World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:6, No:8, 2012

A Hybrid Feature Selection by Resampling, Chi squared and Consistency Evaluation Techniques

Amir-Massoud Bidgoli, Mehdi Naseri Parsa

***Abstract*—**In this paper a combined feature selection method is proposed which takes advantages of sample domain filtering, resampling and feature subset evaluation methods to reduce dimensions of huge datasets and select reliable features. This method utilizes both feature space and sample domain to improve the process of feature selection and uses a combination of Chi squared with Consistency attribute evaluation methods to seek reliable features. This method consists of two phases. The first phase filters and resamples the sample domain and the second phase adopts a hybrid procedure to find the optimal feature space by applying Chi squared, Consistency subset evaluation methods and genetic search. Experiments on various sized datasets from UCI Repository of Machine Learning databases show that the performance of five classifiers (Naïve Bayes, Logistic, Multilayer Perceptron, Best First Decision Tree and JRIP) improves simultaneously and the classification error for these classifiers decreases considerably. The experiments also show that this method outperforms other feature selection methods.

***Keywords*—**feature selection, resampling, reliable features, Consistency Subset Evaluation.

I. Introduction

THE growth of the size of data and number of existing databases exceeds the ability of humans to analyze this data, which creates both a need and an opportunity to extract knowledge from databases [1]. High dimensional datasets usually lead to higher the misclassification rate and deteriorate the accuracy and performance of the system by curse of dimensionality. Datasets with high dimensional features create more complexity and spend longer computational time for classification [2]. Feature selection is a solution to high dimensional data. Feature selection aims to select optimal feature subsets from original features by removing irrelevant and redundant features and that increases classification accuracy, reduces complexity, computational time [3]. Feature selection also aims to improve machine learning performance [4].

For classification, feature selection is used to find an optimal subset of relevant features such that the overall accuracy of classification is increased while the data size is reduced and the comprehensibility is improved. A relevant feature is defined in [5] as one removal of which deteriorates the performance or accuracy of the classifier; an irrelevant or redundant feature is not relevant. Because irrelevant information is cached inside the totality of the features, these irrelevant features could deteriorate the performance of a classifier that uses all features.

A. M. Bidgoli is with the MIEEE Manchester University, Electrical and Computer Engineering College, Islamic Azad University, Tehran North Branch, Ph.D., am_bidgoli@iau_tub.ac.ir.

M. N. Parsa, is with the Electrical and Computer Engineering College, Islamic Azad University, Tehran North Branch, Postgraduate Student, mehdi_parsan@yahoo.com.

Feature selection methods contain two important aspects: evaluation of a candidate feature subset and search through the feature space [6]. The fundamental function of a feature selector is to extract the most useful information from the data, and reduce the dimensionality in such a way that the most significant aspects of the data are represented by the selected features [7].

Janecek [8] showed the relationship between feature selection and data classification and the impact of applying PCA on the classification process. Assareh [9] proposed a hybrid random subspace fusion model that utilizes both the feature space and sample domain to improve the diversity of the classifier ensemble. Hayward [10] showed that data preprocessing and choosing suitable features will develop the performance of classification algorithms. In another attempt Duangsoithong and Windeatt [3] presented a method for reducing dimensionality in the datasets which have huge amount of attributes and few instances. By removing irrelevant and redundant features, the precision and speed of classification are improved. Dhiraj [11] used clustering and K-means algorithm to show the efficiency of this method on huge amount of data. Xiang [12] proposed a hybrid feature selection algorithm that takes the benefit of symmetrical uncertainty and genetic algorithms. Zhou [13] presented a new approach for classification of multi class data. The algorithm performed well on two kind of cancer datasets. Azofra [14] tried to adopt a method to seek effective features of dataset by applying a fitness function to the attributes.

In fact, most of the dimensionality reduction methods concentrate on feature space and none of them check the effect of filtering the samples on the feature selection process. Moreover, most of the feature selection methods just focus on improving one specific classifier performance. Therefore, only a part of the sample space patterns are covered [9]. In this paper, we try to improve a group of classifiers performance by taking the advantages of combining sample domain filtering, resampling and feature subset evaluation methods. We also try to propose an adaptive feature selection method that is applicable for most of datasets with different sizes.

In section II, III, IV, V, VI and VII we focus on the definition of feature selection, SMOTE, Consistency subset evaluator, Chi squared, Naïve Bayes classifier, and Genetic algorithm which are used in our proposed method. In section VIII, we describe our hybrid method and explain the two phases involved in the feature selection process. In section IX, the performance of the proposed method is tested on various datasets. Conclusions are given in section X.

II. Feature Selection Structure

Feature selection consists of four basic steps (Figure 1): subset generation, subset evaluation, stopping criterion, and result validation [15].

The feature selection algorithms create a subset, evaluate it, and loop until an ending criterion is satisfied [16]. Finally the subset found is validated by the classifier algorithm on real data.

- *Subset Generation:* Subset generation is a search procedure; it generates subsets of features for evaluation. The total number of candidate subsets is 2^N, where N is the number of features in the original dataset, which makes exhaustive search through the feature space infeasible with even moderate N. Non-deterministic search like evolutionary search is often used to build the subsets [17]. It is also possible to use heuristic search methods. There are two main families of these methods: forward addition [18] (starting with an empty subset, we add features after features by local search) or backward elimination (the opposite).

- *Subset Evaluation:* each subset generated by the generation procedure needs to be evaluated by a certain evaluation criterion and compared with the previous best subset with respect to this criterion. If it is found to be better, then it replaces the previous best subset. A simple method for evaluating a subset is to consider the performance of the classifier algorithm when it runs with that subset. The method is classified as a wrapper, because in this case, the classifier algorithm is wrapped in the loop. In contrast, filter methods do not rely on the classifier algorithm, but use other criteria based on correlation notions.

- *Stopping Criteria:* Without a suitable stopping criterion, the feature selection process may run exhaustively before it stops. A feature selection process may stop under one of the following reasonable criteria: (1) a predefined number of features are selected, (2) a predefined number of iterations are reached, (3) in case addition (or deletion) of a feature fails to produce a better subset, (4) an optimal subset according to the evaluation criterion is obtained.

- *Validation:* the selected best feature subset needs to be validated by carrying out different tests on both the selected subset and the original set and comparing the results using artificial datasets or real world datasets.

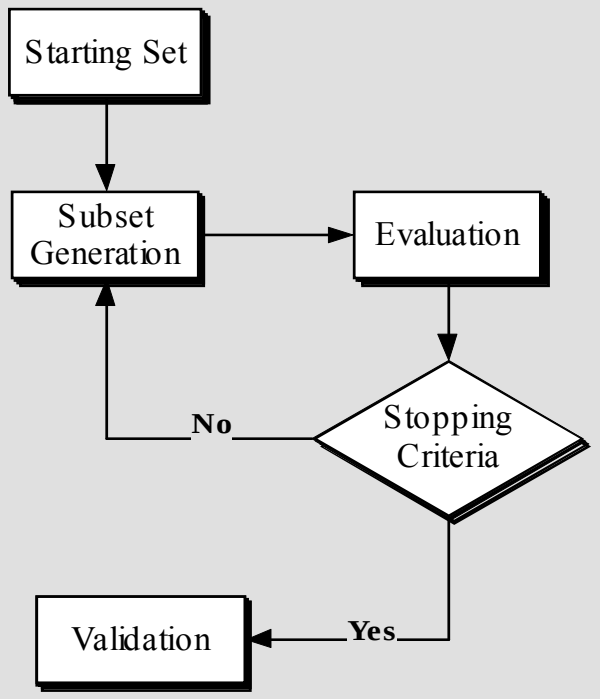

Fig. 1 Feature selection structure

III. SMOTE: Synthetic Minority Over-sampling Technique

Often real world datasets are predominantly composed of normal examples with only a small percentage of abnormal or interesting examples. It is also the case that the cost of misclassifying an abnormal example as a normal example is often much higher than the cost of the reverse error. Under sampling of the majority (normal) class has been proposed as a good means of increasing the sensitivity of a classifier to the minority class. By combination of over-sampling the minority (abnormal) class and under-sampling the majority (normal) class, the classifiers can achieve better performance than only under-sampling the majority class. SMOTE adopts an over-sampling approach in which the minority class is over-sampled by creating synthetic examples rather than by over-sampling with replacement. The synthetic examples are generated in a less application specific manner, by operating in feature space rather than sample domain. The minority class is over-sampled by taking each minority class sample and introducing synthetic examples along the line segments joining any of the k minority class nearest neighbors. Depending upon the amount of over-sampling required, neighbors from the k nearest neighbors are randomly chosen [19].

IV. Consistency Subset Evaluator (CSE)

Class consistency has been used as an evaluation metric by several approaches to attribute subset evaluation [20]. Attribute subset evaluation is done to look for combinations of attributes whose values divide the data into subsets containing a strong single class majority [21]. The search is in favor of small feature subsets with high class consistency.

This consistency subset evaluator uses the consistency metric as shown in Equation (1).

$$Consistency_s = 1 - \frac{\sum_{i=0}^{J} |D_i| - |M_i|}{N} \quad (1)$$

Where s is an attribute subset, J is the number of distinct combinations of attribute values for s, |Di| is the number of occurrences of the ith attribute value combination, |Mi| is the cardinality of the majority class for the ith attribute value combination and N is the total number of instances in the data set [21].

V. Chi-Squared

Feature Selection via chi square (X2) test is another, very commonly used method [22]. Chi-squared attribute evaluation evaluates the worth of a feature by computing the value of the chi-squared statistic with respect to the class. The initial hypothesis H0 is the assumption that the two features are unrelated, and it is tested by chi squared formula as is shown in equation (2):

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \quad (2)$$

Where O_{ij} is the observed frequency and E_{ij} is the expected (theoretical) frequency, asserted by the null hypothesis. The greater the value of $\chi 2$, the greater the evidence against the hypothesis H0 is.

VI. Naïve Bayes

The Naive Bayes algorithm is based on conditional probabilities. It uses Bayes' Theorem, a formula that calculates a probability by counting the frequency of values and combinations of values in the historical data. Bayesian classifiers find the distribution of attribute values for each class in the training data [23]. When given a new instance d, they use the distribution information to estimate, for each class c_j, the probability that instance d belongs to class cj, denoted by $p(c_j \mid d)$. The class with maximum probability becomes the predicted class for instance d. To find the probability $p(c_j \mid d)$ of instance d being in class c_j, Bayesian classifiers use Bayes theorem as shown in equation(3).

$$P(c_j|d) = \frac{P(d|c_j)P(c_j)}{P(d)} \quad (3)$$

Where $p(d \mid c_j)$ is the probability of generating instance d given class c_j, $p(c_j)$ is the probability of occurrence of class cj, and p(d) is the probability of instance d occurring. The Naive Bayes classifier is designed for use when features are independent of one another within each class, but it appears to work well in practice even when that independence assumption is not valid. Thereby it estimates $p(d|c_j)$ as shown in equation(4).

$$P(d|c_j) = P(d_1|c_j) \times P(d_2|c_j) \times \ldots \times P(d_n|c_j) \quad (4)$$

Naïve Bayes classifies data in two steps:

- Training step: Using the training samples, the method estimates the parameters of a probability distribution, assuming features are conditionally independent given the class.
- Prediction step: For any unseen test sample, the method computes the posterior probability of that sample belonging to each class. The method then classifies the test sample according the largest posterior probability.

VII. Genetic Algorithm

The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution [24]. The genetic algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population evolves toward an optimal solution. The genetic algorithm could be used to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, stochastic, or highly nonlinear.

The genetic algorithm uses three main types of rules at each step to create the next generation from the current population:

- Selection rules select the individuals, called parents that contribute to the population at the next generation.
- Crossover rules combine two parents to form children for the next generation.
- Mutation rules apply random changes to individual parents to form children.

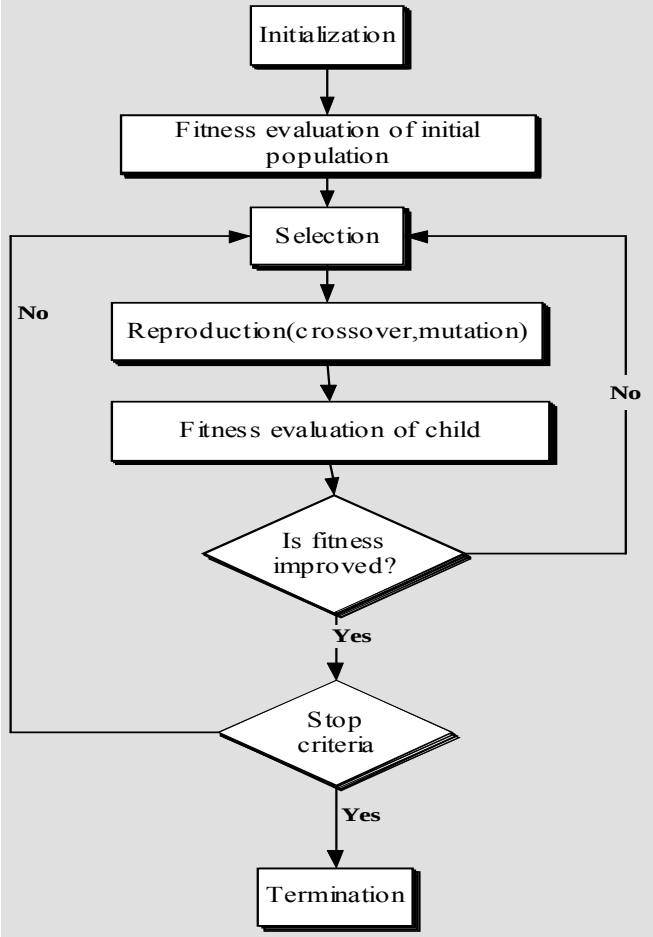

Fig. 2 Flow diagram of Genetic Algorithm steps

VIII. Proposed Method

A. Initial Phase

In the first step, we try to analyze the sample domain to find and remove some instances that could mislead the classifiers and result to misclassification. In our experiments we find out that some specific instances are misclassified by nature if taking part into the classification process. Hence, as a preparatory measure these irrelevant instances are filtered and then the Synthetic Minority Oversampling Technique (SMOTE) is applied on the training dataset. This action leads to increase the number of reliable instances and causes more diversity in the sample domain. After resampling, eliminated instances are added to the sample domain to form the final training dataset. In order to find the unreliable instances, we apply Naïve Bayes classifier on the training dataset. By elimination of misclassified instances, the accuracy of resampling is increased. Surprisingly, we find out that Naïve Bayes classifier is the best option for applying on the sample domain to eliminate unreliable instances. This classifier is not only cheap and has less computations but also more quick and reliable than the other classifiers to discover reliable instances.

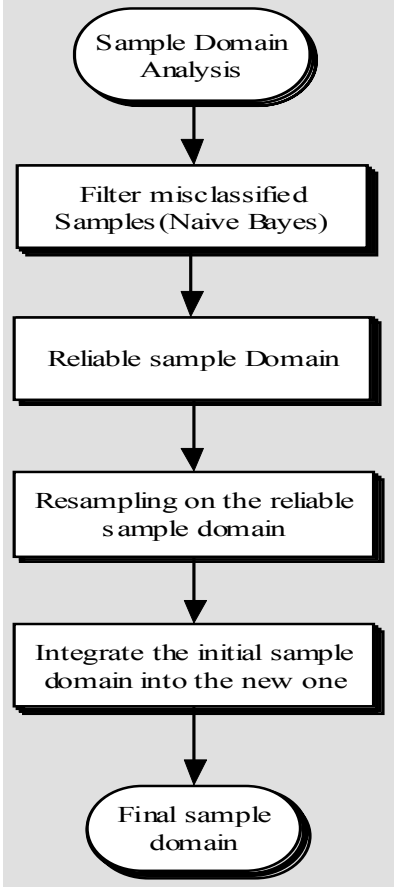

Fig. 3 Flow diagram of first step in the first phase

B. Secondary Phase

In this phase, we use a hybrid policy to find the most irrelevant features and remove them from the feature space. To find and remove irrelevant features, we use Consistency subset evaluation method and genetic search. The fitness function of GA is set to Naïve Bayes classifier and we use 10-fold-cross validation method to evaluate the goodness of the feature space. In order to set the initial population for GA, Chi squared subset evaluation method with ranker search is applied on the feature space and the features which gain higher ranks are selected as the initial population for GA search. Finally, Consistency method with GA search is run on the feature space and the features which are selected less than 4 times by GA are eliminated from the feature space. The remaining features form the optimal feature space and removing any more features from this optimal set is harmful and deteriorates the classification performance for all classifiers due to loss of necessary data.

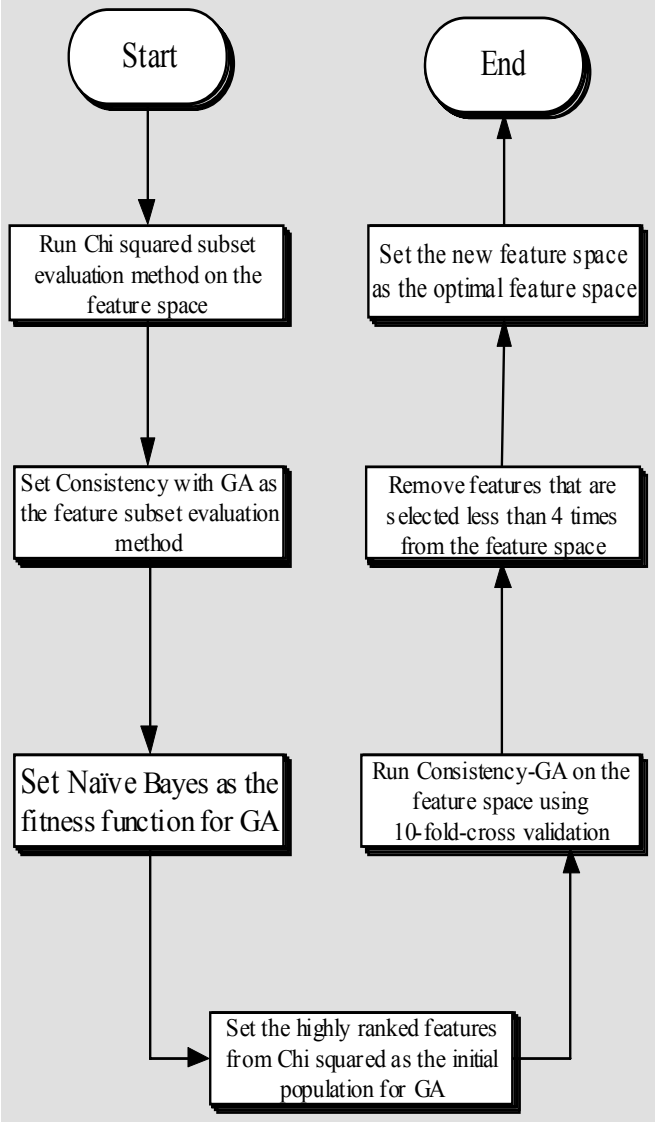

Fig. 4 Flow diagram of the second phase

IX. Empirical Study

A. Experimental Setup and Conditions

To evaluate our feature selection method, we choose 5 datasets from UCI Repository of Machine Learning databases [25] and apply 5 important classifiers before and after implementation of our feature selection method. A summary of datasets are presented in table I. The initial states of all algorithms used are shown in table II, III, IV, and V.

TABLE I
CHARACTERISTICS OF UCI DATASETS

Dataset Name	Instance	Features	Classes
Lung-Cancer	32	56	3
Ionosphere	351	34	2
Breast-Cancer	178	13	3
Sonar	208	60	2
Iris	150	4	3

TABLE II
BEST FIRST DECISION TREE INITIAL STATES

Parameter	Value
Heuristic	True
Minimum Number of Objects	2
Number Folds Pruning	5
Pruning Strategy	Post Pruning
Seed	1
Size Per	1.0
Use Error Rate	True
Use Gini	True

TABLE III
JRIP INITIAL STATES

Parameter	Value
Check Error Rate	True
Folds	3
Minimum Number	2.0
Seed	1
Optimizations	2
Use Pruning	True

TABLE IV
MLP INITIAL STATES

Parameter	Value
Learning Rate	0.3
Momentum	0.2
Random Seed	0
Training Time	500
Validation Set Size	0
Validation Threshold	20

TABLE V
GA INITIAL STATES

Parameter	Value
Crossover Probability	True
Maximum Generations	3
Mutation Probability	2.0
Population Size	1

In table II, Best First Decision Tree parameters are: Heuristic parameter that if heuristic search is used for binary split for nominal attributes this parameter is set to true. Minimum Number of Objects sets minimal number of instances at the terminal nodes and is set to 2. Number Folds Pruning is number of folds in internal cross-validation and is set to 5. Pruning Strategy sets the pruning strategy which is post pruning. The random number seed to be used is set to 1. The percentage of the training set size is determined by Size Per parameter and is set to 1.0. If error rate is used as error estimate the parameter Use Error Rate is set to true. If Use Gini parameter is true the Gini index is used for splitting criterion, otherwise the dataset data is used. In table III, JRIP parameters are: Check Error Rate which checks the error rate and is set to true. The parameter Folds determines the amount of data used for pruning and is set to 3.

Minimum Number is the minimum total weight of the instances in a rule and is set to 2.0. Seed parameter used for randomizing the data and is set to 1.

The number of optimization runs is determined by Optimizations parameter which is set to 2. Use Pruning parameter determines whether pruning is performed (true) or not (false) that is set to true in our experiments. In table IV, MLP parameters are: Learning Rate which is the amount the weights are updated and is set to 0.3. Momentum applies to the weights during updating and is set to 0.2.Random Seed parameter which is used to initialize the random number generator and is set to zero. Training Time parameter which implies the number of epochs to train through and is set to 500. The percentage size of the validation set is defined by Validation Set Size which is set to 0. This indicates no validation set will be used and instead the network will train for the specified number of epochs. Validation Threshold parameter used to terminate validation testing which is set to 20. In table V, GA parameters are set as follows: Crossover Probability is the probability that two population members will exchange genetic material and is set to 0.6. Max Generations parameter shows the number of generations to evaluate and is set to 20. Mutation Probability is the probability of mutation occurring and is set to 0.033 and the last parameter is the number of individuals (attribute sets) in the population that is set to 20.

B. Experimental Results

To evaluate the performance of the proposed method, we choose 5 datasets with different dimensions from both aspects of sample domain and feature space. The results of sample domain filtering and resampling in the first phase are presented in table VI.

TABLE VI
RESULTS OF SAMPLE DOMAIN FILTERING AND RESAMPLING IN THE FIRST PHASE ON 5 UCI DATASETS.

Name	samples	Misclassified	Resample	Final Sample Domain
Lung-Cancer	32	4	56	88
Ionosphere	351	105	492	658
Breast-Cancer	569	49	1040	1707
Sonar	208	2	352	530
Iris	150	6	288	438

TABLE VII
RESULTS OF FEATURE SUBSET EVALUATION ON 5 UCI DATASETS

Name	Feature	Chisquare-Consitency-GA	Final Feature Space
Lung-Cancer	56	21	21
Ionosphere	34	13	13
Breast-Cancer	60	10	10
Sonar	13	15	15
Iris	4	3	3

According to the results of table VI and table VII, we observe that this method leads to a higher level of dimensionality reduction. In fact, the proposed method acts on two dimensions.

At first, it tries to optimize the sample domain and then reduces the number of features by a hybrid procedure and finally comes up with the optimal sample domain and feature space.

Hence, the model which is made on the optimal training dataset results in a better accuracy and improves the classifiers performance. So, the proposed method is an effective feature selection method and works well on the wide range of different datasets.

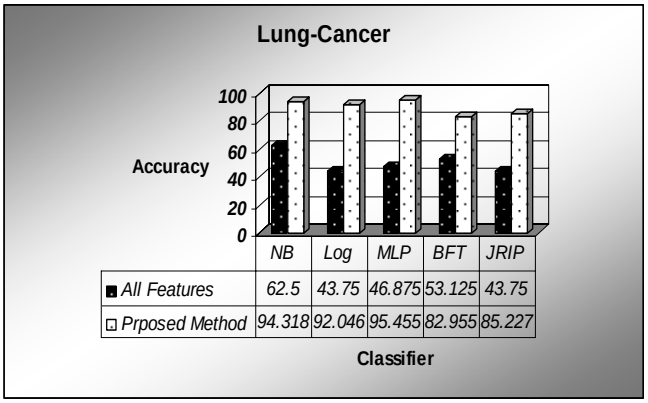

Fig. 5 The classifiers accuracy on lung-cancer dataset

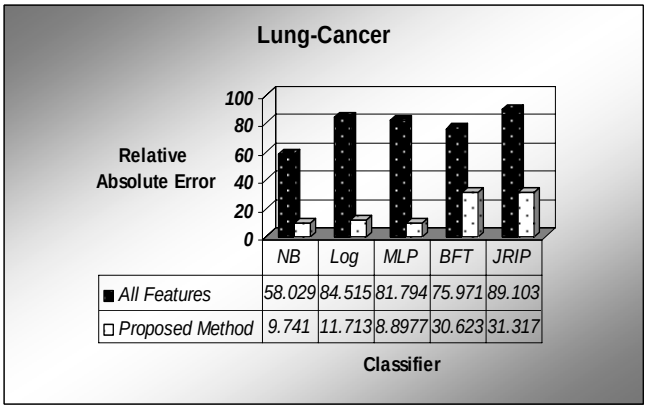

Fig. 6 The classifiers relative absolute error on lung-cancer dataset

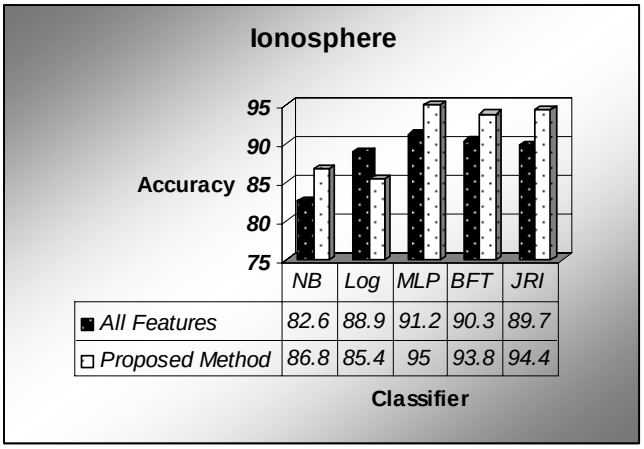

Fig. 7 The classifiers accuracy on Ionosphere dataset

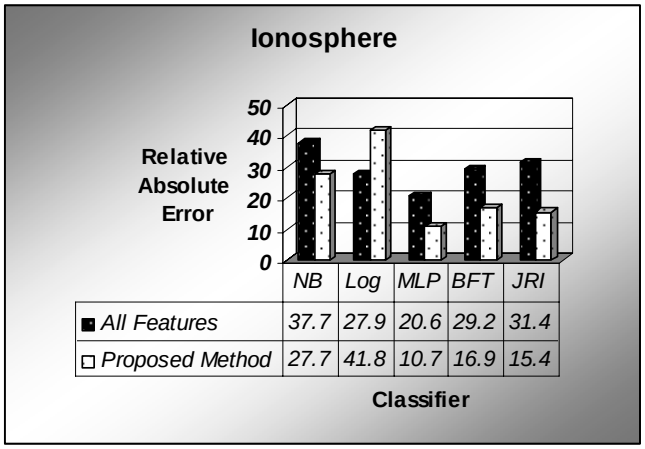

Fig. 8 The classifiers relative absolute error on Ionosphere dataset

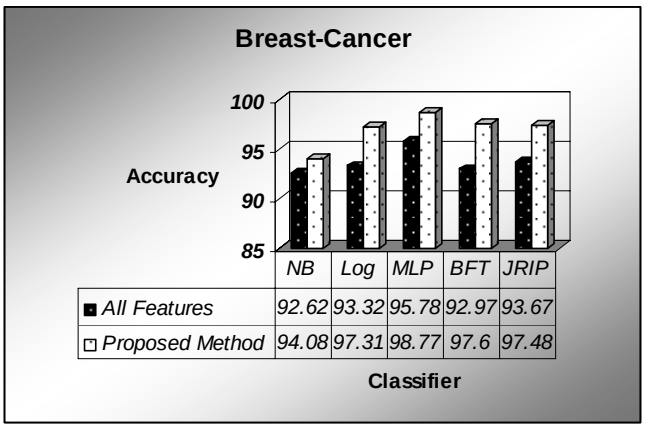

Fig. 9 The classifiers accuracy on Breast-Cancer dataset

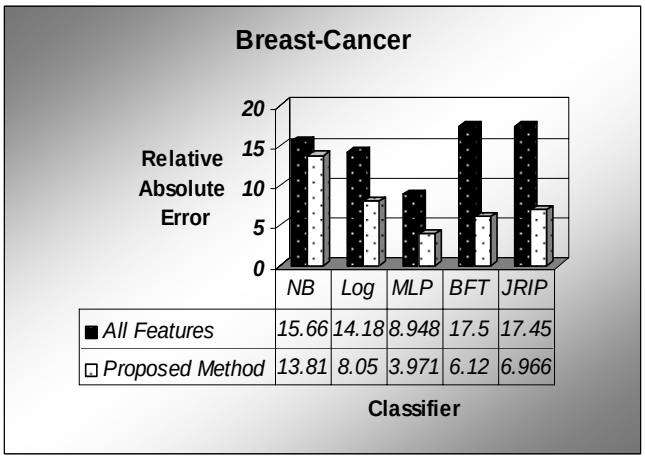

Fig. 10 The classifiers relative absolute error on Breast-Cancer dataset

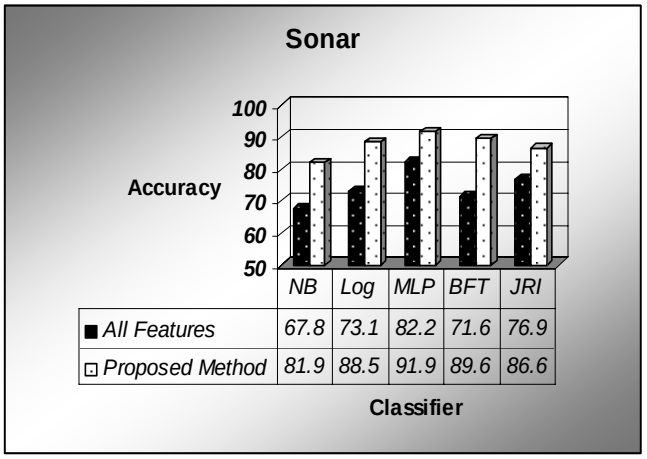

Fig. 11 The classifiers accuracy on Sonar dataset

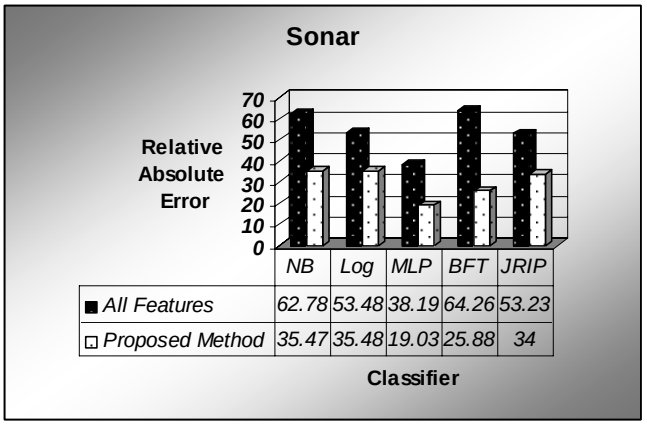

Fig. 12 The classifiers relative absolute error on Sonar dataset

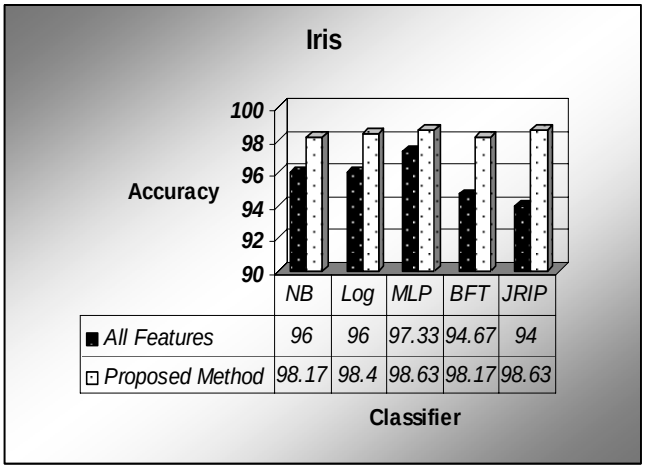

Fig. 13 The classifiers accuracy on Iris dataset

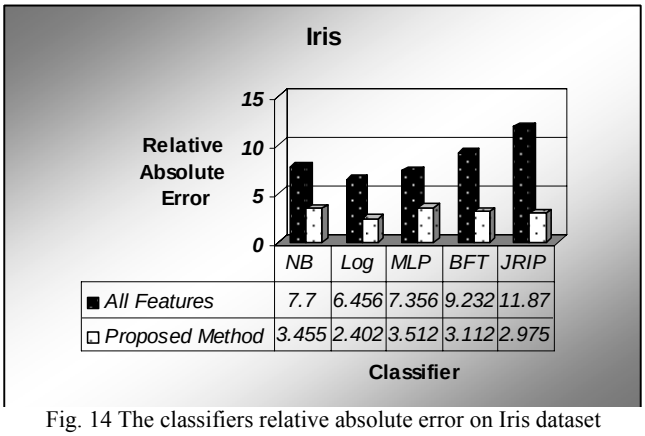

Fig. 14 The classifiers relative absolute error on Iris dataset

From figure 5 to 14, it is clear that our proposed method works well on different datasets with various sizes. Actually, the performance of five classifiers (Naïve Bayes, Logistic, Multilayer Perceptron, Best First Decision Tree and JRIP) improves considerably and simultaneously after applying our hybrid feature selection method. The accuracy of the classifiers is increased and the relative absolute error is decreased in most cases. For the datasets with larger feature space like Lung-Cancer and Sonar, the performance improvement is more noticeable.

C. Comparison and Discussion

According to the results presented in figure 15 to 24, the proposed method outperforms GA-wrapper subset evaluation and GA-classifier subset evaluation methods based on the accuracy and relative absolute error of the group of classifiers.

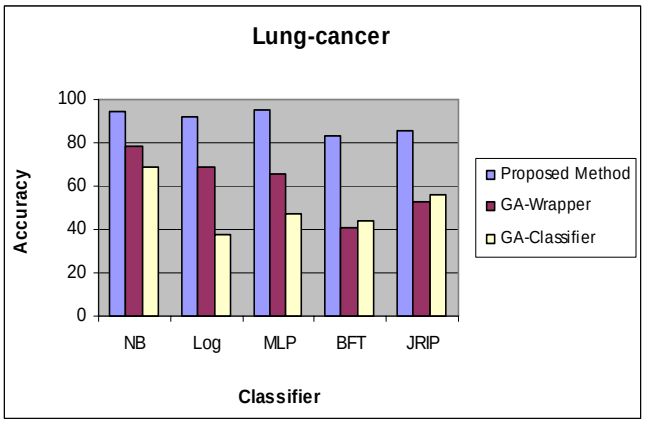

Fig. 15 Accuracy comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Lung-Cancer dataset

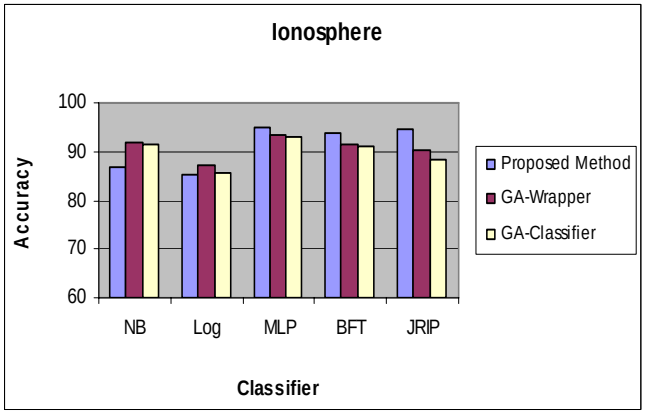

Fig. 16 Accuracy comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Ionosphere dataset

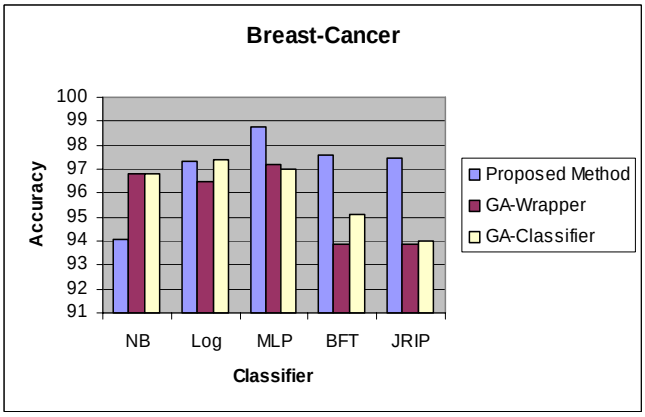

Fig. 17 Accuracy comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Breast-Cancer dataset

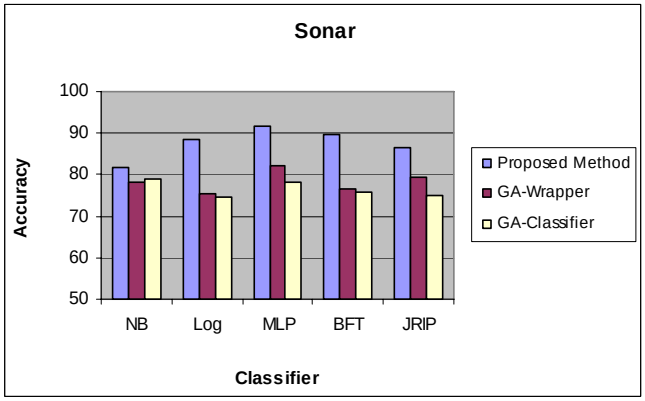

Fig. 18 Accuracy comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Sonar dataset

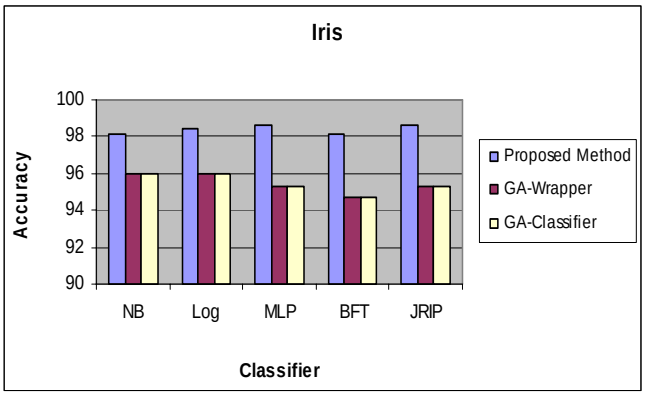

Fig. 19 Accuracy comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Iris dataset

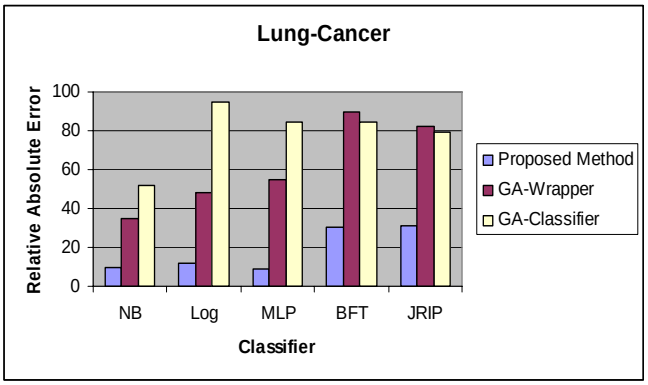

Fig. 20 Relative absolute error comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Lung-Cancer dataset

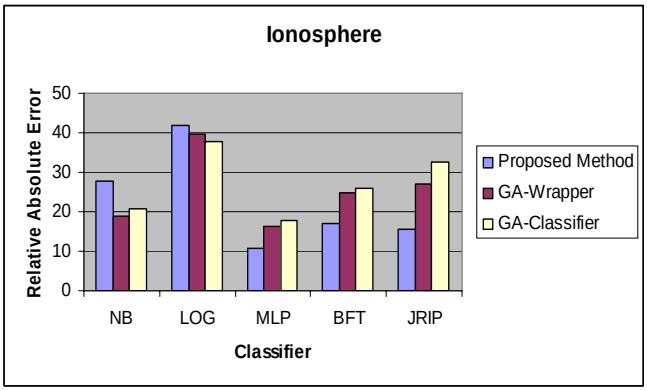

Fig. 21 Relative absolute error comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Ionosphere dataset

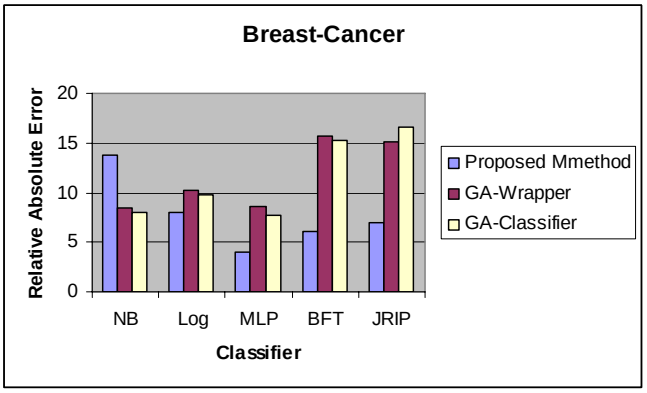

Fig. 22 Relative absolute error comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Breast-Cancer dataset

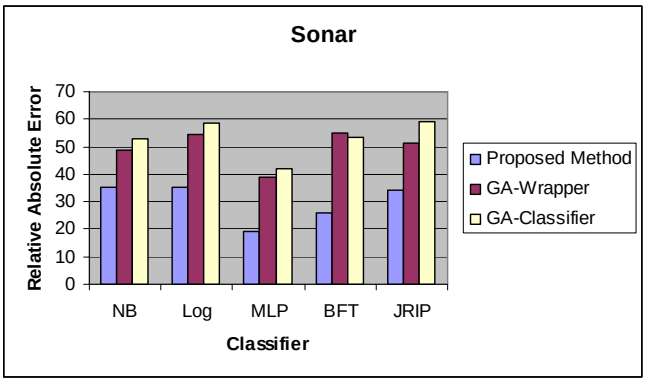

Fig. 23 Relative absolute error comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Sonar dataset

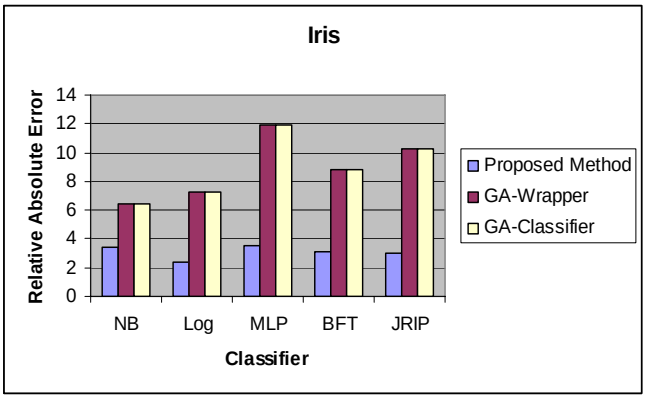

Fig. 24 Relative absolute error comparison between the proposed method, GA-Wrapper and GA-Classifiers techniques on Iris dataset

From figure 15 to 19, we observe that the proposed method achieves higher classification accuracy for the group of classifiers in comparison to GA-Wrapper and GA-Classifier methods. Moreover, the cost of our proposed method is considerably smaller than the GA-Wrapper and GA-Classifier methods, because it achieves higher level of dimensionality reduction and optimizes the sample domain before applying genetic search on the dataset. From figure 20 to 24, it is clear that for most cases, the relative absolute error for the proposed

method is smaller than the GA-Wrapper and GA-Classifier methods. This shows that our feature selection method is efficient and more reliable than the other methods.

X. Conclusion

In this paper, a hybrid feature selection method is proposed that tries to find the optimal sample domain and feature space. This method is combination of sample domain filtering, resampling and feature subset evaluation approaches. In the first phase, some irrelevant samples are removed by filtering misclassified samples and uses SMOTE technique to resample and increase the diversity of the dataset. In the second phase, the method tries to find the optimal feature space by applying Consistency subset evaluation method and genetic search on the feature space to remove irrelevant features. The experiments on small, medium and large sized datasets show that this trend leads to improve a group of classifiers performance simultaneously and decrease the classification error. This shows that combination of resampling and feature subset evaluation methods could improve the classification performance by a lower cost. The results also show that the proposed method outperforms two other feature selection methods.

References

[1] J. Han, and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, 2006.
[2] R. Bellman, *Adaptive Control Processes: A Guided Tour*, Princeton University Press, Princeton 1961.
[3] R. Duangsoithong, and T. Windeatt," Relevance and Redundancy Analysis for Ensemble Classifiers", *Springer-Verlag*, Berlin Heidelberg, 2009.
[4] H. Liu and, Z. Zhao, "Manipulating Data and Dimension Reduction Methods: Feature Selection", *Journal of Computational Complexity*, pp. 1790-1800, 2012.
[5] A. L. Blum, P. Langley, "Selection of relevant features and examples in machine learning ," *Artificial Intelligence*, vol. 97, pp. 245-271,1997.
[6] M. Dash, H. Liu, "Consistency-based search in feature selection", *Artificial Intelligence*, vol. 151, pp. 155-176, 2003.
[7] P. A. Devijver, J. Kittler, *Pattern Recognition : A Statistical Approach*, Prentice Hall, Englewood Cliffs, NJ, 1982.
[8] A. G. K. Janecek, W. N. Gansterer, M. A. Demel, and G. F. Ecker, "On the relationship between feature selection and classification accuracy," *Journal of Machine Learning and Research. JMLR: Workshop and Conference Proceedings 4*, pp. 90–105, 2008.
[9] A. Assareh, M. Moradi, and L. G. Volkert, "A hybrid random subspace classifier fusion approach for protein mass spectra classification," *Springer, LNCS*, vol. 4973, pp. 1–11, Heidelberg, 2008.
[10] J. Hayward, S. Alvarez, C. Ruiz, M. Sullivan, J. Tseng, and G. Whalen, "Knowledge discovery in clinical performance of cancer patients," *IEEE International Conference on Bioinformatics and Biomedicine*, USA, pp. 51–58, 2008.
[11] K. Dhiraj, S. K. Rath, and A. Pandey, "Gene Expression Analysis Using Clustering," *3rd international Conference On Bioinformatics and Biomedical Engineering*, 2009.
[12] B. N. Jiang, X. Q. Ding, L. T. Ma, Y. He, T. Wang, and W. W. Xie, "A Hybrid Feature Selection Algorithm: Combination of Symmetrical Uncertainty and Genetic Algorithms," *The Second International Symposium on Optimization and Systems Biology*, pp. 152–157, Lijiang, China, October 31– November 3, 2008.
[13] J. Zhou, H. Peng, and C. Y. Suen, "Data-driven decomposition for multi-class classification," *Journal of Pattern Recognition*, vol. 41, pp. 67 – 76, 2008.
[14] A. A. Azofra, J. M. Benitez, and J. L. Castro: "A feature set measure based on Relief". RASC. 2004.
[15] J. Novakovic, P. Strbac, D. Bulatovic, "Toward optimal feature selection using ranking methods and classification algorithms," *Yugoslav Journal of Operations Research*, vol. 21, pp. 119-135, 2011.
[16] H. Liu, H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*, Kluwer Academic Publishers, 1998.
[17] J. Yang, V. Honavar, "Feature subset selection using a genetic algorithm," *IEEE Intelligent Systems*, vol. 13, pp. 44-49, 1998.
[18] D. Koller, M. Sahami, "Toward optimal feature selection," *International Conference on Machine Learning*, pp. 284-292, 1996.
[19] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321-357, 2002.
[20] H. Almuallim, T. G. Dietterich, "Learning with many irrelevant features," *Proceedings of the ninth National Conference on Artificial Intelligence*, pp. 547-552, 1991.
[21] I. Kononeko, "Estimating attributes: Analysis and extensions of relief," *Proceedings of the Seventh European Conference on Machine Learning*, pp. 171-182, 1994.
[22] H. Liu, R. Setiono, "Chi2: Feature selection and discretization of numeric attributes," *IEEE 7th International Conference on Tools with Artificial Intelligence*, pp. 338-391, 1995.
[23] A. Silberschatz, H. F. Korth, and S.Sudarshan, *Database System Concepts*, McGrawHill, 2010.
[24] Haupt, Randy and S. E. Haupt, *Practical Genetic Algorithms*, John Wiley and Sons, 1998.
[25] C.J. Mertz, and P.M. Murphy, UCI Repository of machine learning databases, http://www.ics.uci.edu/~mlearn/MLRepository.html, University of California, 2011.