ISSN 2320 - 2602
Volume 3, No.2, February 2014
International Journal of Advances in Computer Science and Technology
Available Online at http://warse.org/pdfs/2014/ijacst04322014.pdf

Hybrid Rule Based Feature Subset Selection and Classification

M.Geetharani [1], S.Reha [2]
[1]Assistant Professor, Department of Information Technology, Sri Eshwar College Of Engineering, Coimbatore, Tamilnadu, India, geethaitsece@gmail.com.
[2]Assistant Professor, Department of Information Technology, Sri Eshwar College Of Engineering, Coimbatore, Tamilnadu, India, rekhasece@gmail.com.

ABSTRACT

Feature subset selection can be analyzed as the practice of identifying and removing as a lot of inappropriate and unnecessary features as achievable. This is for the reason that, irrelevant features do not contribute to the predictive accuracy and redundant features do not redound to receiving a better analyst for that they provide typically information which is previously present in other features. To deal with these we develop a novel algorithm called as a fast clustering-based feature selection algorithm which can efficiently and effectively obtain a good feature subset. One more an important issue in feature subset selection is Feature interaction. Still, the majority of the presented algorithms only spotlight on dealing with irrelevant and redundant features. In this paper, a hybrid FOIL Rule based Feature subset Selection algorithm (FRFS) is developed, with combination of RIPPER algorithm which not only retains relevant features and eliminates irrelevant and redundant ones but also reflects on feature interaction for high dimensional data. First, FRRFS(FOIL Rules RIPPER Based Feature Selection) combines the features emerged in the predecessors of the entire FOIL rules by using RIPPER it generates best FOIL rules for both negative and positive rules by achieving a candidate feature subset which eliminates redundant features and reserves interactive ones. After that, it identifies and eliminates irrelevant features by evaluating features in the candidate feature subset with a new metric Cover Ratio, and achieves the final feature subset. The experimented results shows that the efficiency and effectiveness of FRRFS(FOIL Rules RIPPER Based Feature Selection) upon both synthetic and real world data sets, and it is evaluated with other feature subset selection algorithms and the classification accuracies before and after feature selection.

Keywords: irrelevant and redundant features, fast clustering-based feature selection algorithm, feature subset selection, FOIL Rule based Feature subset Selection algorithm, RIPPER ((Rule Inductive learning algorithm).

1. INTRODUCTION

Feature subset selection is a significant research concern in the domains of machine learning and data mining, which is the procedure of identifying the nearly everyone salient features for learning. Its principle is to spotlight a learning algorithm on the aspects of the irrelevant and redundant features of data most constructive for analysis and prospect forecast. It be able to diminish the dimensionality of the data and may recover a learner moreover in terms of learning concert, generalization competence or model straightforwardness. It advance assists to recognize and better interpret the end results attained by the learner, lessen its volume of storage, reduce the noise generated by irrelevant or redundant features and reduce these ineffective features [2].

Feature subset selection accomplishes its proposed principle by identifying and removing as several irrelevant and redundant features as probable. As irrelevant features are inadequate or yet unconstructive to the predictive accuracy [3], and redundant features, yet if being significant to the learning objective, given that most of the information they hold is previously 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 [3] or/and redundant features [4]. Comparing these algorithms, several of them can efficiently eliminate irrelevant features other than can't identify redundant ones [5–6], a little of them can remove the irrelevant features while taking into the redundant ones into description.

Of course in many feature subset selection algorithms, some can efficiently get rid of irrelevant features but be unsuccessful to handle redundant features [7] up till now some of others can get rid of the irrelevant although taking care of the redundant features [8],[9]. The Based on the MST method, we propose a Fast clustering bAsed feature Selection algoriThm (FAST) is developed. Usually, feature subset selection investigate has spotlighted on hunting for relevant features. A familiar illustration is Relief [10], which considers each feature according to its capability to distinguish occurrences under dissimilar targets based on distance-based criteria function.
Though, Relief is unproductive at removing redundant features as two predictive but highly correlated features are likely both to be extremely weighted [11]. Relief extends to Relief-F [12], enabling this technique to exertion with strident and unfinished data sets and to compact with multiclass troubles, other than still cannot recognize redundant features. Conversely, together with irrelevant features, redundant features as well distress the speed and accuracy of learning algorithms, and hence should be

eradicated too [13], [14]. CFS [15], FCBF [16], and CMIM [17] are instances that get into concern the redundant features. CFS [15] is accomplished by the supposition that a good feature subset is one that contains features highly correlated with the target, thus far uncorrelated with each further.

FCBF [18], [19] is a fast filter technique which can recognize relevant features in addition to redundancy among relevant features devoid of pair wise correlation analysis. CMIM [20] iteratively chooses features which make the most of their mutual information with the class to predict, temporarily to the response of any feature previously selected. Unlike from these algorithms, the FAST algorithm employs the clustering-based technique to decide features. Excluding the identification of irrelevant and redundant features, an imperative other than frequently being disregarded problem is feature interaction [21], which denotes though a single feature is irrelevant to target concept, when combined with other feature(s), it turns out to be very appropriate. On the other hand, it is understandable that they turn out to be very relevant when we join them as one.

This signifies removing interactive features will go ahead to underprivileged predictive accuracy. Consequently, a feature subset selection algorithm should have the capability of eliminating the irrelevant features or/and redundant features even as taking into consideration the feature interaction. A rule-based learning algorithm is FOIL (First Order Inductive Learner) [22], and the FOIL rules preserve be exercised as classification rules [23]. Evaluated with the Apriori algorithm, it is fairly efficient particularly on high dimensional data [24]. Additionally, the hybrid FOIL algorithm employs an extraordinary performance measure (FoilGain) that obtains into account the dissimilar feature fastenings. This denotes it is logical that by means of the FOIL rules are generated based on RIPPER algorithm to choose features for high dimensional data whereas taking into selflessness feature interaction.

The rest of the paper is organized as follows. In Section 2, dealt with the FAST algorithm. In section 3, commenced the classification rule mining technique based on restricted FOIL algorithm with the new feature subset selection algorithm FRRFS(FOIL Rules RIPPER Based Feature Selection). In Section 4, we descried the experimental results. Finally, in Section 5, we reviewed our work and described some conclusions.

2. FAST ALGORITHM

The FAST algorithm logically consists of three steps: 1) removing irrelevant features, 2) building an MST from qualified ones, and 3) separating the MST and choosing representative features. For a data set D with m features $F = \{F_1, F_2 \dots F_m\}$ and class C, we compute the $T-Relevance\ SU(F_i, C)$ value for each feature $F_i (1 \leq i \leq m)$ in the first step. The features whose $SU(F_i, C)$ values are greater than a predefined threshold θ comprise the target-relevant feature subset $F' = \{F_1', F_2' \dots F_k'\}$ $(K \leq m)$. In the following step, we foremost calculate the $F\ \ Correlation\ SU(F_i', F_j')$ value for each pair of features F_i' and F_j' $(F_i', F_j' \in F' \wedge i \neq j)$. Then, viewing features F_i' and F_j' as vertices and $SU(F_i', F_j')$ $(i \neq j)$ as the weight of the edge between vertices F_i' and F_j', a weighted complete graph $G = (V, E)$ is created where $V = \{F_i' | F_i' \in F' \wedge i \in [1, k]\}$ and $E = \{(F_i', F_j') | (F_i', F_j' F' \wedge i, j \in [1, k] \wedge i \neq j)\}$. As symmetric ambiguity is symmetric additional the $F-Correlation\ SU(F_i', F_j')$ is symmetric as well, thus G is an undirected graph.

The absolute graph G reproduces the correlations among all the target-relevant features. Unfortunately, graph G has k vertices and $k(k-1)/2$ edges. For high-dimensional data, it is heavily dense and the edges with different weights are strongly interwoven. Moreover, the decomposition of complete graph is NP-hard [26]. Therefore for graph G, we build an MST, which connects all vertices such that the sum of the weights of the edges is the minimum, using the well known Prim algorithm [54]. The weight of edge (F_i', F_j') is $F-Correlation\ SU(F_i', F_j')$. After building the MST, in the third step, we first remove the edges $E - \{(F_i', F_j') | (F_i', F_j' F' \wedge i, j \in [1, k] \wedge i \neq j)\}$, whose weights are smaller than both of the $T-Relevance\ SU(F_i', C)$ and $SU(F_j', C)$, from the MST. Each deletion results in two disconnected trees T_1 and T_2. Assuming the set of vertices in any one of the final trees to be $V(T)$, we have the property that for each pair of vertices $SU(F_i', F_j') \geq SU(F_i', C)\ \ \vee\ \ SU(F_i', F_j') \geq SU(F_j', C)$ always holds corrected for each $F_i \in S(i \neq j)$, then F_i are redundant features with respect to the given F_j. We know that this property guarantees the features in $V(T)$ are redundant.

For Irrelevant Feature Removal

Step 1: **S**pecified the dataset D as of 1 to m features and division label C. D = $(F_1, F_2, \dots\dots\dots. F_M, C)$ Where F ={F_1;F_2; ... ;F_m} and F_i ($1 \leq i \leq m$)

Step 2: Establish T-relevance SU (F_i; C) value for each feature F_i ($1 \leq i \leq m$). [SU termed as Symmetric uncertainty]

Step 3: If T-relevance is establish to be superior to the predefined threshold value θ, include objective relevant feature subset. $F' = \{F_1', F_2', \dots\dots, F_k'\}$ $(k \leq m)$.

//Minimum Spanning Tress Construction

Step 4: MST graph G is a whole graph and establish it has unacceptable value then precede feature selection.

Step 5: Calculate the F-Correlation $SU(F_i', F_j')$ value for each pair of features F_i' and F_j'

Step 6: Add F_i' *and /or* F_j' to G as vertices with F-correlation $SU(F_i', F_j')$ as the weight of the corresponding edge.

Step 7: Thus a weighted complete graph G (V, E) is constructed.

Step 8: MST is generated using prism algorithm for graph G. All vertices are associated so that the addition of the weights of the edges is the lowest amount, using prism algorithm.

Tree Partition and Representative Feature Selection

Step 9: Remove the edges whose weights are smaller than both of the T-Relevance SU((F_i', C) and SU((F_j', C) from the MST.

Step 10: A Forest is obtained, for each tree in the forest represents a cluster that is denoted as V(T_j).

// Redundancy are Eliminated

Step 11: For each cluster V(T_j) choose a representative feature F_R^j whose T-relevance SU(F_R^j, C) is the greatest.

Step 12: All $F_R^j (j = 1, \ldots [Forest])$ comprise the final feature subset UF_R^j .

3. BASIC FOIL ALGORITHM WITH FRFS

FOIL generates rules to predict classification membership of instances that is a first-order rule learner. Rules are stimulated for each class (i.e. target concept value), one class by the side of a time. At first, for a particular class the instances are divided into two groups: individuals 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 all positive instances other than no pessimistic instances. To finish, through merging the rules created for every class, a set of rules of the known data set is attained.

FOIL first considers all possible rules consisting of a single test (i.e. feature value) When learning an individual rule. It chooses the best of these consistent with FoilGain [22], which supports a test that is true for many positive and not many negative instances. After that FOIL focuses the rule by means of 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 included keen on the rule's antecedent this process continues until the rule covers no negative instances, resulting in a single rule whose antecedent is a combination of examinations. FoilGain is a quantify utilized to estimate the convenience of including a new test, and it is supported on the numbers of positive and negative instances wrapped before and following appending the new investigation. Particularly, suppose a given rule r, and a candidate test (feature value) v that might be included to the antecedent of r. Understand r' is the rule generated by adding test v to rule r.

The assessment $FoilGain(v,r)$ by adding v to r is defined as $FoilGain(v,r) = P^* \cdot \left(log\frac{P^*}{P^*+N^*} - log\frac{P}{P+N}\right)$, where P and N are the numbers of positive and negative instances covered by r, correspondingly. $P^*$and N^* are the numbers of the positive and negative instances wrapped by r', in that order. Firstly FOIL algorithm studies a rule r for a specified target perception value (e.g. Y=0). For every value f_v of feature F_i $(i \in \{1,2,3,red\})$ before the generation of the rule r, the number of positive instances with f_v is equipped that of negative ones, as a result any one is probable to be selected into r's antecedent. Believe $F = 0$ is the first one chosen into r's antecedent, after that the number of positive instances wrapped by r is 2 and the number of negative instances covered by r is 2 additionally.

This indicates merely $F_1 = 0$ is not able to differentiate the different target perception assessments. Consequently, more feature values are required to be chosen into r's antecedent to additional focus the rule based on the FoilGain. Intended for the remaining three features F_{red} ,F_2 and F_3, $FoilGain(F_{red} = 0, r) \cdot \left(\frac{\log 2}{2+2} - \frac{\log 2}{2+2}\right) = 0$, and $FoilGain(F_2 = 0, r) = FoilGain(F_2 = 1, r) = FoilGain(F_3 = 0, r) = FoilGain(F_3 = 1, r) = 1 \cdot \left(\frac{\log 1}{1+1} - \frac{\log 2}{2+2}\right) = 0$

. This denotes that it is unfeasible to differentiate the value of F_{red} as of that of F_2or F_3 by $FoilGain$. Thus, the value of F_{red} may be recognized as the current best value and chosen into the rule's antecedent by fault.

4. FEATURE SUBSET SELECTION ALGORITHM WITH FOIL BASED RIPPER

In this sector, explain our proposed feature subset selection algorithm, FRRFS(FOIL Rules RIPPER Based Feature Selection) , which not merely preserves relevant features and excludes irrelevant and redundant features, but also captures feature interaction into concern. This method consists of two connected steps of the (i) redundant feature exclusion and interactive feature reservation and the (ii) irrelevant feature identification. At the very first, by the restricted-FOIL algorithm classification rule set (CRSet) is extracted from a given data set. The major disadvantages of FOIL rules based only single test for verification ,so it becomes less result for accuracy ,improve the efficiency proposed a RIPPER rule based combined with FOIL algorithm, the rule of each candidate set generated using RIPPER algorithm for each and every feature. CRset is derived using RIPPER algorithm. RIPPER is shortened as Repeated Incremental Pruning to Produce Error Reduction. RIPPER is particularly additional well-organized on huge noisy datasets .There are two kind of circle in Ripper algorithm that is Outer loop and Inner loop. Outer loop adds one FOIL rule at a moment to the FOIL rule base and Inner loop adds one situation at a moment to the present FOIL rule. The information gain determines is making the most of by adding together the circumstances to the FOIL rule. This procedure is continued in anticipation of it covers rejection negative exemplar in the feature subset selection and feature subset are derived using FRFS.

After that, a candidate feature subset is attained by joining the features whose values appeared in the antecedents of the classification rules in CRSet. Even though the candidate subset keeps out redundant features and preserves interactive features, there might be some irrelevant features established at the equivalent occasion. The subsequent step is exercised to recognize and eradicate irrelevant features. In favor of this reason, a new rule-support-based metric, which is submitted to as feature CoverRatio, is defined to evaluate the relevance of a feature to the target conception and for a specific feature, the smaller the CoverRatio is, the more expected it is an irrelevant feature. consequently, in this action, we at most initially compute the CoverRatio for every feature in the candidate feature subset obtained in Step 1, afterward, the features whose CoverRatio is lesser than a predefined threshold are believed as irrelevant features and shifted further than the candidate subset. In the subsequent segments, we first explicate why these two steps can efficiently address corresponding issues in feature selection, then present the pseudo-code of our proposed algorithm and examine its time convolution.

With the aim of assurance the first step of our proposed algorithm is capable of keep out redundant features and reserve interactive ones. 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 enclosed through this rule. That is to say, all the instances including r's antecedent A are positive (i.e. with target concept value y). Additionally, 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. As a result, rule r produced by the restricted FOIL has the following FOIL rule property:

$$\left(\forall A' \supset A, Conf(A' \rightarrow \{y\} < Conf(r)\right) \wedge \left(\forall \hat{A} \supset A, Conf(\hat{A} \rightarrow \{y\} \leq Conf(r)\right).$$

Certain a data set D and a predefined CoverRatio threshold δ, our proposed FRFS algorithm selects a feature subset S from D.

The consequent evocative pseudo-code is shown in Algorithm 1.

Algorithm 2. FRRFS

Inputs: *D: The given data set;*

δ: A predefined threshold

Output: S: the selected feature subset

1 $Define\ S\ and\ FSet \leftarrow \emptyset$

2 $FOIL(D) \leftarrow CRSet$

3 $for\ each\ r\ do$

4 $Ripper(Pos, Neg, k)$

5 $CRset \leftarrow LearnRuleSet(Pos, Neg)$

6 $For\ k\ time$

7 $CRset \leftarrow OptimizeRuleSet(CRset, Pos, Neg)$

8 $LearnCRSet(Pos, Neg)$

9 $CRset \leftarrow \emptyset$

10 $DL \leftarrow DescLen(CRSet, Pos, Neg)$

11 $Repeat$

12 $Rule \leftarrow LearnRule(Pos, Neg)$

13 $Add\ Rule\ to\ CRset$

14 $DL' \leftarrow DescLen(CRSet, Pos, Neg)$

15 $If\ DL' > DL + 6$

16 $PruneCRset(CRset, Pos, Neg)$

17 $Return\ CRset$

18 $If\ DL_1 < DL\ DL \leftarrow DL'$

19 $Delete\ instances\ covered\ from\ Pos\ and\ Neg$

20 $Until\ Pos = \emptyset$

21 $Return\ CRset$

22 $FSet - FSet \cup \{\text{instances containing } r\text{'s antecedent}\}$

23 end

24 $for\ each\ F\ do$

25 $if\ FSet \neq \emptyset\ then$

26 $ratio \leftarrow CoverRatio(F, CRSet)$

27 $if\ ratio > \delta\ then$

28 $S = S \cup \{F\}$

29 end

30 $FSet = FSet - \{F\}$

31 end

32 end

33 $return\ S$

The pseudo-code consists of two divisions. In the first division (rows 1–5), classification rule set CRSet is created from the data set D by function $restricted\_FOIL()$. The task is applied based on the basic FOIL algorithm with the constraint. specifically, when choosing the greatest 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. Subsequently the candidate feature subset $FSet$is attained through combining the features whose values materialized in the antecedents of the rules in CRSet. In the subsequent division (rows 6–14), for every feature F in candidate feature subset $FSet$, the CoverRatio of F is calculated by function $ComputeCoverRatio()$ in proportion to,

$$CoverRation(F_i) = \frac{1}{K}\sum_{f \equiv vSet(F_i)} vCoverRatio(f)$$

based on F and CRSet.

Stipulation 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 occasion. Or else, F is simply eradicated from $FSet$. Do again this procedure 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 perception. Accordingly, a large d is associated with a high probability of removing relevant features. Heuristically the threshold δ is set to be $0.1 \times CoverRatio_{max}$ unless otherwise declared, where CoverRatio the maximum CoverRatio of the feature of candidate feature subset $FSet$. Certainly this threshold also can be regulated using the standard cross-validation.

5. EXPERIMENTAL RESULTS

5.1 Classification accuracy

The efficiency concerns the time required to find a subset of features (heart dataset), the effectiveness is related to the quality of the subset of features. Based on these criteria, FOIL RIPPER Rule based subset selection algorithm selects the features based on the rules generated from FOIL RIPPER Rule is proposed and experimentally evaluated. The proposed method has been estimated using the following measures:

$Accuracy = (TP + TN) / (TP + TN + FP + FN)$
Where
TP = Number of true postive cases
FP = number of false postive cases
TN = Number of true negative cases
FN = Number of false negative cases

In this part present the classification accuracies of FRFS and FOIL with RIPPER algorithms. This outcome can be used to further judge against our proposed algorithm with other algorithms. From Fig. 1, we observe that the accuracy of RIPPER with FOIL is statistically better than with FRFS.

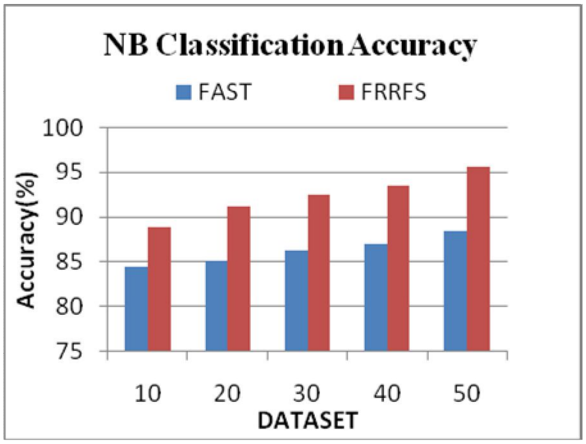

Figure 1: NB (Navie bayes) Classification Accuracy with feature selection

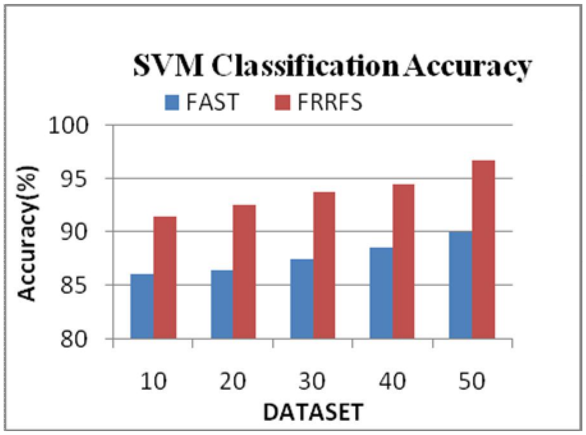

Figure 2: SVM(Support Vector Machine) Classification Accuracy with feature selection

6. CONCLUSION

We presented a new propositional FOIL rule based RIPPER feature subset selection algorithm, which is extremely appropriate, particularly to high-dimensional data. This algorithm is suggested for not only recognizing and removing irrelevant and redundant features, other than moreover dealing with interactive features. To achieve this we first defined relevant, redundant and interactive features based on classification rules. After that based on these concepts, we implemented the feature selection algorithm, which involves two steps (i) redundant feature exclusion and interactive feature reservation and (ii) the irrelevant feature identification. We furthermore clarified why these two steps are capable of eliminate redundant in addition to irrelevant features and reserve interactive features with the help of the propositional FOIL rules produced with the restricted FOIL algorithm similarity within the process also we have evaluated the approach.

REFERENCES

[1] 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.

[2] L.C. Molina, L. Belanche, A. 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.

[3] 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.

[4] 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.

[5] G. Forman, **"An extensive empirical study of feature selection metrics for text classification",** *Journal of Machine Learning Research 3 (2003)* 1289–1305.

[6] M. Scherf, W. Brauer, "**Feature Selection by Means of a Feature Weighting Approach",** *Technical Report, Institut fur Informatik,* Technische Universitat Munchen, 1997.

[7] G. Forman, "**An Extensive Empirical Study of Feature Selection Metrics for Text Classification,**" *J. Machine Learning Research*, vol. 3, pp. 1289-1305, 2003.

[8] R. Battiti, "**Using Mutual Information for Selecting Features in Supervised Neural Net Learning**," *IEEE Trans. Neural Networks*, vol. 5, no. 4, pp. 537-550, July 1994.

[9] L. Yu and H. Liu, **"Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution**," *Proc. 20th Int'l Conf.Machine Leaning*, vol. 20, no. 2, pp. 856-863, 2003.

[10] K. Kira and L.A. Rendell, "**The Feature Selection Problem: Traditional Methods and a New Algorithm**," *Proc. 10th Nat'l Conf. Artificial Intelligence*, pp. 129-134, 1992.

[11] D. Koller and M. Sahami, "**Toward Optimal Feature Selection**," *Proc. Int'l Conf. Machine Learning*, pp. 284-292, 1996.

[12] I. Kononenko, "**Estimating Attributes: Analysis and Extensions of RELIEF**," *Proc. European Conf. Machine Learning*, pp. 171-182, 1994.

[13] R. Kohavi and G.H. John, "**Wrappers for Feature Subset Selection**," *Artificial Intelligence*, vol. 97, nos. 1/2, pp. 273-324, 1997.

[14] M.A. Hall, **"Correlation-Based Feature Selection for Discrete and Numeric Class Machine Learning,"** *Proc. 17th Int'l Conf. Machine Learning*, pp. 359-366, 2000.

[15] M.A. Hall, "**Correlation-Based Feature Subset Selection for Machine Learning**," PhD dissertation, Univ. of Waikato, 1999.

[16] L. Yu and H. Liu, **"Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution**," *Proc. 20th Int'l Conf. Machine Leaning*, vol. 20, no. 2, pp. 856-863, 2003.

[17] F. Fleuret, **"Fast Binary Feature Selection with Conditional Mutual Information,"** *J. Machine Learning Research*, vol. 5, pp. 1531-1555, 2004.

[18] L. Yu and H. Liu, **"Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution,"** *Proc. 20th Int'l Conf. Machine Leaning*, vol. 20, no. 2, pp. 856-863, 2003.

[19] L. Yu and H. Liu, **"Efficient Feature Selection via Analysis of Relevance and Redundancy,"** *J. Machine Learning Research*, vol. 10, no. 5, pp. 1205-1224, 2004.

[20] F. Fleuret, **"Fast Binary Feature Selection with Conditional Mutual Information,"** *J. Machine Learning Research*, vol. 5, pp. 1531-1555, 2004.

[21] 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.

[22] 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.

[23] W.W. Cohen, **"Fast effective rule induction",** *in: Proceedings of the Twelfth International Conference on Machine Learning*, Morgan Kaufmann, 1995, pp. 115–123.

[24] 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.