

Discovering Accurate and Interesting Classification Rules Using Genetic Algorithm

Janaki Gopalan*

Reda Alhadj†

Ken Barker‡

Abstract

Discovering accurate and interesting classification rules is a significant task in the post-processing stage of a data mining (DM) process. Therefore, an optimization problem exists between the accuracy and the interesting metrics for post-processing rule sets. To achieve a balance, in this paper, we propose two major post-processing tasks. In the first task, we use a genetic algorithm (GA) to find the best combination of rules that maximizes the predictive accuracy on the sample training set. Thus we obtain the maximized accuracy. In the second task, we rank the rules by assigning objective rule interestingness (RI) measures (or weights) for the rules in the rule set. Henceforth, we propose a pruning strategy using a GA to find the best combination of interesting rules with the maximized (or greater) accuracy. We tested our implementation on three data sets. The results are very encouraging; they demonstrate the applicability and effectiveness of our approach.

Keywords: *post-processing, data mining, classification rules, rule interestingness, genetic algorithms.*

1 Introduction

Data mining is generally defined as the process of extracting previously unknown knowledge from a given database. A DM process is divided into three stages namely, the pre-processing, mining, and the post-processing stages [1, 20]. The post-processing stage of the DM process involves interpretation of the discovered knowledge or some post-processing of this knowledge. An example of

a post-processing task is improving the comprehensibility of the discovered knowledge using visualization techniques [Lee *et al.* 95]. Another important form of post-processing, particularly in the context of the rule-induction paradigm, is rule-set refinement.

Data mining algorithms like “Association Rule Mining” (ARM) perform an exhaustive search to find all rules satisfying some constraints. Hence, the number of discovered rules from datasets can be very large. Based on the earlier work [2, 3], it is clear that it is difficult to identify the most effective rule at classifying new cases. Therefore, instead of identifying a single effective classification rule, identifying a set of rules with a high classification accuracy is a recent trend in data mining [4]. At the same time, in any DM process, finding the most interesting set of rules is a key motivation.

When a DM algorithm generates a set of classification rules (or an ensemble) rather than a single rule, it is often necessary to find a subset of this rule set for better predictive accuracy. For a fixed number of test iterations, the members from the derived rule set exhibit improved predictive accuracy but with the addition of extra members of the rule set, accuracy declined and often fell below the predictive accuracy of the initial rule set. Several researchers have reported similar problems [2, 3]. It has been shown that in several cases an ensemble of classifiers has a better predictive accuracy than a single classifier [5, 6]. However, there is a risk of generating too many classifiers which may over-fit the training set. Thompson [4] proposes a GA for pruning a classifier ensemble to address this problem efficiently.

The task in this paper is to find an efficient methodology to identify the best combination of *accurate interesting* classification rules. The task is divided into two sections. In the first section, the original dataset is divided into a sample set (to train) and a validation set (to test). We then, propose a strategy using a GA to find the best combination of rules that maximizes the predictive *accuracy* on the sample set (problem space). We denote the maximum accuracy as γ .

In the second part, we assign objective rule interestingness measures based on Freitas [7] to the rule set. Thompson’s work utilizes the weights calculated based on a *sub-*

*Janaki Gopalan is with the Department of Computer Science, University of Calgary, Calgary, Alberta, Canada, gopalanj@cpsc.ucalgary.ca

†Reda Alhadj is with the Department of Computer Science, University of Calgary, Calgary, Alberta, Canada, alhadj@cpsc.ucalgary.ca; he is also affiliated with the Department of Computer Science, Global University, Beirut, Lebanon

‡Ken Barker is with the Department of Computer Science, University of Calgary, Calgary, Alberta, Canada, barker@cpsc.ucalgary.ca

jective boosted hypothesis rule [4]. Therefore, the accuracy of the obtained results are biased by the accuracy with which these weights are obtained. Moreover, these weights are based on one metric which is the *classification accuracy* of the classifier. In this paper, we propose a pruning strategy by extending the idea proposed by Thompson. In our strategy, we use a GA with *objective* rule interestingness measures (based on Freitas [7]) to find the most *interesting* subset with the performance accuracy of *at least* γ on the sample set (problem space). These measures are based on several objective metrics (including the accuracy metric) to derive interesting as well as accurate rules. Therefore, the resulting rule set from the solution of our GA method is the best combination of *accurate interesting* classification rules. These rules are then tested for their accuracy on the unknown validation set (the solution space).

The rest of the paper is organized as follows. Section 2 describes the related work in rule-set refinement for classification rules. Section 3 discusses the implementation using a GA for this problem. In Section 4, we give the experimental results using the GA method. Section 5 is conclusions and future work.

2 Related Work

In this section, we discuss the related work in rule-set refinement for *classification rules*. They are: 1) The rule interestingness (RI) principles proposed for classification rules; and 2) finding the best set (or subset) of rules from the discovered rule-set. The task of assigning a RI measure is discussed first, followed by the discussion of deriving the best combination of accurate rules.

Methods for the selection of interesting classification rules can be divided into subjective and objective methods. Subjective methods are user-driven and domain-dependent. By contrast, objective methods are data-driven and domain-independent. A comprehensive review about subjective aspects of RI is available [10]. Piatetsky-Shapiro [11], Major and Mangano [12], and Kamber and Shinghal [13] propose objective principles for RI to include the rule quality factors of coverage, completeness, and a confidence factor. Freitas (1999) [7] extended the objective RI principles [11, 12, 13] to include additional factors such as the disjunct size, imbalance of class distributions, attribute interestingness, misclassification costs, and the asymmetric nature of classification rules.

We consider the next problem of pruning rule sets. Prodromidis *et al.* [14] present methods for pruning classifiers in a distributed meta-learning system. A pre-training pruning is used to select a subset of classifiers from an ensemble which are then combined by a meta-learned combiner. Margineantu and Deitterich [2] use a backfitting algorithm for pruning classifier sets. This involves choosing an ad-

ditional classifier to add to a set of classifiers by a greedy search and then checking that each of the other classifiers in the set cannot be replaced by another to produce a better ensemble. Thompson [4] proposes a GA to prune a classifier ensemble to find the right combination of classifiers without over-fitting the training set. The proposed GA uses a real-valued encoding. Each chromosome has n real-valued genes, where n is the number of classifiers in the ensemble. Each gene represents the voting weight of its corresponding classifier calculated using a boosted hypothesis (WVBH) rule. The fitness function consists of measuring the predictive accuracy of the classifier ensemble with the weights proposed by the chromosomes on a hold-out set (different from the training set). Two major conclusions are drawn: 1) In a majority of the experiments that were performed with the classifier sets, it was found that a subset of classifiers from the original classifier ensemble had a better classification accuracy. 2) The GA method was very efficient in finding the right set of pruned classifiers. Moreover, the pruned classifier sets from the GA method have a better classification accuracy over the pruned classifiers sets from earlier work [4].

In this paper, we propose a pruning strategy using a GA to find the best set of *interesting* and *accurate* rules by extending the idea proposed by Thompson. In the rest of the paper, we first discuss our proposed approach that employs GA to address this problem. Finally, we present the experimental validations along with future work.

3 Post-processing Rule Sets Using Genetic Algorithms

GAs were introduced by Holland [8], as a general model of an adaptive processes, but subsequently widely exploited as optimizers [8]. Basically, a GA can be used for solving problems for which it is possible to construct an objective function (also known as fitness function) to estimate how a given representative (solution) fits the considered environment (problem). In general, the main motivation for using GAs in any data mining process is that they perform a global search and cope better with interaction than the greedy rule induction algorithms often used in data mining.

Genetic algorithms can be used in the post-processing stage of the DM process. Very little work has been reported in the literature in this area. As reviewed in earlier sections, Thompson [4] proposes a GA to prune a classifier ensemble efficiently.

We implemented GAKPER, a GA based Knowledge Discovery algorithm for deriving Efficient Rules to achieve our goal. In our implementation, the original dataset is divided into a sample set (to train) and a validation set (to test). This task is divided into two parts. In the first part, a binary encoded GA is used to find the most accurate sub-

set of rules with the best classification accuracy γ on the sample set (problem space). In the second part, a binary encoded GA is used to find the most *interesting* subset with accuracy of at least γ on the sample set. Finally, the derived accurate interesting rules are tested on the unknown validation set (solution space) for their accuracy. Each of these parts are described in the subsections below.

3.1 GAKPER ALGORITHM - PART I

A binary encoded GA is used to search for the best combination of accurate rules. Each chromosome in the population is a subset of the classification rules. The length of the chromosome is the number of rules in the rule set; in that, each gene represents the corresponding classification rule. For example, if there are 10 rules in the original rule set, then "1001100111" is a possible chromosome, where the first, fourth, fifth, eighth, ninth, and tenth rules from the rule set are chosen to represent the set of accurate rules. A solution in the phenotype space is only represented by a single chromosome and all possible chromosomes are valid. Hence a 1-1 mapping exists between the genotype and phenotype spaces.

The fitness function first measures the predictive accuracy of the rules (represented by the chromosome) on the entire *sample set*. This is achieved as follows. The true class values for all instances in the sample set is stored prior to running the GA. The classes predicted by the rule set representing the chromosome are known. Therefore, to classify the test instances from the sample set, the fitness function takes a vote of the rules from the rule set. Thus, to calculate the class of a single test instance, the class predicted by the rules representing the chromosome (based on majority vote) is matched with the original stored class value of the test instance. This process is repeated for all the test instances in the sample set. Finally, the best combination of rules that maximizes the predictive *accuracy* on the sample set (i.e., the number of correctly predicted test instances) is obtained. The maximum accuracy is denoted as γ . If C_i represents the number of correctly predicted instances by the i^{th} chromosome, the fitness value of the i^{th} chromosome is defined as:

$$FitnessFunction = C_i \tag{1}$$

All these aspects are precisely encoded and implemented into the GA and all the chromosomes (potential solutions) should be awarded or punished according to the criteria stated above during the process of evolution. The outcome of several evolutions modeled by this GA generates the right set of accurate rules.

The GAKPER algorithm (Part I) is presented below:

ALGORITHM: GAKPER-I

Input: Classification rule set Output: Set of accurate rules

Method: 1) Search for the right set of accurate

- rules using the GA Method as follows:
- 1a) Randomly create an initial population of potential accurate rules.
 - 1b) Iteratively perform the following sub steps on the population until the termination criterion in the GA is satisfied:
 - a) FITNESS :Evaluate fitness $f(x)$ of each chromosome in the population
 - b) NEW POPULATION
 - b1) SELECTION :Based on $f(x)$
 - b2) RECOMBINATION:2-point Cross-over of chromosomes
 - b3) MUTATION :Mutate chromosomes
 - b4) ACCEPTATION:Reject or accept new one
 - c) REPLACE :Replace old with new population in new generation
 - d) TEST :Test problem criterium (Number of generations is >100)
 - 2) After the termination criterion is satisfied, the best chromosome in the population produced during the run is designated as the required combination of the accurate rules.

3.2 GAKPER ALGORITHM - PART II

A binary encoded GA is used to search for the best combination of interesting accurate rules. The representation for the chromosome is the same as described in Part 1. Each chromosome in the population is a subset of the classification rules. The length of the chromosome is the number of rules in the rule set, in that, each gene represents the corresponding classification rule.

The maximum predictive accuracy γ is known from the result of the GAKPER algorithm - Part I. The RI measure (or weight) proposed by Freitas is assigned to all the rules. The fitness function optimizes the weights to find the best combination of interesting rules with a classification accuracy of at least γ . It is important to note that accuracy of the rules when tested on the test instances (from the sample set) are based on the *weighted* majority vote. Thus, a rule with a higher RI measure ranks higher in classification over a rule with a lower RI measure. The weights of the k rules representing a chromosome are denoted as $W_1, W_2, W_3 \dots W_k$. If the accuracy of the k rules is greater than (or equal) to γ , then the fitness value of the chromosome is the sum of the k weights, otherwise the chromosome is given a default fitness value. The intuitive idea of choosing this default value is: 1) the value has to be greater than zero to continue the GA runs in subsequent generations, and 2) since the fitness criteria is not satisfied by the respective chromosome, an arbitrary value is chosen to punish the least fit chromosome. The fitness value of the chromosome is defined as:

$$FitnessFunction = \begin{cases} W_1 + W_2 \dots W_k & \text{If Classification Accuracy } \geq \gamma \\ (W_1 + W_2 \dots W_k) / 2 & \text{Otherwise} \end{cases} \tag{2}$$

All these aspects are precisely encoded and implemented into the GA and all the chromosomes (potential solutions) should be awarded or punished according to the criteria stated above during the process of evolution. The outcome of several evolutions modeled by this GA generates the right set of accurate interesting rules.

The GAKPER algorithm (Part II) is presented below:

```

ALGORITHM: GAKPER-II
Input: Classification rule set
Output: Set of accurate interesting rules Method:
(1) Assign weights for the rules in the input rule
    set.
(2) Search for the right set of accurate
    interesting rules using the GA Method as
    follows:
    2a) Randomly create an initial population of
        potential accurate interesting rules.
    2b) Iteratively perform the following sub steps
        on the population until the termination
        criterion in the GA is satisfied:
        a) FITNESS : Evaluate fitness  $f(x)$  of each
            chromosome in the population
        b) NEW POPULATION
            b1) SELECTION :Based on  $f(x)$ 
            b2) RECOMBINATION:2-point Cross-over of
                chromosomes
            b3) MUTATION :Mutate chromosomes
            b4) ACCEPTATION :Reject or accept new one
        c) REPLACE :Replace old with new population:
            the new generation
        d) TEST :Test problem criterium
            (Number of generations is > 100)
(3) After the termination criterion is satisfied,
    the best chromosome in the population produced
    during the run is designated as the right
    combination of the accurate interesting rules.
  
```

The approach is implemented as a standard GA written in C, similar to Grefenstette's GENESIS program; Baker's SUS selection algorithm [21] is employed; 2-point crossover is maintained at 60% and mutation is very low; and selection is based on proportional fitness. It is important to note that this approach optimizes the predictive accuracy and the interesting measures of the rule set on the entire sample set, which is the problem space.

4 Results and Discussions

All the tests have been conducted using a single Processor, Intel(R) Xeon(TM) UNIX machine with CPU power of 2.80GHz and cache size of 512 KB.

The GAKPER algorithm implementation is tested on five datasets. The data-splitting, that is, dividing the dataset into the *sample set* and *validation set* is performed using random sampling technique. The classification rules are obtained using the *sample set*. Recall that, to prune the rule set to derive the set of interesting accurate rules, the following tests are performed. The GAKPER (part I) is used

to derive the subset of rules that maximizes the classification accuracy on the *sample set*. To derive the most interesting subset, a RI measure or weight (based on Freitas) is assigned to the rules. GAKPER (part II) is used again, this time, to maximize the interestingness measure of the rules whose classification accuracy on the *sample set* is at least γ . The accuracy of the derived interesting accurate rules (on the validation set) from this approach is compared with: 1) pruning the rule set using GA without assigning initial weights for the rules, and 2) without any pruning methods, using the entire rule set. The results for the five datasets are presented below.

Data Set 1: Breast Cancer Data Set

This is a real data set obtained from Tom Baker Cancer Center, Calgary, Alberta, Canada. The original dataset consists of 221 records and 16 attributes. Each record represents follow-up data for one breast cancer case. Breast cancer "recurred" in some of these patients after the initial occurrence. Hence, each patient is classified as "recurrent" or "non-recurrent", depending on his or her status. With respect to classification of the dataset, there are 2 classes: 1) Recurrent Patients, and 2) Non Recurrent Patients.

The original dataset is divided into a *sample set* and a *validation set* using the random sampling technique. The sample set has 121 instances whereas the validation set has 100 instances. The GA based approaches, *i.e.*, GAKPER-I and GAKPER-II, is used (on the sample set) to find the best combination of accurate interesting rules. These rules are tested on the unknown validation set for accuracy. The GA parameters are: *Population Size* = 100, *2-Point CrossOver Fraction* = 0.6, *Reproduction Fraction* = 0.1, *Mutation Fraction* = 0.1%, and the *Selection Method* is *Proportional*. 30 experiments were performed with the GA approaches and every GA experiment was run for 100 generations in both approaches. The post-processing results on the sample set and the validation sets are presented in Table 1 and Table 2 respectively.

Table 1. Post-processing Results using Different Approaches on the Sample Set for Dataset1

	With Weights	Without Weights	Entire Rule Set
Correctly Predicted (in %)	80	80	79
Incorrectly Predicted (in %)	10	10	11
Unknown (in %)	10	10	10

In Table 1, the accuracy of the rule sets, *while* pruning, using the different approaches on the sample set (the prob-

Table 2. Post-processing Results using Different Approaches on the Validation Set for Dataset1

	With Weights	Without Weights	Entire Rule Set
Correctly Predicted (in %)	78	77	75
Incorrectly Predicted (in %)	7	8	10
Unknown (in %)	15	15	15

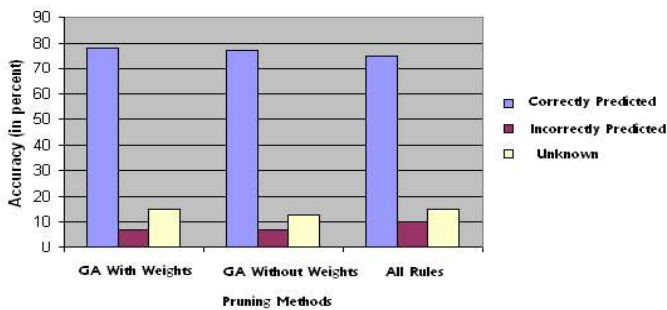


Figure 1. Plot of Post-processing Results on the Validation Set using Different Approaches for Dataset1

lem space) for Dataset 1 is presented. Table 2 presents the accuracy results of the *discovered* rule sets (using the different approaches) when tested on the validation set (the solution space) for Dataset 1. It is important to observe that the accuracy of the result on the unknown validation set using our approach (as presented in the first column in Table 2) is higher than the traditional approaches (as presented in the second and third columns in Table 2). This is also depicted graphically in Fig 1.

Data Set 2: Benchmark IRIS dataset

This is a dataset obtained from the UCI repository [19]. It is perhaps the best known database in the pattern recognition literature. It contains 150 instances with 3 classes namely, Iris-setosa, Iris-versicolor and Iris-virginica. Each class refers to a type of Iris plant. One class is linearly separable from the other 2; the latter two are not linearly separable from each other. The predicted attribute is the class of Iris plant.

The original dataset is divided into a *sample set* and a *validation set* using the random sampling technique. The sample set has 100 instances whereas the validation set has 50 instances. The GA based approaches, *i.e.*, GAKPER-I

and GAKPER-II, is used (on the sample set) to find the best combination of accurate interesting rules. These rules are tested on the unknown validation set for accuracy. Finally, the same GA parameters enumerated earlier for Dataset 1 is used here. The post-processing results are presented in Table 3 and Table 4.

Table 3. Post-processing Results using Different Approaches on the Sample Set for Dataset2

	With Weights	Without Weights	Entire Rule Set
Correctly Predicted (in %)	94	91	94
Incorrectly Predicted (in %)	0	0	0
Unknown (in %)	6	9	6

Table 4. Post-processing Results using Different Approaches on the Validation Set for Dataset2

	With Weights	Without Weights	Entire Rule Set
Correctly Predicted (in %)	91	91	91
Incorrectly Predicted (in %)	3	3	3
Unknown (in %)	6	6	6

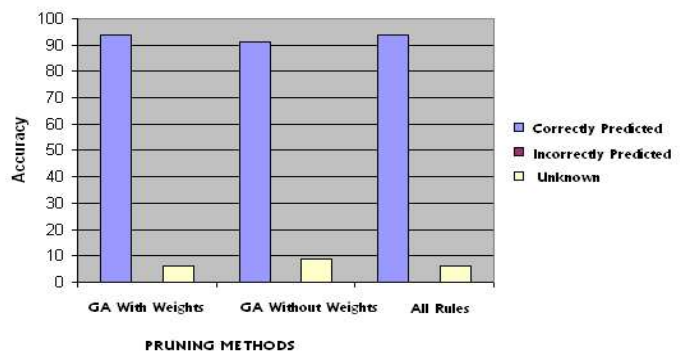


Figure 2. Plot of Post-processing Results on the Validation Set using Different Approaches for Dataset2

In Table 3, the accuracy of the rule sets, *while* pruning,

using the different approaches on the sample set (the problem space) for Dataset 2 is presented. Table 4 presents the accuracy results of the *discovered* rule sets (using the different approaches) when tested on the validation set (the solution space) for Dataset 2. For this dataset, we found that the accuracy of the result on the unknown validation set using different approaches is the same (as presented in the first, second and third columns in Table 4). This is also depicted graphically in Fig 2.

Data Set 3: US-CENSUS-DATASET

This data is the USCensus1990raw data set obtained from the UCI repository [19]. The data was collected as part of the 1990 census. The original dataset consisted of 68 categorical attributes and 100000 instances. The dataset used here consists of 5000 instances and 10 attributes derived from the original USCensus1990rawdaset. It contains 3 classes namely, iClass=0, iClass=5, and iClass=1. Each class refers to the native country of the candidate under consideration.

The original dataset is divided into a *sample set* and a *validation set* using the random sampling technique. The sample set has 3335 instances while the validation set has 1665 instances. The GA based approaches, *i.e.*, GAKPER-I and GAKPER-II, is used (on the sample set) to find the best combination of accurate interesting rules. These rules are tested on the unknown validation set for accuracy. Finally, the same GA parameters enumerated earlier for Dataset 1 is used here. The post-processing results are presented in Table 5 and Table 6.

In Table 5, the accuracy of the rule sets, *while* pruning, using the different approaches on the sample set (the problem space) for Dataset 3 is presented. Table 6 presents the accuracy results of the *discovered* rule sets (using the different approaches) when tested on the validation set (the solution space) for Dataset 3. It is very important to observe that, for this dataset, the accuracy of the result on the unknown validation set using our approach (as presented in the first column in Table 6) is much greater than the accuracy of the results using the traditional approaches (as presented in second and third columns in Table 6). This is also depicted graphically in Fig 3.

Table 5. Post-processing Results using Different Approaches on the Sample the US census data

	With Weights	Without Weights	Entire Rule Set
Correctly Predicted (in %)	70	66	50
Incorrectly Predicted (in %)	30	34	50
Unknown (in %)	0	0	0

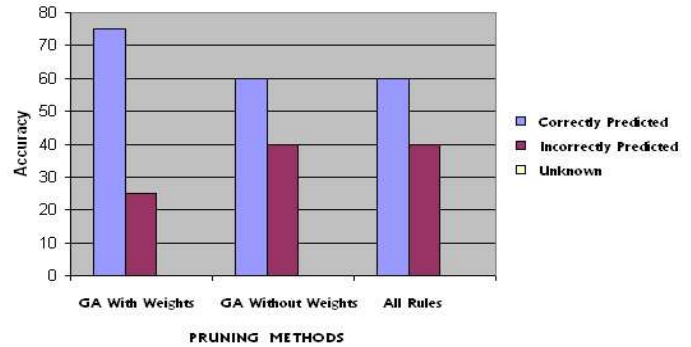


Figure 3. Plot of Post-processing Results on the Validation Set using Different Approaches for the US census data

Table 6. Post-processing Results using Different Approaches on the Validation Set the US census data

	With Weights	Without Weights	Entire Rule Set
Correctly Predicted (in %)	75	60	60
Incorrectly Predicted (in %)	25	40	40
Unknown (in %)	0	0	0

From the results, it can be observed that: 1) in majority of the tests performed, a subset of rules pruned from the original set has a better performance accuracy; and 2) it is possible to derive the most *interesting* subset with a higher classification accuracy as compared to the original rule set. Therefore, the GA with its inherent robust search strategies is most suitable for the post-processing problem considered in this paper.

5 Conclusions and Future Work

In this paper, we propose and implement a GA based methodology to derive interesting and accurate classification rules from a dataset.

The fundamental goal of any data mining model is to derive *interesting* rules. At the same time, accuracy is a key issue. Therefore, in the post-processing component, the problem of deriving interesting accurate rules is addressed. The earlier works in this area addressed the following two problems independently: 1) the problem of finding the accurate set (or subset) of rules using pruning strategies to the

original rule set [4]; and 2) the problem of assigning subjective or objective RI measures to the rules to determine their *interestingness* [7, 11, 12, 13]. In our work, a new methodology is proposed by first assigning an objective RI measure based on Freitas [7] to the rules and using pruning strategies with a GA to search for the right set of *interesting accurate* rules.

An alternative approach worth investigating is to find the interesting accurate rules using multi-objective genetic algorithms. The goal is to optimize two parameters, namely, the *interesting* and the *accuracy* metrics for the classification rules simultaneously.

References

- [1] Dorain Pyle, "Data Preparation For Data Mining", Morgan Kaufmann, 1999.
- [2] D. D. Margineantu and T. G. Dietterich, "Pruning Adaptive Boosting". *Proceedings of the 14th International Conference on Machine Learning*, San Francisco, CA, pp.211-218, 1997.
- [3] J. R. Quinlan, "Boosting First Order Learning". *Proceedings of the 14th International Conference on Machine Learning*, 1997.
- [4] S. Thompson, "Genetic Algorithms as Postprocessors for Data Mining". *Data Mining with Evolutionary Algorithms: Research Directions - Papers from the AAAI Workshop*, pp 18-22, 1999.
- [5] R. E. Schapire and Y. Freund and P. Bartlett and W. S. Lee, "Boosting the Margin: a New Explanation for the Effectiveness of Voting Methods". *Machine Learning: Proceedings of the 14th International Conference*, pp.322-330, 1997.
- [6] P. Domingos, "Knowledge Acquisition from Examples via Multiple Models". *Machine Learning: Proceedings of the 14th International Conference*, pp 98-106, 1997.
- [7] A.A. Freitas, "On Rule Interestingness Measures". *Advances in Evolutionary Computation, Knowledge-Based Systems*, 12, 1999.
- [8] D.E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning". *Addison Wesley, Longman Publishing Co., Inc.*, Boston, MA, 1989.
- [9] A.A. Freitas, "A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery". *Advances in Evolutionary Computation*, Springer-Verlag, 2001.
- [10] B. Liu, W. Hsu, and Yiming Ma, "Integrating Classification and Association Rule Mining". *Proc. of KDD*, pp.80-86, 1998.
- [11] G. Piatetsky-Shapiro, "Discovery, Analysis, and Presentation of Strong Rules". *Knowledge Discovery in Databases*, pp.229-248, 1991.
- [12] J. A. Major and J. J. Mangano, "Selecting Among Rules Induced from a Hurricane Database". *Proceedings of AAI-93, Workshop on Knowledge Discovery in Databases*, pp.28-44, 1993.
- [13] M. Kamber and R. Shinghal, "Evaluating the Interestingness of Characteristic Rules". *Proceedings of the 2nd International Conference on KDD*, pp.28-44, 1993.
- [14] A. L. Prodromidis and S. Stolfo, "Pruning Classifiers in a Distributed Meta-Learning System". *Proceedings of the KDD*, pp.151-160, 1998.
- [15] C. M. Fonseca and P. J. Fleming, "Genetic Algorithms for Multi-Objective Optimization: Formulation, Discussion and Generalization". *Proceedings of the Fifth International Conference on Genetic Algorithms*, pp.93-100, 1985.
- [16] R.J. Bayardo, "Brute-Force Mining of High-Confidence Classification Rules". *Proc. of KDD*, pp.123-126, 1997. C. M. Fonseca and P. J. Fleming
- [17] Han, "CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules". *Proc. of IEEE-ICDM*, pp.369-376, 2001.
- [18] R. Agrawal, and R. Srikant, "Fast algorithms for mining association rules," *Proc. of VLDB*, Santiago, Chile, September 1994.
- [19] C.L. Blake, and C.J. Merz, "UCI Repository of machine learning databases". University of California, Department of Information and Computer Science, Irvine, CA, 1998.
- [20] I.H. Witten and E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations". *Morgan Kaufmann*, October 1999.
- [21] J.E. Baker, Reducing Bias & Inefficiency in the Selection Algorithm. *Proc. of the International Conference on Genetic Algorithms*, pp.14-21, 1987.