In depth Coverage and Analysis of Information Fusion Technique (with Enhanced Algorithm for Feature Selection with Multiple Classifier System) for Data Mining

Amit Ganatra, Y P Kosta

Abstract — The main objective of Information Fusion techniques in Data Mining is to show that final information generated to be of superior quality and more meaningful, than the information available from the actual (primary) sources. Fusion, by definition, requires a qualitative difference between the final output and the output of the original sources. Information fusion is the process of acquisition, filtering, correlation and integration of relevant information from various sources into one representational format that is appropriate for deriving decisions regarding the interpretation of the information. In theory, the fusion of redundant information from different sources can reduce redundancy and overall uncertainty and thus increase the accuracy of the system. The fusion can be performed on three levels: raw data level, feature level, or decision level. This paper presents a novel idea of a multiple (ensemble) classification (classifier) system with feature selection where Neural Networks (Multilayer Feed-forward Networks with Back Propagation learning) are boosted for scalable (High Dimensional) datasets. The method uses Genetic Algorithms for Feature Selection with various Evaluation Techniques (Evaluators) like subset evaluation, consistency subset evaluation and wrapper subset approaches to enhance the performance of the feature selection and overall system.

Keywords— Classification, Multiple Classifier Pre-processing, Training, Testing, Feature Selection, AttributeSelectedClassifier (ASC)

I. INTRODUCTION

The system proposed here uses Genetic Algorithms with various Evaluation Techniques (Evaluators) to filter the whole feature set of the data. It then builds as ensemble of the classifier – Neural Networks (Multilayer Feed-forward Networks with Back Propagation learning) for classification.

Many real-world applications are born with high dimensionality, i.e., with a large amount of input features. This is a paradigm that can help us to deal with such kind of data and applications, i.e., feature selection. Feature selection is another factor that impacts classification accuracy. Some applications, however, have a plenty of attributes. In most of the applications it can be useful to pre-select a subset of the attributes that will be used to construct the model.

We then see the classifier that is constructed from the reduced set of attributes. In this case it is identical to the original classifier, but attribute selection will usually lead to a different classifier.

Manuscript received October 09, 2011.

Amit Ganatra, Department of Computer Engineering, Charotar Institute of Technology (Faculty of Technology and Engineering), Charotar UniversityofTechnologyChanga,Anand,Gujarat,India,388421
amitganatra.ce@charusat.ac.in

Y P Kosta Marwadi Education Foundation's Group of Institutions, Rajkot-360 003, Gujarat, India,ypkosta@yahoo.com

Many factors affect the success of Machine Learning (ML) on a given task. The representation and quality of the instance data is first and foremost. If there is much irrelevant and redundant information present or noisy and unreliable data, then knowledge discovery during the training phase is more difficult. It is well known that data preparation step take considerable amount of processing time in ML problems. System expects the error rate on unseen test cases to increase.

The objectives of feature selection are:

1. To avoid over fitting and improve model performance, i.e. prediction performance in the case of supervised classification
2. To provide faster and more cost-effective models
3. To gain a deeper insight into the underlying processes that generated the data.

Practical Advantages:

1. A simpler model with relevant attributes is easier to understand and interpret and consumes less resources
2. Less information to be collected for prediction
3. The deployment will be facilitated easily
4. A simpler model is more robust in generalization i.e. when we want to classify an unseen instance from the population.

Search strategy. The search strategy is very important in the current approach.

The more sophisticated method, like Genetic Algorithms is used here.

But there are two drawbacks to explore a very high number of solutions with Genetic Algorithms: (i) we can reach to very specific solution to the used dataset which is not efficient into the population, it is the overfitting problem; (ii)the search becomes computationally intractable when we deal with a very large dataset with a large number of descriptors.

The search method (like Genetic Algorithms) along with various evaluators is used during the attribute selection phase before the classifier is invoked to come out of the above drawbacks.

Thus, Genetic Algorithms as a search technique along with various evaluation techniques like subset evaluations, consistency subset evaluation etc. are actually a very good concession in the most of the situations because of its robustness and versatile nature to work with the overall model. The search method and the evaluator are used during the attribute selection phase before the multiple classifiers are invoked.

II. NEED FOR RESEARCH

The attraction that this topic exerts on machine learning researchers is based on the premise that ensembles are often much more accurate than the individual classifiers that make them up. Most of the research on classifier ensembles is concerned with generating ensembles by using a single learning algorithm, such as decision tree learning or neural network training. Different classifiers are generated by manipulating the training set (as done in boosting or bagging), manipulating the input features, manipulating the output targets or injecting randomness in the learning algorithm. In present stacking method we are not manipulating the training set or manipulating the feature set. Many researches have been done on finding good Meta learner at Meta level. Better accuracy can also be possible by manipulating the training set as boosting and bagging achieves. By applying the boosting method, training set will be manipulated and diversity will be increased among the base classifiers so accuracy will be improved. For the large data set training time is very large, so with use of features.

AdaBoost

Many real-world applications are born with high dimensionality, i.e., with a large amount of input features. There are two paradigms that can help us to deal with such kind of data, i.e., dimension reduction and feature selection. Dimension reduction methods are usually based on mathematical projections, which attempt to transform the original features into an appropriate feature space. After dimension reduction, the original meaning of the features is usually lost. Feature selection methods directly select some original features to use, and therefore they can preserve the original meaning of the features, which is very desirable in many applications. However, feature selection methods are usually based on heuristics, lacking solid theoretical foundation. AdaBoost could be very useful in feature selection, especially when considering that it has solid theoretical foundation. AdaBoost often does not overfit, i.e., the test error of AdaBoost often tends to decrease even after the training error is zero.

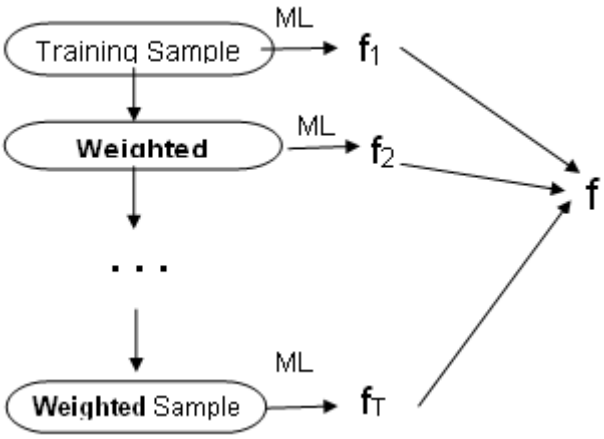
Boosting

Another innovation incorporated in system is adaptive boosting, based on the work of Rob Schapire and Yoav Freund. The idea is to generate several classifiers rather than just one. When a new case is to be classified, each classifier votes for its predicted class and the votes are counted to determine the final class.

But how can we generate several classifiers from a single dataset? As the first step, a single classifier is constructed as before from the training data. This classifier will usually make mistakes on some cases in the data; the first classifier, for instance, gives the wrong class for 7 cases. When the second classifier is constructed, more attention is paid to these cases in an attempt to get them right. As a consequence, the second classifier will generally be different from the first. It also will make errors on some cases, and these become the focus of attention during construction of the third classifier. This process continues for a pre-determined number of iterations or trials, but stops if the most recent classifiers are either extremely accurate or inaccurate.

Naturally, constructing multiple classifiers requires more computation that building a single classifier -- but the effort can pay dividends! Trials over numerous datasets, large and small, show that on average 10-classifier boosting reduces the error rate for test cases by about 5 to 10%.

The classifier constructed on Trial 0 is identical to that produced without the Boosting. Some of the subsequent classifiers produced by paying more attention to certain cases have relatively high overall error rates. Nevertheless, when the classifiers are combined, the final predictions have a lower error rate on the test cases.



Research shows that Neural networks techniques have significant power to support the use of Discriminative model. In addition, prediction accuracy of neural networks is generally high. Furthermore NNs are robust, works satisfactorily when training examples contain errors. Output of NN may be discrete, real-valued, or a vector of several discrete or real-valued attributes fast evaluation of the learned target function can also be achieved.

It is found that genetic search with Neural network provides good result, takes acceptable time to build the classifier model with acceptable computational complexity. This can further be enhanced by using multiclassifier approach.

The results of the studying indicated that the NN built using the proposed feature set has less features but the performance was comparable to Neural networks built using other feature sets generated. This can be optimized using some efficient method for feature selection like Genetic Algorithm with Evaluation schemes.

III. CLASSIFIER DESIGN CRITERIA AND EVALUATION STAGES

There are three primary criteria that have been applied to creating composite classifiers:

1. Accuracy of the component classifiers,
2. Diversity of the component classifiers, and
3. Efficiency of the entire composite classifier.

Evaluation Stages for Classification methods

Classification methods can be evaluated according to the following criteria.

- Predictive accuracy
  This refers to the ability of the model to correctly predict the class label of new or previously unseen data.
- Speed
  This refers to the computation costs involved in generating and using the model.
- Robustness
  This is the ability of the model to make correct predictions given noisy data or data with missing values.
- Scalability
  This refers to the ability of the learned model to perform efficiently on large amounts of data.
- Interpretability
  This refers is the level of understanding and insight that is provided by the learned model.

IV. ANALYSIS OF DATA FROM LITERATURE

Table I. Analysis of Data from Literature

Method	Output variable			Input variables		
	Continuous	Categorical		Continuous	Categorical	
		Ordinal	Nominal		Ordinal	Nominal
Multiple Linear Regression	Y	N	N	Y	Y	N
k-Nearest Neighbor Prediction	Y	N	N	Y	Y	N
Regression Tree	Y	N	N	Y	Y	N
Discriminant Analysis	N	Y	Y	Y	Y	N
Logistic Regression	N	Y	Y	Y	Y	N
Classification Tree	N	Y	Y	Y	Y	N
Naïve Bayes	N	Y	Y	N	Y	Y
Neural Networks Classification	N	Y	Y	Y	Y	N
Neural Networks Prediction	Y	N	N	Y	Y	N
k-Nearest Neighbor Classification	N	Y	Y	Y	Y	N
Association Rules	NA	NA	NA	N	Y (Binary Only)	Y
Principal Component Analysis	NA	NA	NA	Y	Y	N
k-Means Clustering	NA	NA	NA	Y	Y	N
Hierarchical Clustering	NA	NA	NA	Y	Y	N

Table II Comparison of Advantages of Classification Techniques

Neural Networks	Decision Tree	Naive Bayes	SVM-based	K- Nearest Neighbor
Good in many domain and works well with noisy data.	generated rules are easily observed (and modified)	This technique is fast, highly scalable model building and scoring. The build process for Naive Bayes is parallelized.	This classification is often more accurate than Decision Tree classification.	This technique is, it is Robust to noisy training data and it is effective if the training data is large.

Table III: Comparison of Classification based on Performance

Feature	Neural Networks	Decision Tree	Naive Bayes	Support Vector Machine	K- Nearest Neighbour
Learning Type	Eager Learner	Eager Learner	Eager Learner	Eager Learner	Lazy learner
Speed	Somewhat Slow	Fast	Very fast	Fast with active learning	Slow
Accuracy	Good in many domains	Good in many domains	Good in many domains	Significantly high	High – Robust
Scalability	Low	Efficient for small data set	Efficient for large data set	Remaining	Remaining
Interpretability	Bad	Good	Remaining	Remaining	Remaining
Transparency	black box	Rules	No rules (black box)	No rules (black box)	Rules
Missing value interpretation	Remaining	Missing value	Missing value	Sparse data	Missing value

Table IV: Comparison Based on Classification Parameter

a) Parameters	Neural Networks	Decision Tree	Naive Bayes	Support Vector Machine	K- Nearest Neighbour
Accuracy in general	3	2	1	4	2
Speed of learning with respect to number of attributes and the number of instances	1	3	4	1	4
Speed of classification	4	4	4	4	1
Tolerance to missing values	1	3	4	2	1
Tolerance to irrelevant Attributes	1	3	2	4	2
Tolerance to redundant Attributes	2	2	1	3	2
Dealing with discrete/binary/continuous attributes	3(not discrete)	4	3 (not continuous)	2(not discrete)	3 (not discrete)
Tolerance to noise	2	2	3	2	1
Dealing with Over fitting	1	2	3	2	3
Attempts for incremental Learning	3	2	4	2	4
Explanation ability/transparency of knowledge/classifications	1	4	4	1	2

Table IV: Comparison between various ensemble Techniques

	Boosting	Bagging	Stacking
Sampling Method	Least squares (proportionate)	Random, with replacement	Round-robin (cross-validation)
Splitting of Data	Width-wise	Length-wise	Length-wise
Guaranteed improvement of weak classifiers?	Yes	No	No
Hierarchical?	No	No, but can be extended	Yes
Training	Multiple passes	N/A	Single bottom-up pass
Wrapper or mixture?	Wrapper	Wrapper	Both
Sampling Method	Least squares (proportionate)	Length-wise	Round-robin (cross-validation)
Weak learner	Same	Same	Different
Training data become available	All at once	All at once	All at once
Can learn new classes?	No	No	No
Run weak learner on	Same, but differently weighted set	Bootstrap replicates of the training set	Same, but differently weighted set
Emphasizing "difficult" examples?	Yes	No	Yes
Is ensemble of ensemble?	No	No	Yes
Combination rule	Weighted majority voting	Simple majority voting	Meta learner

V. INFORMATION FUSION

- Information fusion is becoming a major need in Data Mining and Knowledge Discovery in database. Due to this need the interest on Information Fusion techniques is increasing in the Data Mining community.
- For e.g. typical applications of these techniques include the pre-processing step or data or information modelling step (e.g. ensemble methods).
- Nevertheless the gap between both data mining and information fusion areas is large.
- Large amounts of data are nowadays available because gathering data is easy and inexpensive.
- Most data is raw and to be useful relevant knowledge has to be extracted from it.
- Data Mining and Knowledge Discovery in Database are fields that study and provide methods for extracting this knowledge.
- Data Mining uses information fusion techniques for improving the quality of the extracted knowledge.
- Three main uses are:
- Information Fusion in pre-processing: Fusion is used to increase the quality of raw data prior to the application of DM methods.
- Information Fusion for building models: The model built from data uses some kind of Information Fusion techniques.
- Information Fusion for extracting information: The knowledge extracted from the data is the result of a particular Information Fusion technique (e.g. aggregated value computed from the data).
- Information Fusion is becoming a major requirement in data mining and knowledge discovery in database.

There are many methods to generate diverse individual classifiers for ensembles. These methods can be broadly categorized into three groups as follows.

- Using different structure or architecture for individual classifiers. For neural network ensembles, the individual networks can have different number of hidden layers and different number of hidden neurons, and/or are trained with different initial weights and biases.
- Using different training samples for individual classifiers. The two most popular methods in this group are bagging and boosting.
- Using different subsets of features for individual classifiers. Each individual classifier is trained with a different subset of features. The feature subsets can be obtained by random selection, or input decimation, or using genetic algorithms (GA).

Exact costs associated with different misclassification are rarely known and are difficult to estimate. As a result, these types of problems typically have the performance requirements being specified in the form of specific level of error for one or both types of errors.

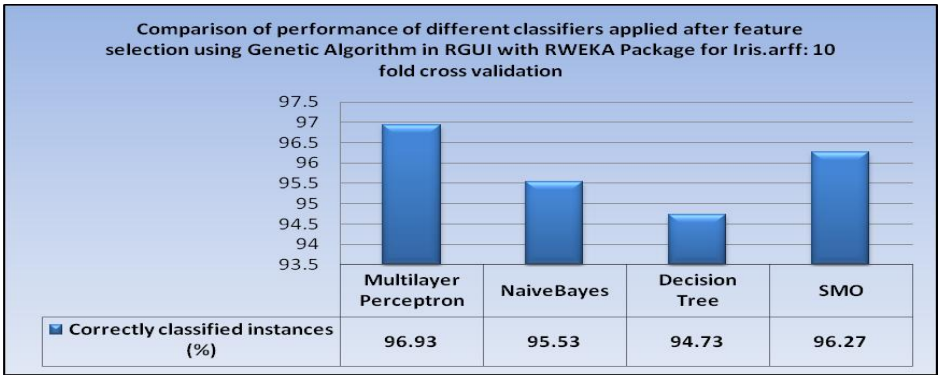


Fig II. Comparison of Performance of Different Classifiers after Feature Selection

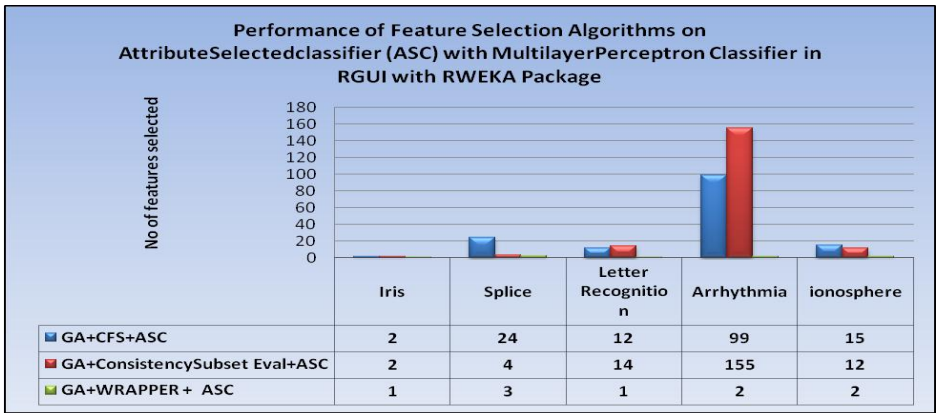


Fig III. Performance of Feature Selection Algorithm with AttributeSelectedClassifier (ASC)

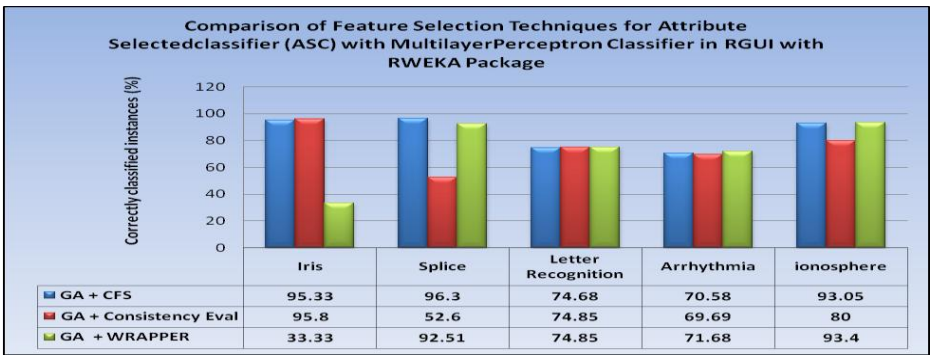


Fig IV. Comparison of Feature Selection Technique for Attribute Selected Classifier

VI. PROPOSED ALGORITHM

Input:
Sequence of N examples $S = [(xi, yi)]$, $i = 1, \ldots, N$
Learning algorithm of Neural Network: NN;
Integer T specifying number of iterations.
Initialize $D1(i) = 1/N$, $i = 1, \ldots, N$
Do for $t = 1, 2, \ldots, T$:
1. Select a training data subset St, drawn from the distribution Dt .
2. Train **NN** with S, receive hypothesis ht.
3. Calculate the error of ht : εt
If $\varepsilon t > 1/2$, **abort**.
4. Set $\beta t = \varepsilon t/(1 - \varepsilon t)$
5. Update distribution
$Dt : Dt+1(i) = [Dt(i)/Zt] \times \beta t$, *if* $ht(\mathbf{x}i) = yi$
$Dt : Dt+1(i) = [Dt(i)/Zt] \times 1$, *otherwise* (14)
where $Zt =$ is a normalization constant chosen so that $Dt+1$ becomes a proper distribution function.

Test -- Weighted Majority Voting: Given an unlabeled instance **x**,
1. Obtain total vote received by each class $Vj =$, $j=1,\ldots,C$
2. Choose the class that receives the highest total vote as the final classification.

- Ensemble learning using feature selection is more accurate than that without feature selection.
- The GA method can be modified by adding evaluator to obtain higher accuracy.
- Ensemble learning is more efficient than single classifier when dealing with high-dimensional datasets.
- The GA plays a good role of selector to select a subset of features that can best describe the classification information.
- The Comparative Analysis show that Ensemble of NN is as accurate and more robust than Decision Tree and SMO classifiers while it is more robust than Naïve Bayes but at the same time, is less efficient than it.

- Also, AdaBoost proves to be a better performer than LogitBoost, MultiBoost.AB, Bagging and Stacking.
- Key Components of algorithm can be improved :
  - The selection of the subsequent training dataset
  - The Base Classifier
- The algorithm parameters, such as base classifier architecture, error goal, number of hypotheses to be generated, are currently chosen by GA, there also any other selection strategy may be employed like Best First etc.

VII. CHARACTERISTICS OF FEATURE SELECTION METHODS

Feature selection aims to search the relevant features in the feature space. Researchers have studied various aspects of feature selection. From the point of view of heuristic search, Blum and Langley argue that the following four issues, which affect the nature of the search, can characterize any feature selection method.

1. The starting point in the feature space.

Depending on which point to start with, the search direction will vary. Search from no features and successively add others is called forward selection. In contrast, search from all features and successively remove features is called backward selection. A third method could be to combine forward and backward search.

2. The organization of the search procedure.

Obviously, if the number of features is too large, the exhaustive search of all the feature subspace is prohibitive, as there are 2N possible combinations for N features. For example, heuristic search is more realistic than exhaustive search, but it doesn't guarantee finding the optimal solutions.

3. The evaluation strategy.

How feature subsets are evaluated is an important problem. As for classification, the ideal feature subset should have the best separation of the data. Data separation is usually computed by an inter-class distance measure.

4. The criterion for stopping the search.

Without a suitable stopping criterion the FS process may run exhaustively or forever through the space of subsets.

During the process of evaluation, we might want to stop the search, when observing that there are no improvements of the classification accuracy. Stopping criteria can be:

5. whether addition (or deletion) of any feature does not produce a better subset; and
6. Whether an optimal subset according to some evaluation function is obtained.
7. Grouped different Feature Selection methods into two broad groups (i.e., filter and wrapper) based on their dependence on the inductive algorithm that will finally use the selected subset. Filter methods are independent of the inductive algorithm, whereas wrapper methods use the inductive algorithm as the evaluation function.

VIII. RESULTS AND DISCUSSIONS

AdaBoost.M1 is a versatile learning algorithm based on synergistic performance of an ensemble of weak classifiers/learners. Boosting single classifiers provides more accuracy than single base classifier. NNs are powerful tools for modeling causes and effects in a wide variety of domains. They are compact networks of probabilities that capture the probabilistic relationship between variables, as well as historical information about their relationships. Boosting these NNs form ensemble that retains the classification strengths of NN while increasing its accuracy. However the execution time may become somewhat higher.

IX. CONCLUSION AND FUTURE WORK

- Boosting is a practical tool for classification and other learning problems
- Grounded in rich theory
- Performs well experimentally
- Often (but not always!) resistant to overfitting
- Many applications and extensions
- Many ways to think about boosting
- None is entirely satisfactory by itself, but each useful in its own way
- Considerable room for further theoretical and experimental work

The algorithm has two key components, both of which can be improved. The first one is the selection of the subsequent training dataset; this is where we use GA. Any other selection strategy may be used like Best First, Simulated Annealing, Greedy forward/backward search etc. The second key factor is the Base Classifier which we have taken as NN (Multilayer Perceptrons). Many other classifiers can be employed for future work instead; like Support Vector Machines, Decision Tree Induction etc.

Here we obtain a homogenous ensemble of classifiers i.e. base classifier in each iteration of boosting is same i.e. NN. For future work, we can also use an ensemble system of heterogeneous classifiers.

The selection strategy used in GA in this work is the Tournament selection, any other selection strategy can be employed like Stochastic selection, Rank selection etc.

ACKNOWLEDGEMENT

The authors' wishes to thank all the colleagues for their guidance, encouragement and support in undertaking the research work. Special thanks to the Management for their moral support and continuous encouragement.

REFERENCES

1. Pei-Yong Xia, Xiang-Qian Ding, Bai-Ning Jiang, "A Ga-Based Feature Selection And Ensemble Learning For High-Dimensional Datasets", IEEE 2009, Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, 12-15 July 2009.
2. Laura Emmanuella A Santana, Diogo F de Oliveira, Anne M P Canuto and Marcilio C P de Souto, "A Comparative Analysis of Feature Selection Methods for Ensembles with Different Combination Methods", IEEE 2007, Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007.
3. Lin Yu Tseng and Shiueng Bien Yang, "Genetic Algorithms for Clustering, Feature Selection and Classification", IEEE 1997.
4. Faten Hussein, Nawwaf Kharma, and Rabab Ward, "Genetic Algorithms for Feature Selection and Weighting, a Review and Study", IEEE 2001.

5. Daming Shi & Wenhao Shu and Haitao Liu, "Feature Selection for Handwritten Chinese Character Recognition Based on Genetic Algorithms", IEEE 1998.
6. Laura E A Santana, Ligia Silva and Anne M P Canuto, "Feature Selection in Heterogeneous Structure of Ensembles: A Genetic Algorithm Approach", IEEE 2009, Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA, June 14-19, 2009.
7. Jihoon Yang and Vasant Honavar, "Feature Subset Selection Using a Genetic Algorithm", IEEE Intelllgent Systems, 1998.
8. Michael L. Raymer, William F. Punch, Erik D. Goodman, Leslie A. Kuhn, and Anil K. Jain, "Dimensionality Reduction Using Genetic Algorithms", IEEE Transactions On Evolutionary Computation, Vol. 4, No. 2, July 2000. (Journal Paper).
9. N. Srinivas and Kalyanmoy Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms", Journal of Evolutionary Computation, Vol. 2, No. 3. (Journal Paper).
10. Mr. Kashif Waqas, Dr. Rauf Baig1and, Dr. Shahid Ali, "Feature Subset Selection Using Multi-Objective Genetic Algorithms", IEEE 2009.
11. L. S. Oliveira, R. Sabourin, F. Bortolozzi, and C. Y. Suen, "Feature Selection for Ensembles: A Hierarchical Multi-Objective Genetic Algorithm Approach", IEEE, 2003.
12. Kushan Ahmadian, Abbas Golestani, Nasser Mozayani, and Peyman Kabiri, "A New Multi-Objective Evolutionary Approach for Creating Ensemble of Classifiers", IEEE, 2007.
13. Abdullah Konaka, David W. Coitb and Alice E. Smith, "Multi-objective optimization using genetic algorithms: A tutorial", Reliability Engineering and System Safety, Available at www.ElsevierComputerScience.com.
14. Christm Emmanouilidis, Andrew Hunter, John Machtyre, and Chris Cox, "Multiple - Criteria Genetic Algorithms for Feature Selection in Neurofuzzy Modeling", IEEE, 1999.
15. L. S. Oliveira, R. Saburin, F. Bortolozzi, and C. Y. Suen, "Feature Selection Using Multi-Objective Genetic Algorithms for Handwritten Digit Recognition", IEEE, 2002
16. Pier Luca Lanzi & Politecnico di Milano, "Fast Feature Selection with Genetic Algorithms: A Filter Approach", IEEE 1997.
17. Huan Liu and Rudy Setiono, "A Probabilistic Approach to Feature Selection – A Filter Solution".
18. Hussein Almuallim and Thomas G. Dietterich, "Learning Boolean concepts in the presence of many irrelevant features", Artificial Intelligence 69 (1994) 279-305, Available at www.ElsevierComputerScience.com.
19. Mark A. Hall, "Correlation based Feature Selection for Machine Learning", Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy at The University of Waikato.
20. Nigel Williams, Sebastian Zander, Grenville Armitage, "A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification", ACM SIGCOMM Computer Communication Review , Volume 36, Number 5, October 2006. (Journal Paper).
21. Frans Coenen and Paul Leng, "Obtaining Best Parameter Values For Accurate Classification", IEEE, 2005.Alden H. Wright, "Genetic Algorithms for Real Parameter Optimization".
22. Zhang Dezhen and Yang Kai, "Genetic Algorithm based Optimization for AdaBoost", International Conference on Computer Science and Software Engineering, 2008.
23. Dragos D. Margineantu and Thomas G. Dietterich, "Pruning Adaptive Boosting", ICML, 1997.
24. Simon Thompson, "Genetic Algorithms as Post Processors for Data Mining", AAAI Technical Report WS-99-06, 1999.
25. Janaki Gopalan, Reda Alhajj and Ken Barker, "Discovering Accurate and Interesting Classification Rules Using Genetic Algorithm".
26. Karina Gibert, Joaquín Izquierdo, Geoff Holmes, Ioannis Athanasiadis, Joaquim Comas, Miquel Sànchez-Marrè, "On the role of pre and post-processing in environmental data mining", International Congress on Environmental Modelling and Software Integrating Sciences and Information Technology for Environmental Assessment and Decision Making, 2008.
27. Manoranjan Dash and Huan Liu, "Consistency-based search in feature selection", Artificial Intelligence 151 (2003) 155–176.