A STUDY ON THE DATA CLASSIFICATION ALGORITHMS AND CLASSIFIER MODEL BASED ARTIFICIAL NEURAL NETWORK AND FUZZY LOGIC

R.Siva Kumar[1] & Dr.M.Sridhar[2]
[1]Research Scholar, Dept. of ECE, Bharath University, India.
[2]Dept. of ECE, Bharath University, India.

Abstract **- This paper aims to provide a brief overview of some of the important classification algorithm proposed in the research and there are number of data classification algorithms have been reported. This includes the Statistical approach and machine learning approaches of data classification algorithms. This paper also gives an outline of the proposed work on data classifier design using intelligent techniques such as Artificial Neural Network and Fuzzy Logic.**

Index Terms **- Data Classification Algorithm, Statistical approach, machine learning approaches, Artificial Neural Network approach, Fuzzy Logic approach**.

I. INTRODUCTION

Data classification is the task of assigning a data into a predefined group or class based on a number of observed input features related to that data. Demands on automatic data classification systems are rising enormously due to the availability of large databases and stringent performance requirements (speed, accuracy, and cost) [1]. The rapidly growing and available computing power while enabling faster processing of huge data sets has facilitated the use of elaborate and diverse methods for data analysis and classification.

II. REVIEW OF DATA CLASSIFICATION ALGORITHMS

Most of the data classification algorithms proposed in the literature is based on statistical and machine learning approach in which individual items are placed into groups based on quantitative information on one or more characteristics inherent in the items and based on a training set of previously labeled items. Some of the popular and widely used classification algorithm are Decision Tree (Wu et al, 1975 and Swain et al, 1977), Bayes Classification (Ben Bassat et al, 1980), k-nearest neighbor (Dasarathy, 1991), Hidden Morkov Model (Lawrence, 1989), and Case based reasoning (Marir and Watson, 1994).

Statistical approaches Bayes classification is a simple probabilistic classifier based on Bayes theorem that can predict class membership probabilities i.e., the probability that a given sample belongs to a particular class. Pedro et al, (1997) and Christopher et al, (1998) used the Bayesian approach for data classification [3]. In these procedures, an underlying probability model must be assumed in order to calculate the posterior probability upon which the classification decision is made. The problem with this approach is that they work well only when the underlying assumptions are satisfied. Also the effectiveness of this approach depends to a large extent on the various assumptions or conditions under which the models are developed.

Edgoose and Allison, (1999) and Li et al, (2000) uses Hidden Markov Model (HMM) for data classification. HMM is used to statistically model a classification process that varies in time. It can be seen as a doubly embedded stochastic process with a process that is not observable (hidden process) and can only be observed through another stochastic process (observable process) that produces the time set of observations [4]. HMM is suitable for modeling time-varying signals or classification tasks where data exist across a sequence of observations

A. Machine Learning approaches

A decision tree is a flow-chart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and leaf nodes represent classes or class distribution. Mui and Fu, (1980) used a binary decision tree classifier for automatic classification of nucleated blood cells [4]. A decision tree is built by repeatedly splitting the data to be classified into smaller and smaller partitions. This prone to the problems of fragmentation, repetition and replication. Furthermore, decision trees can lead to large errors if the number of training samples per class is small. For larger problems with lots of training data and many attributes, finding the split at each node can become expensive.

Denoeux, (1995) and Trevor et al, (1996) use k-Nearest Neighbor approach for data classification in which all the training samples are stored in an n-dimensional pattern space[5]. When an unknown sample is given, it searches the pattern space for the k training samples that are closest to the unknown sample which is find out by calculating Euclidean distance. Data classification based on k-nearest neighbor are instance based or lazy learners in that they store all of the training samples and do not build a classifier until a new sample needs to be classified. They can incur expensive computational costs when the number of potential neighbors with which to compare a given unlabeled sample is great.

Jurisica and Glasgow, (1997) and Lenz et al, (1998) use Case-based reasoning for data classification [4]. Classifiers developed using case based reasoning are instance based. When given a new case to classify, a case-based reasoner will first check if an identical training case exists. If one is found, then the accompanying solution to that case is returned. If no identical case is found, then the case based reasoner will search for training cases having components that are similar to those of the new case. The major problem with this approach is that when data of very large size is provided, then it takes a long time for searching which results in high computation cost.

III. Data Classification Using Artificial Neural Networks

Artificial Neural Network (Guoqiang, 2000) has emerged as an important tool for data classification where a classification procedure seeks a functional relationship between the set of input features and the output class label. The classifiers based on neural networks are non-parametric adaptive data classifiers inspired by biological neural network that are developed to provide high performance and real time response with real world data.

Lippmann, (1989) categorize the neural network based classifiers into three groups[5]: Hyperplane classifiers, Kernel classifiers and Exemplar classifiers. Classifiers

A.Hyperplane classifier

Hyperplane classifiers form complex decision regions using nodes that form hyperplane decision boundaries in the space spanned by the inputs. Then the node calculates a weighted sum of the inputs and passes this sum through a nonlinear activation function. Examples of this type of classifiers are multi-layer perceptron (MLP) (kurt, 1991) and boltzman machine (Lin and Lee, 1995) [6].

B.Kernel classifiers

Kernel classifiers create complex decision regions from kernel function nodes that form overlapping receptive fields. Kernel functions typically have gaussian shapes, and the smoothing parameter or width of the kernel may be fixed or may vary across nodes. Smoothing parameters and the number of nodes are typically adjusted empirically to provide good performance on test data. Radial basis function network (Ghosh and Chakravarthy, 1994) is the popular form of kernel classifiers [7].

C.Exemplar classifiers

Exemplar classifiers perform classification based on the identity of the training examples, or exemplars, that are nearest to the input. Exemplar nodes compute weighted Euclidean distance between inputs and the centroids correspond to previously presented labeled training samples. Exemplar classifiers include Self Organizing Map (SOM)(Suganthan, 1999) classifiers, the Learning Vector Quantizer (LVQ) (Baras and dey, 1999) and Adaptive Resonance Theroy (ART) (Muchoney and Williamson, 2001)[7].

D.Techniques to Improve Performance

A neural network based classifier needs lot of data to develop the network before being put to use for real time applications. Getting the real time data in sufficient amount for each and every type of an engineering application is not an easy job. To make the neural network approach applicable for large scale problems, some dimensionality reduction is mandatory. Feature extraction and feature selection are two forms dimensionality reduction for ANN based classifier system that can improve the performance of the ANN based classifier design.

Principle component analysis (PCA) is a popular feature extraction technique that reduce dimension without loss of the intrinsic information contained in the original data. Karhunen and Joutsensalo, (1995) have discussed many aspects of PCA performed by neural networks. One problem with PCA is that it is a kind of unsupervised learning procedure and does not consider the correlation between target outputs and input features.

A number of heuristic measures have been proposed to estimate the relative importance or contribution of input features to the output variable. One of the simplest measures is the sum of the absolute input weights (Sen et al, 1995) to reflect the impact of that input variable on the output. The limitation of this measure is obvious since it does not consider the impact of perhaps more important hidden node weights.

Steppe and Bauer, (1996), Steppe et al, (1996), and Hu et al, (1996) use the Bonferroni-type or likelihood-ratio test statistic as the model selection criterion and the backward sequential elimination approach to select features. Based on Garson’s measure of saliency, Glorfeld, (1996) presents a backward elimination procedure to select more predictive feature variables.

Steppe and Bauer, (1997) classify all feature saliency measures used in neural networks into derivative-based and weight-based categories with the former measuring the relative changes in either neural network output or the estimated probability of error and the latter measuring the relative size of the weight vector emanating from each feature[8]. Since exhaustive search through all possible subsets of feature variables is often computationally prohibitive, heuristic search procedures such as forward selection and backward elimination are often used. Setiono and Liu, (1997) also develop a backward elimination method for feature selection [7]. Although significant progress has been made in data classification using neural networks, a number of issues in applying neural networks still remain and have not been solved successfully or completely.

IV.PROPOSED ANN BASED CLASSIFIER MODEL

A feed forward neural network trained by back propagation algorithm is proposed in this paper for performing the data classification task. The classifier model developed using ANN is supposed to work autonomously. The basic characteristic of the design process is the performance level, i.e., the correct classification percentage. In the development of ANN-based classifier model, the following issues are addressed:

Selection of suitable subset of features: Selection of suitable subset input feature variables is an important issue in building neural classifiers. The purpose of feature variable selection is to find the smallest set of features that can result in satisfactory predictive performance. Because of the curse of dimensionality, it is often necessary and beneficial to limit the number of input features in a classifier in order to have a good predictive and less computationally intensive model. Numerous statistical feature selection criteria and search algorithms have been developed in the pattern recognition literature. Mutual information based feature selection technique is proposed in this paper to select the relevant input features of the neural classifier system.

Model Configuration and validation: Once the appropriate features of the model are identified, the model is configured to capture the underlying relationship between the input and output using the training data. The trained models are tested with a separate set of input-output data to assess its generalization capability.

V.Fuzzy Logic For Data Classification

Fuzzy Logic (Timothy, 1995) has been successfully applied in solving classification problems where boundaries between classes are not well defined. The approach considered in fuzzy classification is to create so-called "fuzzy category memberships functions", which convert an objectively measurable parameter into a subjective "category memberships", which are then used for classification. There are two main categories of fuzzy classifiers: pure fuzzy classifiers and fuzzy rule based classifiers.

Pure fuzzy classifications methods are based on fuzzy clustering (Bezdek, 1981), fuzzy pattern matching (Debois et al, 1988), fuzzy integral (Grabish and Sugeno, 1992)[8]. These methods are poorly suited for classification problems because of their lack of normalization. Also these methods do not make use of training data and have unacceptable performance.

Fuzzy Rule Based Systems (FRBS) have been successfully applied to many control (Sugeno, 1985 and Lee, 1990), modeling (Takagi et al, 1985 and Sugeno et al, 1993), and classification problems (Ishibuchi et al, 1995) [9]. The key to the success of the FRBS is its ability to incorporate human expert knowledge. Typical fuzzy rule based classifiers consist of interpretable if-then rules with fuzzy antecedents and class labels in the consequent part. The antecedents (if-parts) of the rules partition the input space into a number of fuzzy regions by fuzzy sets, while the consequents (then-parts) describe the output of the classifier in these regions. An important issue in the design of fuzzy rule based classifier system is the formation of fuzzy if-then rules and the membership functions.

In general the rules and membership function are formed from the experience of the human experts. With an increasing number of variables, the possible number of rules increases exponentially, which makes it difficult for experts to define a complete rule set for good system performance. Data-driven approaches (Jang, 1992 and Wang et al, 1992) have been proposed for developing the fuzzy rule based system from numerical data without domain experts [10].

Abe et al, (1995) proposed a rule generation method in which each fuzzy if-then rule was represented by a hyper box in multidimensional pattern spaces. But they are very weak in self learning and determining the required number of fuzzy if-then rules. Ishibuchi et al, (1992 and 1996) proposed a heuristic method for generating fuzzy if-then rules for pattern classification problems using grid-type fuzzy partition in which a priori knowledge on linguistic values is required for specifying the membership function which fails to handle high dimensional problems with many input variables due to the curse of dimensionality.

During the late 1990s attempts have been made to improve the learning capability of the fuzzy rule based systems using soft computing techniques. Genetic Fuzzy Rule Based System (GFRBS) is one such approach in which a fuzzy rule based system is augmented by a learning process based on a Genetic Algorithm (GA) (Goldberg, 1989). Genetic Algorithms are search algorithms based on the mechanics of natural genetics. While many papers have dealt with the application of GA for generating fuzzy if-then rules and tuning membership functions for system identification and control problems, only a few papers have focused on pattern classification problems.

Cordon et al, (2004) gives a brief review of most of the approaches found in the literature on GFRBS. Accordingly, the GFRBS proposed in the literature which falls into four categories: Learning fuzzy rules with fixed fuzzy membership functions (Ishibuchi et al, 1999), Learning fuzzy membership functions with fixed fuzzy rules (Yuhui et al, 1999), Learning fuzzy rules and membership functions in stages (i.e., first evolving good fuzzy rule sets using fixed membership function, then tuning membership functions using the derived fuzzy rule sets (Setnes, 2000 and Roubos, 2001) and Learning fuzzy rules and membership functions simultaneously (Wang, 1998 and Russo, 2000). This paper follows the last approach.

Fuzzy system modeling usually comes with two contradictory requirements to the obtained model: the interpretability capability to express the behavior of the real system in a comprehensive way, and the accuracy to faithfully represent the real system. To design a fuzzy system with the above objectives, improvements have been made over the binary-coded genetic algorithm proposed in (Goldberg, 1989). An enhanced genetic algorithm was proposed in (Cheong and Lai, 2000) for optimizing a fuzzy logic controller. Rojas et al, (2001) proposed a parallel GA to learn FRBS.

A.Genetic Algorithm Approaches

Real coded genetic algorithm was proposed in (Yuhui et al, 1999 and Setnes et al, 2000) for GA-Fuzzy modeling and classification. Yuhui et al, (1999) used two-point cross over with a default cross over probability of 0.75 (Goldberg, 1989). Setnes et al, (2000) used simple arithmetic cross over operator in addition with that two special real coded operators used are whole arithmetic cross over and heuristic cross over. A number of cross over operators are used in the literature for real coded parameter. Herrera et al, (2003) gives a detailed taxonomy for the crossover operators for the real coded genetic algorithms.

A hybrid GA approach which uses a combination of binary strings and floating point numbers are proposed in (Wang et al, 1998, Russo et al, 2000, Casillas et al, 2005, Alcala et al, 2006 and Alcala et al, 2007). Wang et al, (1998) used two-substring cross over for both the binary string and real parameter. Russo et al, (2000) used multi-cut crossover for binary strings and weighted mean cross over for real parameter.

Casillas et al, 2005 used a threefold coding scheme that uses real and integer coded chromosomes [11]. The author has used max-min arithmetical crossover for real part and two-point crossover for integer part.

Alcala et al, (2006) proposed a double coding scheme for both rule selection and weight derivation and used two-point crossover for binary string and max-min arithmetical crossover for real parameter[12]. Alcala et al, (2007) used two different kinds of coding schemes based on the two different types of tuning (global tuning of the semantics and local tuning of the rules)[13][14]. For both cases, a real coding is considered and uses Parent Centric BLX (PCBLX) and HUX crossover operators.

Even though a number of works have been reported in the literature in designing a genetic-fuzzy system for data classification, still there remain a number of issues in applying genetic algorithms in terms of problem representation and genetic operations.

VI.PROPOSED WORK ON FUZZY CLASSIFIER DESIGN

The classifier model developed using Fuzzy Logic is used as a helping tool for the user in the decision processes. This paper proposes Genetic Algorithm for optimal design of fuzzy classifier model. Initially the fuzzy classifier model is designed using Simple Genetic Algorithm (SGA) that uses binary representation and basic genetic operators only.

In a standard Simple Genetic Algorithm, crossover is the main genetic operator responsible for the exploitation of information while mutation brings new nonexistent bit structures. It is widely recognized that the SGA scheme is capable of locating the neighborhood of the optimal or near-optimal solutions, but, in general, SGA requires a large number of generations to converge.

To address this issue, an Improved Genetic Algorithm (IGA) is developed which incorporates a set of advanced and problem-specific genetic operators namely Gene Cross-Swap Operator (GCSO), Gene Inverse Operator (GIO) and Gene Max-Min Operator (GMMO) in addition to the basic cross over and mutation operator applied in SGA.

The conventional binary-coded GA has Hamming cliff problems (Devaraj et al, 2005) which sometimes may cause difficulties in the case of coding continuous variables. Also, for discrete variables with total number of permissible choices not equal to 2^k (where k is an integer) it becomes difficult to use a fixed length binary coding to represent all permissible values.

To overcome the above difficulties a Mixed Genetic Algorithm (MGA) is proposed in which a mixed form of representation is followed to encode the membership function and the rule set. In the proposed MGA, floating point numbers are used to represent the membership function and binary strings are used to represent the rule set. This type of representation has a number of advantages over binary coding for membership function.

The efficiency of the GA is increased as there is no need to convert the input variables to the binary type. For effective genetic operation, modified forms of crossover and mutation operators which can deal with the mixed string are proposed. This proposed MGA improves the convergence speed and the quality of the solution compared with binary coded genetic algorithms.

VII. CONCLUSION

In this paper, details of data classification algorithm are presented with a brief review on the classification algorithms, Classifier design using Artificial Neural Network and Fuzzy Logic for data classification. Also this paper discusses the proposed classifier model and their issues.

REFERENCES

[1] Abe S. and Lan M.S. (1995) ‘A Method for fuzzy rule extraction directly from numerical data and its application to pattern classification’, IEEE Trans. Fuzzy Syst., Vol. 3, No. 1, pp.18-28.
[2] Alcalá R., Alcalá-Fdez J. and Herrera F. (2006) ‘Hybrid learning models to get the interpretability-accuracy trade-off in Fuzzy Modelling’, Soft Computing, Vol.10, No. 9, pp. 717-734.
[3] Alcalá R., Alcalá-Fdez J. and Herrera F. (2007) ‘Rule Base Reduction and Genetic Tuning of Fuzzy Systems based on the Linguistic 3-Tuples Representation’, Soft Computing, Vol.11, No. 5, pp. 401-419.
[4] Almuallim, H. and Dietterich,T.G. (1994) ‘Learning boolean concepts in the presence of many irrelevant features’, Artificial Intelligence, Vol. 69, No. 1-2, pp. 279-305.
[5] Baras J.S. and Dey S. (1999) ‘Combined compression and classification with learning vector quantization’, IEEE Trans. on Information Theory, Vol. 45, No. 6, pp. 1991-1920.
[6] Ben-Bassat M., Klove K. L. and Weil M. H. (1980) ‘Sensitivity analysis in Bayesian classification models: Multiplicative deviations’, IEEE Trans. Pattern Anal. Mach. Intell., Vol. 2, pp. 261–266.
[7] Bezdek J.C. (1981) ‘Pattern Recognition with fuzzy objective function algorithms’, New York: Plenum.
[8] Bishop C. M. (1995) ‘Neural Networks for Pattern Recognition’, Oxford University Press, Oxford, New York.
[9] Casillas J., Cordon O. and Herrera F. (2005) ‘Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction’, IEEE Transactions on Fuzzy Systems, Vol. 13, No. 1, pp. 13-28.
[10] Cheong F. and Lai R. ‘Constraining the optimization of a fuzzy logic controller using an enhanced genetic algorithm’, IEEE Trans. Systems Man Cybernet., Part- B, Vol. 30, No.1, pp.31-46.
[11] Christopher K.I. Williams and David Barber (1998) ‘Bayesian classification with Gaussian process’ IEEE Trans. Pattern Anal. Mach. Intell., Vol.20, No.12, pp.1342-1351.
[12] Comon P. (1994) ‘Independent Component Analysis, a new concept?’, Signal Processing, Elsevier, Vol.36,No.3,pp.287-314.
[13] Corcoran A.L. and Sen S. (1994) ‘Using real-valued genetic algorithms to evolve rule sets for classification’, Proc. IEEE Conf. Evolut. Computation, Orlando, Vol.1, pp. 120-124.
[14] Cordon O., Gomide F. and Herrera F. (2004) ‘Ten Years of Genetic Fuzzy Systems: Current Framework and New Trends’, Fuzzy Sets and Systems, Vol.141, No.1, pp. 5-31.