# Genetic Feature Selection for Optimal Functional Link Artificial Neural Network in Classification

Satchidananda Dehuri[1], Bijan Bihari Mishra[2], and Sung-Bae Cho[1]

[1] Soft Computing Laboratory, Department of Computer Science,
Yonsei University, 262 Seongsanno, Seodaemun-gu, Seoul 120-749, Korea
satchi.lapa@gmail.com, sbcho@yonsei.ac.kr
[2] Department of Computer Science and Engineering,
College of Engineering Bhubaneswar, Patia, 751024, Orissa, India
misrabijan@gmail.com

**Abstract.** This paper proposed a hybrid functional link artificial neural network (HFLANN) embedded with an optimization of input features for solving the problem of classification in data mining. The aim of the proposed approach is to choose an optimal subset of input features using genetic algorithm by eliminating features with little or no predictive information and increase the comprehensibility of resulting HFLANN. Using the functionally expanded selected features, HFLANN overcomes the non-linearity nature of problems, which is commonly encountered in single layer neural networks. An extensive simulation studies has been carried out to illustrate the effectiveness of this method over to its rival functional link artificial neural network (FLANN) and radial basis function (RBF) neural network.

**Keywords:** Classification, Data mining, Genetic algorithm, FLANN, RBF.

## 1 Introduction

For the past few years, there have been a lot of studies focused on the classification problem in the field of data mining [1,2]. The general goal of data mining is to extract knowledge from large gamut of data. The discovered knowledge should be predictive and comprehensible. Knowledge comprehensibility is usually important for at least two related reasons. First, the knowledge discovery process usually assumes that the discovered knowledge will be used for supporting a decision to be made by a human user. Second if the discovered knowledge is not comprehensible to the user, he/she will not be able to validate it, hindering the interactive aspect of the knowledge discovery process, which includes knowledge validation and refinement. In this work the proposed method for classification is given an equal importance to both predictive accuracy and comprehensibility. We are measuring comprehensibility of the proposed method by reducing the architectural complexity. As we know the architectural complexity of FLANN [3] is directly proportional to number of features and the functions considered for expansion of the given feature value. Therefore, for reducing the architectural complexity we first select a subset of features (i.e. feature selection [4]) and

C. Fyfe et al. (Eds.): IDEAL 2008, LNCS 5326, pp. 156–163, 2008.

then applying the usual procedure of function expansion and training by back propagation learning. As the selection and learning is accomplished by hybridization of FLANN with genetic algorithms (GAs) [5], we named this method as hybrid FLANN (HFLANN).

Neural networks [6] have emerged as an important tool for classification. Pao et al. [7] shows a direction that their proposed FLANN may be conveniently used for function approximation and can be extended for classification with faster convergence rate and lesser computational load than an multi-layer perceptron (MLP) structure. The FLANN is basically a flat network and the need of the hidden layer is removed and hence the learning algorithm used in this network becomes very simple. The functional expansion effectively increases the dimensionality of the input vector and hence the hyper planes generated by the FLANN provide greater discrimination capability in the input pattern space. Although many types of neural networks can be used for classification purposes [7], we choose radial basis function neural network and our previous work FLANN for classification [3] as the benchmark method for comparison.

## 2 Functional Link Artificial Neural Network

The FLANN architecture uses a single layer feed forward neural network by removing the concept of hidden layers. This may sound a little harsh at first, since it is due to them that non-linear input-output relationships can be captured. Encouragingly enough, the removing procedure can be executed without giving up non-linearity, provided that the input layer is endowed with additional higher order functionally expanded units of the given pattern. The weighted sum of the functionally expanded features is fed to the single neuron of the output layer. The weights are optimized by the gradient descent method during the process of training.

The set of functions considered for function expansion may not be always suitable for mapping the non-linearity of the complex task. In such cases few more functions may be incorporated to the set of functions considered for expansion of the input dataset. However, dimensionality of many problems itself are very high and further increasing the dimensionality to a very large extent may not be an appropriate choice. So, it prompts us a new research direction to design HFLANN. The HFLANN harness the power of genetic algorithms (GAs) to reduce the dimensionality of the problem.

## 3 Proposed Method

The proposed HFLANN is a single layer ANN with a genetically optimized set of features. It has the capability of generating complex decision regions by non-linear enhancement of hidden units referred to as functional links. Figure 1 shows the topological structure of the HFLANN. The proposed method is characterized by a set of FLANN with a different subset of features.

Let $n$ be the number of original features of the data domain. The number of features selected to become a chromosome of the genetic population is $d$, $d \leq n$. The $d$ varies from chromosomes to chromosomes of the genetic population (i.e. $1 \leq d \leq n$). For simplicity, let us see how a single chromosome with $d$ features is working cooperatively for HFLANN.

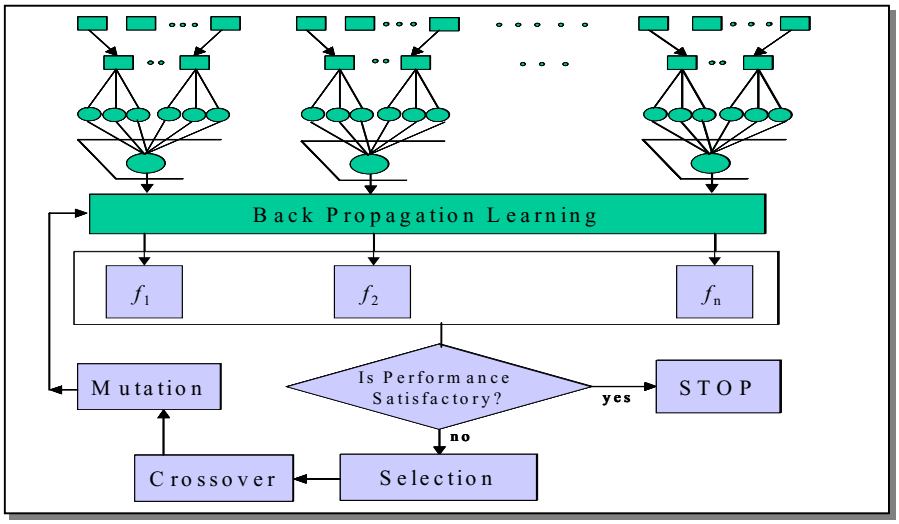


**Fig. 1.** Topological Structure of the HFLANN

In this work, we have used the general trigonometric function for mapping the $d$ feature from low to high dimension. However, one can use a function that is very close to the underlying distribution of the data but it requires some prior domain knowledge. Here we are taking five functions out of which four are trigonometric and one is linear (i.e., keeping the original form of the feature value). Among the four trigonometric functions-two are sine and two are cosine functions. In the case of trigonometric functions the domain is feature values and range is a real number lies between [-1,1]. It can be written as

$$f : D \rightarrow R^{[-1,1]\cup\{x\}}, \tag{1}$$

where $D = \{x_{i1}, x_{i2}, ...., x_{id}\}$, and $d$ is known as the number of features.

In general let us take $f_1, f_2, ...., f_k$ be the number of functions used to expand each feature value of the pattern. Therefore, each input pattern can now be expressed as

$$\begin{aligned} &\vec{x}_i = \{x_{i1}, x_{i2}, ...., x_{id}\} \\ &\rightarrow \{\{f_1(x_{i1}), f_2(x_{i1}), ..... , f_k(x_{i1})\}, ...., \{f_1(x_{id}), f_2(x_{id}), ...., f_k(x_{id})\}\}, \\ &= \{\{y_{11}, y_{21}, ....., y_{k1}\}, ....., \{y_{1d}, y_{2d}, ....., y_{kd}\}\} \end{aligned} \tag{2}$$

The weight vector between hidden layer and output layer is multiplied with the resultant sets of non-linear outputs and are fed to the output neuron as an input. Hence the weighted sum is computed as follows:

$$s = \sum_{j=1}^{m} y_{ij}.w_j \text{ , } I=1,2, ...,N \text{ and } m \text{ be the total number of expanded features.} \tag{3}$$

The network has the ability to learn through back propagation learning. The training requires a set of training data, i.e., a series of input and associated output vectors. During the training, the network is repeatedly presented with the training data and the

weights adjusted by back propagation learning from time to time till the desired input-output mapping occurs. Hence, the estimated output is computed by the following metric:

$$\hat{y}_i(t) = f(s_i)\ ,\ i=1,2,...,N.$$

The error $e_i(t) = y_i(t) - \hat{y}_i(t)$ , $i=1,2,...,N$ be the error obtained from the $i^{th}$ pattern of the training set. Therefore, the error criterion function can be written as,

$$E(t) = \sum_{i=1}^{N} e_i(t)\ , \tag{4}$$

and our objective is to minimize this function by gradient decent approach until $E \leq \varepsilon$ .

This process is repeated for each chromosomes of the GA and subsequently each chromosome will be assigned a fitness value based on its performance. Using this fitness value the usual process of GA is executed until some good topology with an acceptable predictive accuracy is achieved.

### 3.1 High Level Algorithms for HFLANN

The specification of the near optimal HFLANN architecture and related parameters can be obtained by both genetic algorithms and back-propagation learning, as it is explained in the following. Evolutionary algorithms of genetic type are stochastic search and optimization methods. Principally based on computational models of fundamental process, such as reproduction, recombination and mutation. An algorithm of this type begins with a set (population) of estimates (genes), called individuals (chromosomes) appropriately encoded. Each one is evaluated for its fitness in solving the classification task of data mining. During each iteration (algorithm time-step) the most-fit individuals are allowed to make and bear offspring.

**Individual Representation**
For the evolutionary process the length of each particle is *n* (i.e. the upper bound of a feature vector). Each cell of the chromosome contains binary value either 0 or 1. The cell value controls the activation (the value of 1 is assigned) or deactivation (the value of 0 is assigned) of the functional expansion for individuals.

**Objective Function**
During evolution each individual measures its effectiveness by the error criterion function using equation (4) and the predictive accuracy is assigned as it corresponding fitness.

**Pseudocode**
The major steps of HFLANN can be described as follows:

1. DIVISION OF DATASET
   Divide the dataset into two parts: training and testing
2. RANDOM INITIALIZATION
   Initialize each individual randomly from the domain {0,1}.

3. REPEAT
4. FOR THE POPULATION
   4.1 FOR each sample of the training set
   4.2 MAPPING OF INPUT PATTERN
       Map each pattern from low to high dimension, i.e. expand each feature value according to the predefined set of functions.
   4.3 CALCULATE the weighted sum and feed as an input to the node of the output layer.
   4.4 CALCULATE the error and accumulate it.
   4.5 BACK PROPAGATION LEARNING
       Minimize the error by back propagation learning.
   4.6 ASSIGN THE FITNESS
5. FOR THE POPULATION
   5.1 Perform Roulette Wheel Selection to obtain the better chromosomes.
6. FOR THE POPULATION
   6.1 Perform recombination
   6.2 Mutation
7. UNTIL < *Maximum Iteration is Reached*>

## 4 Experimental Studies

The performance of the EFLANN model was evaluated using a set of ten public domain datasets like IRIS, WINE, PIMA, BUPA, ECOLI, GLASS, HOUSING, LED7, LYMPHOGRAPHY and ZOO from the University of California at Irvine (UCI) machine learning repository [8]. In addition we have taken VOWEL dataset to show the performance of HFLANN for classifying six overlapping vowel classes [9]. We have compared the results of HFLANN with other competing classification methods such as radial basis function network (RBF) and our previously proposed FLANN with gradient descent. Table 1 summarizes the main characteristics of the databases that have been used in this paper.

**Table 1.** Summary of the Dataset used in Simulation Studies

| Sl. No. | Dataset | Instances | Attribute | Classes |
|---|---|---|---|---|
| 1 | IRIS | 150 | 4 | 3 |
| 2 | WINE | 178 | 13 | 3 |
| 3 | PIMA | 768 | 8 | 2 |
| 4 | BUPA | 345 | 6 | 2 |
| 5 | ECOLI | 336 | 7 | 8 |
| 6 | GLASS | 214 | 9 | 6 |
| 7 | VOWEL | 871 | 3 | 6 |
| 8 | HOUSING | 506 | 13 | 5 |
| 9 | LED7 | UD | 7 | 10 |
| 10 | LYMPHOGRAPHY | 148 | 18 | 4 |
| 11 | ZOO | 101 | 16 | 7 |

### 4.1 Parameter Setup

For evaluating the proposed algorithm, the following user defined parameters and protocols related to the dataset need to be set beforehand.

A two fold cross validation is carried out for all the dataset by randomly dividing the dataset into two parts (datasets1. dat and dataset2.dat). Each of these two sets was alternatively used either as a training set or test set.

The quality of each individual is measured by the predictive performance obtained during training. It is also very important to set the optimal values of the following parameters to reduce the local optimality. The parameters are described as follows:

Population size: The size of the population denoted as |P|=50 is fixed for all the datasets.

Length of the individuals is fixed to $n$, where $n$ is the number of input features. The probability for crossover is 0.7 and mutation is 0.02. The number of iterations is 1000 for all the datasets.

### 4.2 Comparative Performance

The predictive performance obtained from HFLANN for the above datasets were compared with the results obtained from FLANN with back propagation learning and radial basis function network (RBF). Table 2 summarizes the average training and test performances of HFLANN and compared with FLANN and RBF.

**Table 2.** Average Comparative Performance of HFLANN, FLANN, and RBF

| Dataset | Algorithms | | | | | |
|---|---|---|---|---|---|---|
| | HFLANN | | FLANN | | RBF | |
| | Training | Testing | Training | Testing | Training | Testing |
| IRIS | 98.0001 | 97.3335 | 96.6665 | 96.6665 | 38.5000 | 38.5000 |
| WINE | 99.4380 | 90.4495 | 97.1910 | 88.7640 | 85.3935 | 79.2130 |
| PIMA | 80.7290 | 72.1355 | 79.5570 | 72.1355 | 77.4740 | 76.0415 |
| BUPA | 77.6820 | 69.2785 | 77.9725 | 69.2800 | 71.0125 | 66.9530 |
| ECOLI | 55.1670 | 50.8020 | 49.9625 | 47.3075 | 31.1780 | 26.1100 |
| GLASS | 63.5565 | 51.5075 | 60.7510 | 50.3800 | 48.9865 | 34.6440 |
| VOWEL | 40.4395 | 38.1965 | 27.9250 | 24.7220 | 25.2555 | 24.3250 |
| HOUSING | 82.2130 | 72.5295 | 76.4825 | 69.7630 | 67.1940 | 65.4150 |
| LED7 | 30.8110 | 27.5280 | 22.4185 | 19.7000 | 20.2820 | 16.5720 |
| LYMPHO. | 97.2970 | 77.0270 | 91.8920 | 74.3245 | 85.1350 | 72.2927 |
| ZOO | 99.0385 | 86.1850 | 97.1155 | 85.1645 | 96.1540 | 81.0830 |

From Table 2, one can easily verify that except BUPA case in all other cases on an average the proposed method is giving promising results in both training and test cases. In the case of BUPA, FLANN is performing better. Figure 1 shows the percentage of feature selected by the HFLANN, which is very important in the context of comprehensibility. The X-axis represents the datasets and Y-axis represents the percentage of active bits in the optimal chromosome obtained during the training.

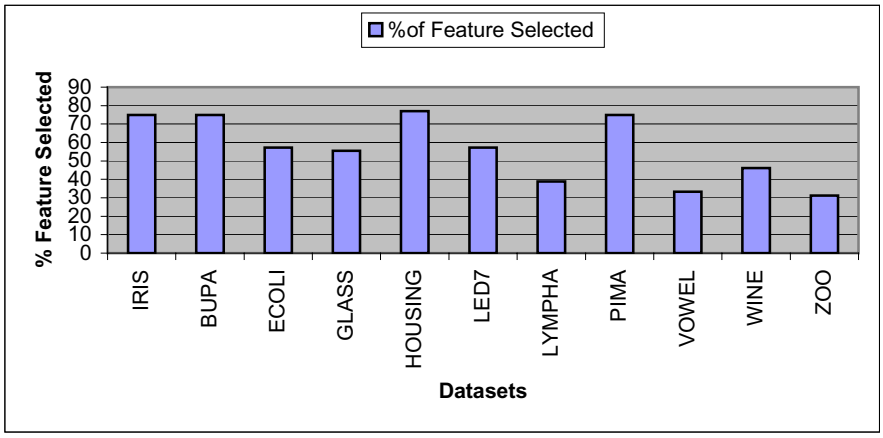


**Fig. 2.** Percentage of Optimal Set of Selected Features

### 4.3 Knowledge Incorporation in Predictive Accuracy

Let $n$ be the total number of features in the dataset; $T_1$ and $T_2$ denote the number of feature selected using the training set 1 and testing set 2 alternatively.

Notations and their Meaning: $|N|$ represent the total number of features in the dataset, $|T_1|$ denote the total number of selected features in test set 2, $|T_2|$ denote the total number of selected features in test set 1.

The fitness of the chromosome with respect to $T_1$ is

$$f(T_1) = \frac{|PA|\times|N|-\tau\times|T_1|}{|N|}. \tag{5}$$

Similarly the fitness of the chromosome with respect to $T_2$ is

$$f(T_2) = \frac{|PA|\times|N|-\tau\times|T_2|}{|N|} \tag{6}$$

where $|PA|$ represent the predictive accuracy and $\tau$ represent the tradeoff between two criteria and its value is 0.01.

**Table 3.** Predictive Accuracy of HFLAN by Knowledge Incorporation with $\tau$ =0.01

| Dataset | $N$ | P.A. Test Set 1 Chromosome | P.A. Test Set 2 Chromosome |
|---|---|---|---|
| IRIS | 4 | 95.9925 | 97.3260 |
| WINE | 13 | 89.8826 | 91.0064 |
| PIMA | 8 | 71.6125 | 72.6548 |
| BUPA | 6 | 70.3343 | 68.2013 |
| ECOLI | 7 | 54.6939 | 46.8973 |
| GLASS | 9 | 57.1374 | 45.8642 |
| VOWEL | 3 | 34.5063 | 41.8733 |
| HOUSING | 13 | 67.9771 | 77.0673 |
| LED7 | 7 | 19.6551 | 35.3923 |
| LYMPH | 18 | 78.3724 | 75.6721 |
| ZOO | 16 | 87.7494 | 84.6119 |

Table 3 shows the predictive accuracy using equation (5) and (6) of the HFLANN by incorporating a kind of knowledge of each chromosome optimally selected with respect to test set 1 and test set 2.

## 5 Conclusions

In this paper, we have evaluated the HFLANN for the task of classification in data mining by giving an equal importance to the selection of optimal set of features. The HFLANN model functionally maps the selected set of feature from lower to higher dimension. The experimental studies demonstrated that the classification performance of HFLANN model is promising. In almost all cases, the results obtained with the HFLANN proved to be better than the best results found by its competitor like RBF and FLANN with back propagation learning. The architectural complexity is low, whereas training time is little bit costly as compared to FLANN. As we know one of the most important criterions of data mining is how comprehensible the model is? If the architectural complexity increases than the comprehensibility decreases. Therefore, from this aspect we can claim that the proposed model can fit in data mining task of classification.

**Acknowledgments.** The authors would like to thank Department of Science and Technology, Govt. of INDIA and BK21 research program on Next Generation Mobile Software at Yonsei University, SOUTH KOREA for their financial support.

## References

1. Kriegel, H.-P., et al.: Future Trends in Data Mining. Data Mining and Knowledge Discovery 15(1), 87–97 (2007)
2. Ghosh, A., Dehuri, S., Ghosh, S.: Multi-objective Evolutionary Algorithms for Knowledge Discovery from Databases. Springer, Heidelberg (2008)
3. Misra, B.B., Dehuri, S.: Functional Link Artificial Neural Network for Classification Task in Data Mining. Journal of Computer Science 3(12), 948–955 (2007)
4. Oh, I.-S., et al.: Hybrid Genetic Algorithms for Feature Selection. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(11), 1424–1437 (2004)
5. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Morgan Kaufmann, San Francisco (1989)
6. Zhang, G.P.: Neural Networks for Classification: A Survey. IEEE Transactions on Systems, Man, Cybernetics-Part C: Application and Reviews 30(4), 451–461 (2000)
7. Pao, Y.–H., et al.: Neural-net Computing and Intelligent Control Systems. International Journal of Control 56(2), 263–289 (1992)
8. Blake, C.L., Merz, C.J.: UCI Repository of Machine Learning Databases, http://www.ics.uci.edu/~mlearn/MLRepository.html
9. Pal, S.K., Majumdar, D.D.: Fuzzy Sets and Decision Making Approaches in Vowel and Speaker Recognition. IEEE Transactions on Systems, Man, and Cybernetics 7, 625–629 (1977)