rejected, and all the remaining hypotheses after H_i are retained.

Table 11 shows the results of Holm's sequential Bonferroni tests at the significance level $\alpha = 0.05$. In the table, α_{HM} denotes the adjusted significance level. If the *p-value* is less than the corresponding α_{HM}, then the alternative hypothesis is accepted.

From Table 11 we can observe that:

1. For Naive Bayes, the *p-values* of the pairs (Relief-F, FRFS), (INTERACT, FRFS) and (Consistency, FRFS) are all smaller than the corresponding α_{HM}, while the *p-values* of the pairs (FCBF, FRFS) and (CFS, FRFS) are greater than the corresponding α_{HM}. This means that FRFS is statistically better than Relief-F, Consistency and INTERACT in terms of accuracy improvement on Naive Bayes; and the accuracy improvement of FRFS is not statistically significant compared with those of FCBF and CFS. This may be due to that Naive Bayes assumes conditional independence among features [46], thus interactive features pose negative impact on it. These results conform to the analysis of Naive Bayes.
2. For C4.5, PART and IB1, the *p-values* of the pairs (Relief-F, FRFS), (Consistency, FRFS), (INTERACT, FRFS), (CFS, FRFS) and (FCBF, FRFS) are all smaller than the corresponding α_{HM}, so all the alternative hypotheses are accepted. This states that FRFS is statistically better than all the other five algorithms in terms of accuracy improvements on these three classification algorithms. That is, the proposed rule-based algorithm FRFS is superior to the other representative feature selection algorithms for not only rule-based classifier PART, but also tree-based C4.5 and instance-based IB1.

5.4.4. Comparison of FRFS and FSBAR

Since FSBAR was not available on 30 out of the 35 data sets due to its high time complexity, we compared FRFS and FSBAR over the remaining five data sets in terms of the number of selected features, run time and the classification accuracies of Naive Bayes, C4.5, PART, and IB1 after feature selection. Table 12 shows the details.

From Table 12 we observe that although the number of features selected by FRFS is greater than that of FSBAR, its runtime is a very small fraction of that of FSBAR. Moreover, FRFS outperforms FSBAR by 25.18% for Naive Bayes, 20.94% for C4.5,

Table 10
Classification accuracy of IB1 with the six feature selection algorithms.

Data name	FullSet	CFS	FCBF	Consistency	Relief-F	INTERACT	FRFS
Internet advertisements	–	–	93.92	96.94	96.07	97.23	96.37
isol5	86.62	86.38	70.34	52.09	87.99	64.46	84.71
mfeat-fac	95.69	96.92	95.12	81.22	95.66	85.99	95.45
mfeat-fou	76.32	79.02	77.76	69.87	78.77	78.72	79.62
multiple-features	97.77	98.57	98.48	84.84	97.75	92.03	95.98
spambase	92.21	90.69	91.53	91.68	90.18	92.86	92.17
movement_libras	77.39	73.89	66.83	74.50	77.11	76.94	76.56
ORL10P	100.00	–	100.00	88.40	100.00	91.00	99.00
AR10P	96.00	98.00	92.62	80.92	96.00	88.62	92.15
PIE10P	99.62	–	100.00	88.95	99.62	86.86	98.67
Arrhythmia	66.95	70.04	66.11	69.34	63.89	68.94	67.39
Audiology (standardized)	74.78	72.83	73.36	71.86	72.12	75.28	79.12
splice	75.98	81.64	81.64	85.29	87.19	84.92	84.95
tiger	81.56	77.07	71.51	78.16	–	77.23	80.33
CLL-SUB-111	84.68	–	97.48	87.93	84.32	82.91	94.77
TOX-171	96.02	–	99.53	79.06	97.08	77.07	93.57
SMK-CAN-187	78.07	–	81.39	84.17	83.85	82.66	88.13
Gloabel cancer map	75.97	–	88.89	63.33	75.28	66.96	83.33
leukemia_test	100.00	100.00	100.00	100.00	100.00	100.00	100.00
leukemia_train	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Musk(Version1)	89.41	86.05	83.74	88.74	90.55	88.79	90.92
spectrometer	59.85	61.09	46.59	56.72	59.85	56.72	61.05
BASEHOCK	90.75	92.36	90.71	94.53	91.77	93.91	88.05
fibs.wc	–	–	64.30	66.40	73.57	71.14	80.71
la2s.wc	66.19	–	70.48	76.69	80.07	78.79	82.52
oh0.wc	73.46	78.40	67.68	78.01	76.37	77.38	79.10
oh5.wc	73.64	77.19	74.92	78.15	77.12	80.35	81.35
re0.wc	79.55	76.56	69.64	77.93	80.85	78.38	81.18
tr12.wc	85.62	87.03	83.26	84.86	85.94	85.79	90.22
tr21.wc	91.61	96.61	88.10	92.20	91.96	93.26	93.93
tr23.wc	91.86	93.43	93.24	95.98	90.20	96.37	94.31
tr31.wc	91.97	96.91	94.24	95.71	94.39	98.58	96.74
tr41.wc	88.43	94.37	91.55	86.33	89.41	92.87	94.81
tr45.wc	85.01	91.68	85.80	82.61	86.38	83.97	93.22
wap.wc	66.04	70.59	62.90	63.58	70.96	68.88	72.96
AverageAcc	84.52	85.67	83.25	81.34	85.95	83.31	87.52
AverageRank	–	3.67	4.31	4.01	3.33	3.41	2.26

* In the column "FullSet", '–' denotes that the classifier is so time consumption and not available on the corresponding data set due to the large number of instances or features.

Table 11
Holm's sequential Bonferroni test results on accuracy differences of four classifiers.

Alternative hypothesis	Naive Bayes		C4.5		PART		IB1	
	p-value	α_{HM}	*p-value*	α_{HM}	*p-value*	α_{HM}	*p-value*	α_{HM}
CFS < FRFS	0.0666	0.05	0.0000	0.01	0.0000	0.01	0.0008	0.0167
FCBF < FRFS	0.0551	0.025	0.0000	0.0167	0.0004	0.025	0.0000	0.01
Consistency < FRFS	0.0000	0.01	0.0002	0.025	0.0000	0.0167	0.0000	0.0125
Relief-F < FRFS	0.0011	0.0125	0.0004	0.05	0.0005	0.05	0.0083	0.05
INTERACT < FRFS	0.0000	0.0167	0.0000	0.0125	0.0000	0.0125	0.0048	0.025

Table 12
Comparison between FSBAR and FRFS.

Data name	Features		Runtime (s)		Naive Bayes		C4.5		PART		IB1	
	FSBAR	FRFS	FSBAR	FRFS	FSBAR	FRFS	FSBAR	FRFS	FSBAR	FRFS	FSBAR	FRFS
multiple-features	3	23	1328.31	0.68	42.43	95.83	45.83	90.09	45.54	93.10	33.63	95.98
spambase	20	29	3414.41	1.80	85.76	90.11	88.28	92.77	88.76	93.36	84.07	92.17
splice	9	13	57.44	0.72	86.24	95.21	86.26	94.51	85.64	93.04	79.14	84.95
tiger	18	35	590.25	0.16	71.54	77.74	75.05	79.16	74.08	79.38	70.16	80.33
spectrometer	16	23	548.09	0.15	45.50	56.01	45.28	55.52	43.81	53.26	39.52	61.05
Average	13.20	24.60	1187.70	0.70	66.29	82.98	68.14	82.41	67.57	82.43	61.30	82.90

21.99% for PART, and 35.22% for IB1 in terms of the classification accuracy, respectively.

As there are only five data sets, the sample size is too small to guarantee the validity of statistical test, thus we did not perform the statistical test for the accuracy difference comparisons.

6. Conclusion

In this paper, we present a novel propositional FOIL rule based feature subset selection algorithm, which is very applicable, especially to high-dimensional data. This algorithm is proposed for not only identifying and removing irrelevant and redundant features, but also dealing with interactive features.

We first defined relevant, redundant and interactive features based on classification rules. Then based on these definitions, we presented the feature selection algorithm, which involves two steps (i) redundant feature exclusion and interactive feature reservation and (ii) the irrelevant feature identification. We also explained why these two steps are able to exclude redundant as well as irrelevant features and reserve interactive features with the help of the propositional FOIL rules generated by the restricted FOIL algorithm.

We have compared our proposed FRFS algorithm with the other six representative feature subset selection algorithms in terms of the number of selected features, runtime and accuracy of four well-known classifiers such as Naive Bayes, C4.5, PART, and IB1 on both the five synthetic data sets and the 35 real world high-dimensional data sets. The experimental results of synthetic data sets show that FRFS can effectively identify the relevant features while eliminating redundant features and reserving interactive features. The results of the real world data sets show that our proposed algorithm has moderate reduction capability. Meanwhile, it is much faster than the other six feature selection algorithms, especially on high-dimensional data. Moreover, our proposed FRFS algorithm obtains the best average accuracies for all the four classification algorithms.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant 61070006.

References

[1] L.C. Molina, L. Belanche, À. Nebot, Feature selection algorithms: a survey and experimental evaluation, in: Proceedings of IEEE International Conference on Data Mining, IEEE Computer Society, 2002, pp. 306–313.
[2] G.H. John, R. Kohavi, K. Pfleger, Irrelevant features and the subset selection problem, in: Proceedings of the Eleventh International Conference on Machine Learning, 1994.
[3] L. Yu, H. Liu, Redundancy based feature selection for microarray data, in: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2004, pp. 737–742.
[4] G. Forman, An extensive empirical study of feature selection metrics for text classification, Journal of Machine Learning Research 3 (2003) 1289–1305.
[5] K. Kira, L.A. Rendell, The feature selection problem: traditional methods and a new algorithm, in: Proceedings of the National Conference on Artificial Intelligence, American Association for Artificial, 1992, pp. 129–134.
[6] I. Kononenko, Estimating attributes: analysis and extensions of RELIEF, in: Proceedings of the 1994 European Conference on Machine Learning, Springer, 1994, pp. 171–182.
[7] H. Park, H.C. Kwon, Extended relief algorithms in instance-based feature filtering, in: Proceedings of the Sixth International Conference on Advanced Language Processing and Web Information Technology, IEEE Computer Society, 2007, pp. 123–128.
[8] M. Scherf, W. Brauer, Feature Selection by Means of a Feature Weighting Approach, Technical Report, Institut fur Informatik, Technische Universitat Munchen, 1997.
[9] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Transactions on Neural Networks 5 (4) (1994) 537–550.
[10] M.A. Hall, Correlation-based feature selection for discrete and numeric class machine learning, in: Proceedings of the Seventeenth International Conference on Machine Learning, 2000, pp. 359–366.
[11] H. Liu, R. Setiono, A probabilistic approach to feature selection-a filter solution, in: Proceedings of the 13th International Conference on Machine Learning, 1996, pp. 319–327.
[12] M. Modrzejewski, Feature selection using rough sets theory, in: Proceedings of the European Conference on Machine Learning, Springer, 1993, pp. 213–226.
[13] L. Yu, H. Liu, Feature selection for high-dimensional data: a fast correlation-based filter solution, in: Proceedings of 20th International Conference on Machine Learning, vol. 20, 2003, pp. 856–863.
[14] Q. Song, J. Ni, G. Wang, A fast clustering-based feature subset selection algorithm for high dimensional data, IEEE Transactions on Knowledge and Data Engineering (99) (2011) 1–14.
[15] A. Jakulin, I. Bratko, Analyzing attribute dependencies, in: Proceedings of Seventh European Conference on Principles and Practice of Knowledge Discovery in Databases, Springer, 2003, pp. 229–240.
[16] A. Jakulin, I. Bratko, Testing the significance of attribute interactions, in: Proceedings of the Twenty-First International Conference on Machine Learning, ACM, 2004, pp. 409–416.
[17] Z. Zhao, H. Liu, Searching for interacting features in subset selection, Intelligent Data Analysis 13 (2) (2009) 207–228.
[18] J. Xie, J. Wu, Q. Qian, Feature selection algorithm based on association rules mining method, in: 2009 Eighth IEEE/ACIS International Conference on Computer and Information Science, IEEE, 2009, pp. 357–362.
[19] R. Agrawal, R. Srikant, Fast algorithms for mining association rules, in: Proceedings 20th International Conference on Very Large Data Bases, vol. 1215, 1994, pp. 487–499.
[20] J.R. Quinlan, R.M. Cameron-Jones, FOIL: a midterm report, in: Proceedings of the European Conference on Machine Learning, Springer, 1993, pp. 1–20.
[21] W.W. Cohen, Fast effective rule induction, in: Proceedings of the Twelfth International Conference on Machine Learning, Morgan Kaufmann, 1995, pp. 115–123.
[22] X. Yin, J. Han, CPAR: classification based on predictive association rules, in: Proceedings of the Third SIAM International Conference on Data Mining, Society for Industrial & Applied, 2003, pp. 331–335.
[23] M. Dash, H. Liu, Feature selection for classification, Intelligent Data Analysis 1 (3) (1997) 131–156.
[24] J. Souza, Feature Selection with a General Hybrid Algorithm, Ph.D. Thesis, University of Ottawa, 2005.
[25] A.L. Blum, P. Langley, Selection of relevant features and examples in machine learning, Artificial Intelligence 97 (1–2) (1997) 245–271.
[26] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of Machine Learning Research 3 (2003) 1157–1182.
[27] D. Koller, M. Sahami, Toward optimal feature selection, in: Proceedings of International Conference on Machine Learning, 1996, pp. 284–292.
[28] M. ElAlami, A filter model for feature subset selection based on genetic algorithm, Knowledge-Based Systems 22 (5) (2009) 356–362.
[29] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artificial Intelligence 97 (1–2) (1997) 273–324.
[30] M.A. Hall, Correlation-Based Feature Selection for Machine Learning, Ph.D. Thesis, University of Waikato, 1999.
[31] L. Yu, H. Liu, Efficient feature selection via analysis of relevance and redundancy, Journal of Machine Learning Research 5 (2004) 1205–1224.
[32] F. Fleuret, Fast binary feature selection with conditional mutual information, Journal of Machine Learning Research 5 (2004) 1531–1555.
[33] I.A. Gheyas, L.S. Smith, Feature subset selection in large dimensionality domains, Pattern Recognition 43 (1) (2010) 5–13.
[34] P. Chanda, Y.R. Cho, A. Zhang, M. Ramanathan, Mining of attribute interactions using information theoretic metrics, in: Proceedings of the 2009 IEEE International Conference on Data Mining Workshops, IEEE Computer Society, 2009, pp. 350–355.
[35] A. Asuncion, D. Newman, UCI machine learning repository, 2007.
[36] U. Fayyad, K. Irani, Multi-interval discretization of continuous-valued attributes for classification learning, in: Proceedings of Thirteenth International Joint Conference on Artificial Intelligence, 1993, pp. 1022–1027.
[37] G.H. John, P. Langley, Estimating continuous distributions in Bayesian classifiers, in: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, vol. 1, Citeseer, 1995, pp. 338–345.
[38] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, 1993.
[39] E. Frank, I.H. Witten, Generating accurate rule sets without global optimization, in: Proceedings of the Fifteenth International Conference on Machine Learning, 1998, pp. 144–151.
[40] D.W. Aha, D. Kibler, M.K. Albert, Instance-based learning algorithms, Machine Learning 6 (1) (1991) 37–66.
[41] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: International Joint Conference on Artificial Intelligence, vol. 14, Citeseer, 1995, pp. 1137–1145.
[42] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, The Annals of Mathematical Statistics 11 (1) (1940) 86–92.
[43] S. Holm, A simple sequentially rejective multiple test procedure, Journal of Statistics 6 (2) (1979) 65–70.

[44] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.
[45] S. García, F. Herrera, An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons, Journal of Machine Learning Research 9 (2008) 2677–2694.
[46] I. Rish, J. Hellerstein, T. Jayram, An analysis of data characteristics that affect naive Bayes performance, in: Proceedings of the Eighteenth Conference on Machine Learning, Morgan Kaufmann, 2001.

Guangtao Wang received the BS degree in computer science from Xi'an Jiaotong University, Xi'an, China, in 2007. He is currently a Ph.D. student in the Department of Computer Science and Technology, Xi'an Jiaotong University. His research focuses on feature subset selection and meta-learning.

Qinbao Song received the Ph.D. degree in computer science from Xi'an Jiaotong University, Xi'an, China, in 2001. He is currently a Professor of software technology in the Department of Computer Science and Technology, Xi'an Jiaotong University, where he is also the Deputy Director of the Department of Computer Science and Technology. He is also with the State Key Laboratory of Software Engineering, Wuhan University, Wuhan, China. He has authored or coauthored more than 80 referred papers in the areas of machine learning and software engineering. He is a board member of the Open Software Engineering Journal. His current research interests include data mining/machine learning, empirical software engineering, and trustworthy software.