Diagnosis of Breast Cancer by Modular Evolutionary Neural Networks

Rahul Kala
rahulkalaiiitm@yahoo.co.in
Soft Computing and Expert System Laboratory,
Indian Institute of Information Technology and Management Gwalior
Morena Link Road, Gwalior, Madhya Pradesh-474010, India

R. R. Janghel
rrj.iiitm@gmail.com
Soft Computing and Expert System Laboratory,
Indian Institute of Information Technology and Management Gwalior
Morena Link Road, Gwalior, Madhya Pradesh-474010, India

Ritu Tiwari
rt_twr@yahoo.co.in
Soft Computing and Expert System Laboratory,
Indian Institute of Information Technology and Management Gwalior
Morena Link Road, Gwalior, Madhya Pradesh-474010, India

Anupam Shukla
dranupamshukla@gmail.com
Soft Computing and Expert System Laboratory,
Indian Institute of Information Technology and Management Gwalior
Morena Link Road, Gwalior, Madhya Pradesh-474010, India

Citation: R. Kala, R. R. Janghel, R. Tiwari, A. Shukla (2011) Diagnosis of Breast Cancer by Modular Evolutionary Neural Networks, *International Journal of Biomedical Engineering and Technology*, 7(2): 194 – 211.

Final Version Available At:
http://www.inderscience.com/info/inarticle.php?artid=43179

Abstract

Machine learning and pattern recognition play a vital role in the field of biomedical engineering, where the task is to identify or classify a disease based on a set of observations. The inability of a single method to effectively solve the problem gives rise to the use multiple models for solving the same problem in a 'Mixture of Experts' mode. Further the data may be too large for any system to effectively solve the problem. This motivates the use of computational modularity in the system where a number of modules independently solve part of the problem. In this paper we construct a Mixture of Experts model where a number of different techniques are applied to solve the same problem. The individual decision by each of these experts is fused by an integrator that gives the final output. Each of the units is a complex modular neural network. The first modularity clusters the entire input space into a set of modules. The second modularity divides the number of attributes. Each cluster is a neural network that solves the problem. The individual neural networks are evolved using Genetic Algorithms which optimizes both the architecture and the parameters. The complete system is used for the diagnosis of Breast Cancer. Experimental results show that the proposed system outperforms the traditional simple and hybrid approaches. The system on the whole is highly scalable to both number of attributes and data items.

Keywords: Modular Neural Networks, Ensembles, Evolutionary Neural Networks, Breast Cancer, Biomedical Engineering, Hybrid Computing, Artificial Neural Networks, Evolutionary Algorithms, Soft Computing.

1. Introduction

Biomedical Engineering deals with a unique amalgamation of the medical technologies with the principles of engineering to enable the creation of systems for problem solving (Bronzino, 2006). Medical diagnosis is a promising domain where we use intelligent automated systems to enable detection of disease. This detection may be used to assist doctors in more efficient and fast decision making. Many times the doctors may not be able to observe the symptoms that may be easily processed and diagnosed by the intelligent systems. These systems hence find widespread use in different diseases and different platforms.

Machine Learning deals with the task of finding of the patterns and trends in the historical data and storing them in a more compact and summered format (Shukla, Tiwari, and Kala, 2010). This enables the system use the same rules for the computation of output to any of the applied input. The outputs would be correct for both the training and the testing data sets, if the extracted rules are general enough to be extended to the testing data set. In this manner we may easily use machine learning techniques in the past medical record of the patients to precisely diagnose or predict the disease possibility in the new patients.

Most of the problems in medical diagnosis are classificatory in nature. Here we are supposed to classify the applied input into one of the more classes that the system possesses. This differs from the functional prediction problems where the output is more continuous in nature denoting some output value to any of the applied inputs. The classification problems mainly deal with the determination of the decision boundaries that separate the different classes from one another in the input space. This separation of the various classes is carried out using the knowledge base of the system that is developed after learning.

Breast Cancer is a prominent disease into the females that has been of special concern due to its widespread attacks in various parts of the world in the previous years. Cancer refers to cells that grow larger than 2mm in every 3 months and multiply out of control and spreads to other parts of the body. Breast cancer is the second leading cause of cancer deaths in women in worldwide and occurring in nearly one out of eight women. It occurs mostly in women. The detection of breast cancer mostly makes use of three techniques namely, Mammography, Biopsy and Fine Needle aspiration (Breastcencer.org, 2010).

The intelligent systems make extensive use of neural networks for classification and machine learning. The neural networks are an inspiration from the human brain that contains 10^{11} neurons and 10^{22} connections that all operate together for a massive amount of computation resulting in the human capability to solve really complex tasks. The artificial counterpart of the same, consisting of the artificial neural networks, is employed to solve much simpler problems with much limited number of neurons. These systems are effectively able to give high performances with much limited number of neurons (Konar, 2000).

The artificial neural networks face two major problems namely high dimensionality and high data set size. High dimensionality means having too many attributes into the system. The high number of attributes normally leads to a very high complexity with which the inputs map to the outputs. The task of classification deals with determination of decision boundaries across the individual classes. A high dimensionality means a very complex and highly dimensional shape of the input space which may not be easily worked out for the construction of these decision boundaries. The same is the problem when the number of data items becomes very large. This means the existence of very complex relations by which the inputs get mapped to the outputs. The intelligent system being used may not be able to formulate or imitate this complex relationship.

As a result of these problems, the use of a single system may not be able to solve the problem, giving a high degree of accuracy. Here we make use of two common techniques for problem solving. The first technique is the modular neural networks. These systems try to solve the problem by introducing modularity in the computation. The different parts or modules of the model perform different activities. They all solve some or the other part of the problem. Each module is a neural network in itself. The collective task by all the modules solves the complete problems (Matera, 1998).

Ensembles are another commonly used technique for problem solving. Here a number of systems solve the same problem redundantly (Hansen and Salamon, 1990). They work over the same inputs and using their own mechanisms, they generate the outputs. The different modules may generate different outputs. An integrator is made, that collects all these outputs and makes the final decision regarding the correct output of the system, considering the various responses generated by the different modules. Each module here is a neural network in itself. This model is also known as a mixture of experts model. The various modules act as experts that possess enough expertise to themselves solve the problem. The task of the integrator is to collect the decisions of these experts and make the final decision regarding the output of the system.

In this paper we simultaneously solve both the problems of high dimensionality as well as the high input data size. The complete approach is based on a modular neural network architecture. The first modularity is applied at the input space. Here we designate different modules for the different parts of the input space. Each module of the network then further applies the modularity at the level of the dimensionality. The different modules are given different attributes. Evolutionary neural networks are used as the basic constituting modules. Finally the entire architecture is mapped on a mixture of experts architecture. There are three different experts that solve the problem using their own mechanisms. Each of the experts is a complex modular neural network using the above architecture. This completes the entire model.

In this paper we first present the literature survey in section 2. We then use a bottom up approach to tackle the entire problem. We would first build an evolutionary neural network in section 3. Later we develop a model that divides the input attributes into various modules. Each of these modules would be an evolutionary neural network. This is presented in section 4. We then develop a model that divides the entire input space into clusters. Each of the modules of this layer would be a modular neural network as discussed in section 5. This would then be upgraded into a mixture of experts architecture. This is discussed in section 6. Section 7 presents the experimental results. Some discussions are presented in section 8. Finally we give the conclusion remarks in section 9.

2. Literature Review

This paper is an extension of much of the earlier works by the authors where the authors used different models on different diseases and different databases. The motivation is to unite all the different approaches to make a unanimous model that has the advantages of all the different approaches. Kala, Shukla, and Tiwari (2009a) made an implementation of a Modular Neural Network for machine learning. This model clustered the entire input space into clusters. Each cluster was solved using its own neural network. The approach was applied for learning of a self made database of face recognition. Results proved that the approach could better identify the faces. Further the system was scalable to handle much more data.

In another approach Shukla et al. (2009a) made use of ensemble approach for problem solving. Here a variety of models were used for solving the same set of inputs and outputs. The integrator used a voting mechanism for deciding the final output. This approach was on a speech database. The combination of face and speech was applied along with a better integration technique in (Kala et al., 2010). Here each module returned the probabilities of the occurrence of the various classes. These were summed up for all the modules to get the final probability vector. The integrator declared the class corresponding to the maximum sum as the final output class.

Numerous models on medical diagnosis have further been developed and tested (Janghel, Shukla, and Tiwari, 2010; Janghel et al., 2009; Shukla, Tiwari, and Kaur, 2009; Shukla et al., 2009b). These models use a variety of methods namely Multi-Layer Perceptron with Back Propagation Algorithm, Radial Basis Function Networks, Self Organizing Maps, Learning Vector Quantization, Adaptive Neuro Fuzzy Inference Systems, etc. for the diagnosis of diseases. The major diseases include diabetes, heart diseases, epilepsy, breast cancer, thyroid, etc. In all combinations of model and disease, an effective diagnosis could be made. This emphasizes on a high degree of accuracies of the individual systems. There is however always a scope to remove the individual limitations of the models and further enhance the recognition score.

A number of models from hybrid soft computing have also been applied on the problem of PIMA Indian Diabetes (Kala, Shukla, and Tiwari, 2009b). This includes the ensemble approach, neuro-fuzzy systems, and evolutionary neural networks. All the hybrid methods gave a good accuracy for diagnosis. Based on the comparisons in the same work it was clear that the evolutionary neural networks and ensemble techniques remove the limitations existing in the individual neural network models. An extended version of these works may be found in various chapters of the books (Shukla and Tiwari, 2011a, 2011b).

The problem of classification especially requires a good system modelling in order to enable the system separate the various classes in the system. The major problem is especially the classification of the inputs that lie close to the decision boundaries. Kala, Shukla, and Tiwari (2010a) also modified the neuro-fuzzy architecture to enable it carry effective classification. The approach was further extended to evolve the entire system using an evolutionary architecture (Kala, Shukla, and Tiwari, 2010b).

A lot of interesting work may be seen in the works of other authors into the mixed domain of machine learning and bio-medical engineering. One good modular neural network model is presented for the biometric recognition in the work of Melin and Castilo (2005), and Melin et al. (2006). Here the authors make three different modules for a multi-modal biometric recognition system. One module is dedicated to each biometric modality i.e. face, speech and fingerprint. Fuzzy integration is the integration technique of use. Each module in turn uses a hierarchical modular neural network with an evolutionary base and a fuzzy integration technique.

In another work Pedrajas, Martinez, and Perez (2002) used co-evolution as a mechanism of evolution of a modular neural network. In this model the various modules of the modular neural network evolve in a co-evolutionary approach. The various modules help each other to evolve with good recognition rate and develop distinct characteristics for an optimal overall system performance.

A large amount of work is also done at the biomedical front to develop innovative means for effective diagnosis. Ma et al. (2006) proposed a technology to extract micro-calcifications clusters with accurate edge effects to obtain much more hidden information which can't be detected by the naked eye on mammograms in order to help the doctors in diagnosing early breast cancer. Another approach makes use of fuzzy logic, vibro-acoustography and probabilistic neural network on mammograms for computerized microcalcification detection for breast cancer (Cheng, Lui, and Freimanis, 1998; Alizad et al. 2004; Karahaliou et al. 2008).

An effective diagnosis can also be done using image analysis of the screening mammograms of the breast cancer. The important task here is to extract the suitable features and use them for the diagnosis. This helps in the early detection of breast cancer (Sameti et al. 2009). In another work Abdalla et al. (2007) use various feature extraction tools like LDA, NDA, PCA and the classification is done by Support vector Machines (SVM) and neural networks. SVM is able to achieve better classification accuracy. Thermogram is a promising front-line screening tool as it is able to warn women of breast cancer up to 10 years in advance (Tan et al., 2007). The use of modified self-organizing map with nonlinear weight adjustments to reduce number of unnecessary biopsies can be found in the work of Laufer and Rubinsky (2009). Probabilistic neural network to perform supervised classification and rough sets is able to reduce the number of attributes in the dataset without sacrificing classification accuracy as reported in the work of Revett et al. (2005).

For all the experimentation in this paper we would be making use of the database from UCI Machine Learning Repository (Wolberg, Mangasarian, and Aha, 1992). The problem is the diagnosis of the diseases and detection of the type of breast cancer i.e. Malignant or Benign. This is based on some attributes that are given as inputs in the database. This database consists of 29 real valued inputs. These correspond to the following features for each cell nucleus: radius (mean of distances from center to points on the perimeter), texture (standard deviation of gray-scale values), perimeter, area, smoothness (local variation in radius lengths), compactness (perimeter^2 / area - 1.0), concavity (severity of concave portions of the contour), concave points (number of concave portions of the contour), symmetry, fractal dimension (coastline approximation - 1). The entire data set consists of a total of 357 benign and 212 malignant cases, totalling to 569 instances in the database.

3. Evolutionary Neural Network

The first step in the development of the complete system is the use of evolutionary neural network. The neural network model we use is a Multi-Layer Perceptron. In this model a number of artificial neurons are connected to each other in a layered architecture. The first layer is the input layer and the last layer is the output layer. There may be any number of hidden layers in between. Most of the problems can be effectively solved by a single layer alone. On the other hand increasing the layers or the number of neurons makes the output surface very complex, giving poor performance to the testing outputs. Hence we assume that the network would have only a single hidden layer. The task of the evolutionary algorithm here is to both optimize the neural network architecture as well as fix the correct weights and biases. The complete framework of evolutionary neural networks is shown in figure 1.

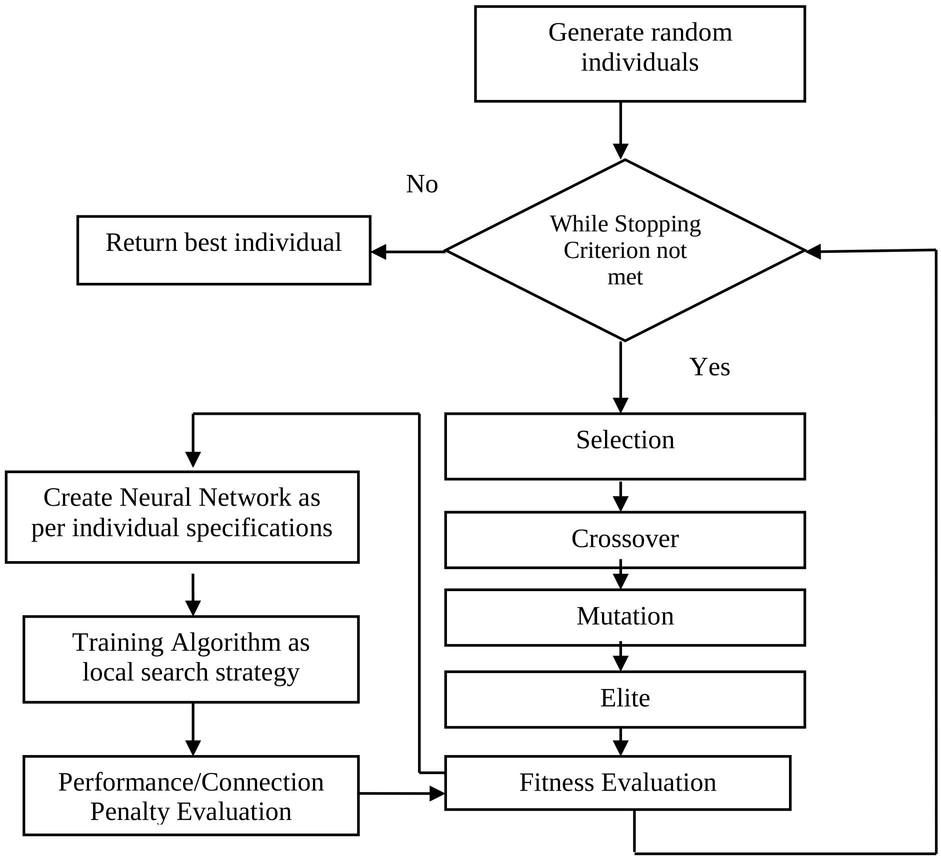

Fig. 1: The Evolutionary Neural Network

3.1 Individual Representation

The first task in the implementation of the evolutionary neural network is the individual representation. Each individual of the genetic algorithm is a neural network. We consider that the maximum number of hidden neurons in the only hidden layer of the neural network to be h_{max}. Each neuron may or may not be connected to any of the input and the output neurons. In this manner the neural network follows a connectionist approach where the connections between the various neurons may or may not exist. This results in a variable architecture to the entire neural network. A hidden neuron not connected to any input neuron or output neuron is equivalent to be absent from the entire network.

The genetic individual contains a set of genes. Each gene stores some information about the network. Let there be a total of i input neuron and o output neurons. The first $2ih_{max}$ genes of the individual contain information about the existence of the connection and its weights value between the input layer and hidden layer. There are in forms of pairs (c_{ab}, w_{ab}) for all the $i\ h_{max}$ connections between the input and hidden layer. Here c_{ab} denotes the presence or absence of the connection between the input node a and hidden layer node b. Similarly w_{ab} denotes the presence or absence of the connection between the input node a and hidden layer node b.

The next $2h_{max}o$ genes of the individual contain information about the existence of the connection and its weights value between the hidden layer and output layer. There are in the same format of connection followed by weight or (c_{ab}, w_{ab}) for all the $h_{max}o$ connections between the hidden layer and output layer.

The last section of the genetic individual consists of the biases of the various neurons in the hidden layer as well as the output layer. This makes the last $h_{max}+o$ genes of the individual.

3.2 Genetic Operators

The genetic evolution tries to find the optimal values of the various genes. In other words it tries to find whether a connection must exist between a pair of nodes, and if it must exist, what must be its value. We use all conventional operators for the same. Rank based selection with stochastic uniform selection, scattered crossover, Gaussian mutation and a small elite count is used.

3.3 Fitness Function

The fitness function is a measure of the accuracy by which the neural network solves the problem out of the training input data set that is given to it. We assume that the data had already been divided into the training and testing data sets. The fitness function first initializes the neural network as per the specifications of the neural network stored in the individual. This network is then trained using Back Propagation Algorithm. This serves as a local search technique. The neural network optimization may have a very complex fitness landscape. It is hence important to supplement it with an appropriate local search technique that returns the optimal point in the surroundings of the individual. Both the learning rate as well as the momentum may hence be kept low.

The variable architecture neural networks trained on the training data set only may have the intention to grow up in size indefinitely. This is because of the fact that large sized neural networks give a very high performance on the training data and a poor performance on the testing data. The purpose is to enhance the performance of the system on testing data which requires small sized networks. Hence the total number of connections is computed and a penalty is added to the fitness function directly proportional to the number of connections. The net fitness may be hence given by equation (1). We assuming the GA is invoked in such a manner so as to minimize equation (1).

$$Fit(I) = -Performance(I) + \alpha\ NC(I) \tag{1}$$

Here *Fit(I)* is the fitness function (which needs to be minimized),
Performance(I) is the performance measurement function measured as diagnosis percentages,
NC(I) is the function returning the total number of connections
α is the penalty constant

4. Division of Input Attributes

At the next level we divide the input attributes into a number of sets. Each set is given to one of the modules for the diagnosis and decision making. The module here is simply an evolutionary neural network that we discussed in section 3. There are a total of two modules. The entire set of attributes is divided in between these two modules. Any attribute may be given to any, or both the modules. It is usually preferred that every attribute be present in one or more modules.

In this problem we make a total of two modules. The first module is given the first half attributes. The second module is given the second half set of attributes. There are a total of 30 attributes. Hence the first 15 attributes are given to first module and the other 15 are given to the other module.

Previously we had divided the data set into inputs and outputs and then into training and testing data sets. The complete training and testing data set inputs are further divided into two parts. This forms a total of 2 training input data sets and 2 testing input data sets. Both sets have a common output set. The two modules are independently formed and trained using the two training data sets. This forms two independent modules to be used in the modular neural network.

Whenever training or testing input is given to the system, it is first broken down into two parts. The first part contains the first half of the attributes and the second half contains the other half attributes. These are given to the two neural networks or modules. Both the neural networks independently assess their part of the applied inputs and make some decisions regarding the output. These networks output the probability of the cancer being Benign (or not being Malignant). This probability lies between 0 and 1.

The last part to be performed is to make the integrator. The two modules independently assess their inputs and output the probability. We simply take the average of the probabilities produced by these two networks. This forms the resultant probability of the cancer being Benign. In case the probability is greater than 0.5, the output is Benign, else the output is Malignant. The system so far is shown in figure 2.

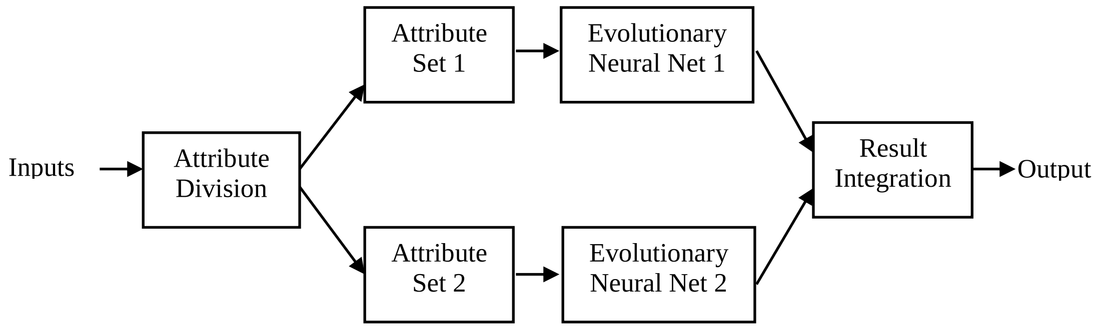

Fig. 2: Attribute Division in Modular Neural Network

5. Clustering of Input Space

In the next level we cluster the input space into clusters. Each cluster is an independent module that solves the problem by its own mechanisms. The clustering is done using the training data. A good means of clustering may be by the use of Fuzzy C Means clustering. This algorithm forms fuzzy clusters. Each data item is hence the member of the cluster by some degree and not a member by some other degree. We are primarily interested only in the cluster centers in this approach.

The cluster centers are an indicator of how the entire input space may get divided in a discrete and not fuzzy manner. Any point in the input space is said to be belonging to the cluster whose center is closest to it. Let the various cluster centers be $c_1, c_2, c_3, \ldots c_n$. Any input I belongs to the cluster j given by equation (2).

$$\text{Cluster}(I) = j : \| c_j - I \| < \|c_k - I\| \text{ for all } k \neq j \tag{2}$$

In this manner we may divide the entire training data into clusters. Each cluster of training data is used for training of an independent modular neural network. This modular neural network is the same network that was used in section 5. The number of clusters is fixed to 3 in the used approach. Each module of this network, representing a modular neural network, is independently trained with the training data belonging to that cluster.

Hence whenever any of the training or testing data is applied, the first job is to compute the cluster to which it belongs. The corresponding modular neural network is then invoked that carries forward the rest of the task of computation and deciding of the class to which the input classifies to. Since only one of the modules is invoked, there is no need of the making of an integrator. Only one of the available modules gives the output and the others are all passive.

The entire approach so far is given in figure 3.

6. Mixture of Experts

At the highest level in the system we build an ensemble. The ensembles are used to redundantly solve the problem by the various modules. Each of the modules takes the same inputs and processes them to give the output. The outputs are then combined using an integrator. The various modules give the probability of the occurrence of the various classes as their outputs. These are then averaged up to get the final probability sum. The class corresponding to the highest probability count is declared as the winner. This is the same approach that was discussed in section 4. It may be noted here that the final declaration of the result to which the input belongs to is done at this stage and not at the stage of attribute division in section 4.

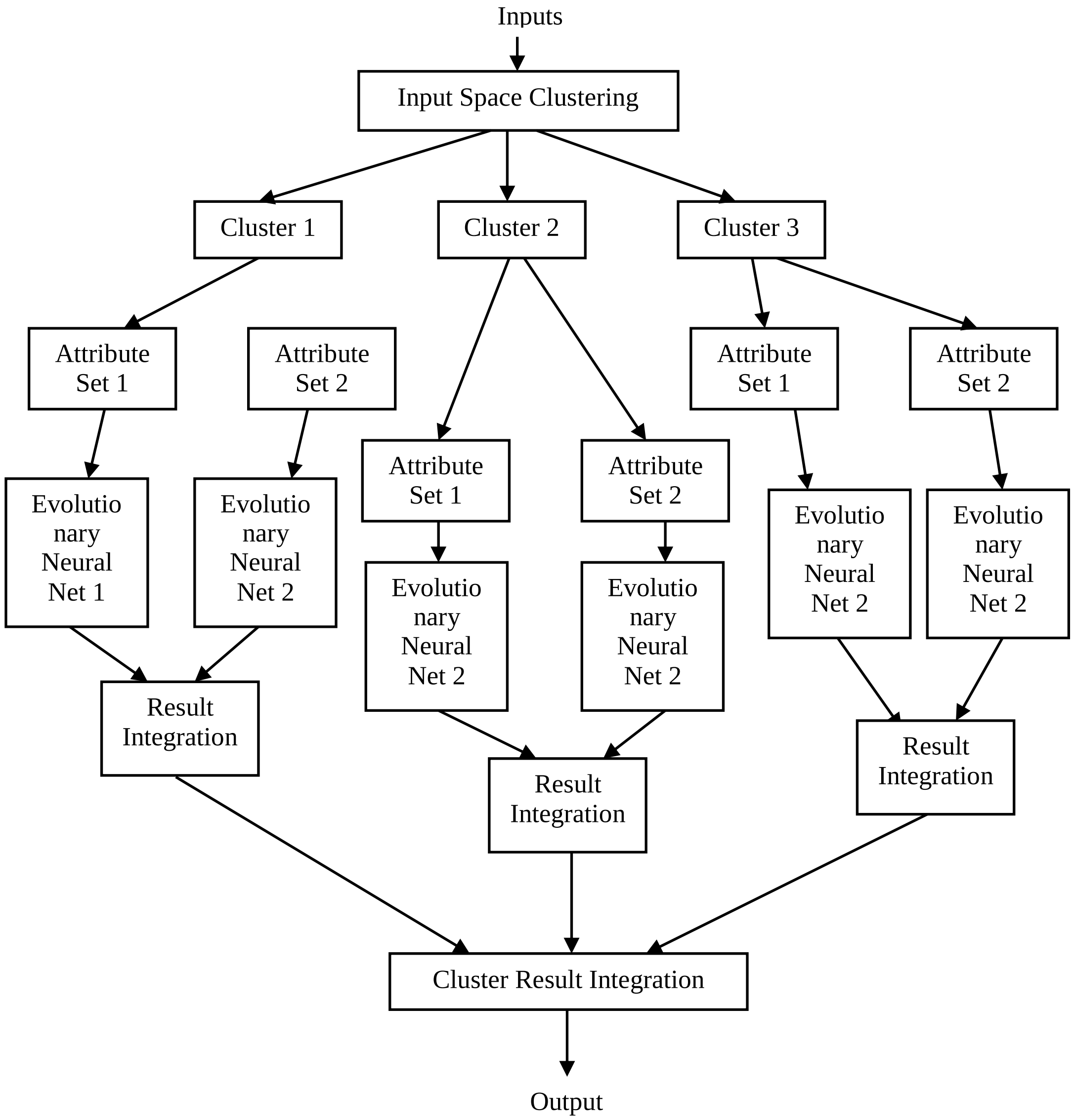

Fig. 3: Clustering of Input Space

Now whenever any input is given to the system, it is given to each of the modules. The modules are complex modules, each representing a modular neural network. Each module independently computes the output as a probability vector. The probability vectors from each of the modules are averaged and the final decision regarding the class to which the input belongs is made.

In this approach we make 3 modules. All the three modules are exactly similar in nature, except for the ground level neural network architecture. The first module used a Multi-Layer perceptron neural architecture. The second module uses Radial Basis Function Network neural network architecture. The last and the third module also uses the Multi-Layer perceptron neural architecture, but with a different architecture specifications. These are all tuned and evolved using genetic algorithm as discussed in section 3.

The final model is presented in figure 4.

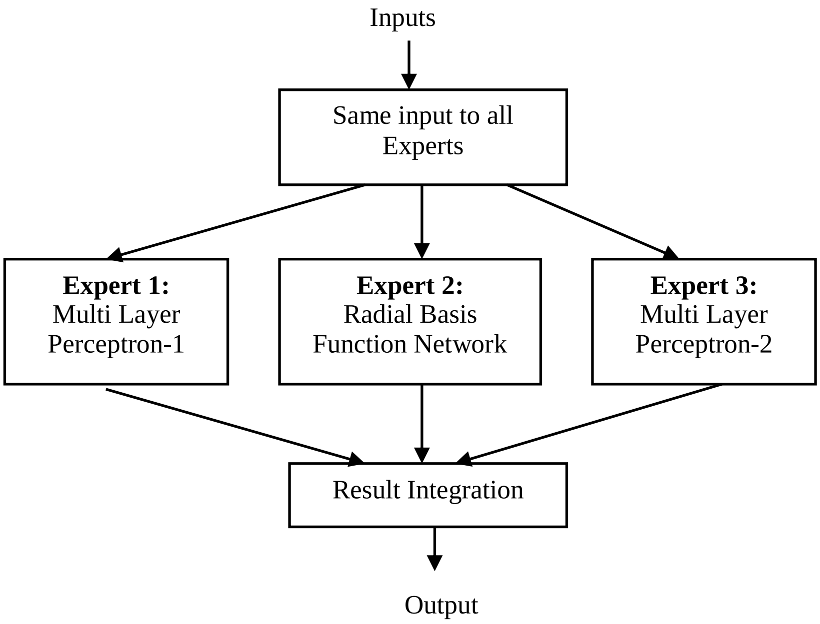

Fig. 4: Mixture of Experts

7. Results

The above approach was simulated on MATLAB using the Breast Cancer database from UCI Machine Learning Repository. The database was initially normalized and distributed into inputs and outputs. 70% of the database was used for training and 30% was used for testing. The entire dataset was randomly divided into the training and testing data sets and into the inputs and the outputs. This was done by a small code snippet.

At the highest level, three modules were to be built up that form the three complex modules of the modular neural network. Each module was developed independently without considering the other modules. Only when the independent modules were trained and tested, the task of integration of the results was done by the code written for this part of the hierarchy.

Each expert of this mixture of experts module was itself a complex modular neural network. For this we first clustered the data set consisting of the training data. This was done to make three clusters. The Fuzzy C Means clustering algorithm was used which is an inbuilt feature in MATLAB. The cluster centers were used to distribute the entire training and testing data set into clusters. Each of the cluster hence got some training and testing data that was used for the formulation of the module of this complex modular neural network. Each module is in reality a modular neural network.

The attributes of the entire training and testing data set was divided into two parts. The first part contained half the input attributes and the second part contained the other half of the input attributes. In this manner each of the complex module further got partitioned into two separate modules. Each of these modules was an evolutionary neural network. The model of the neural network depended upon the expert being made. This could be a Multi-Layer Perceptron or a Radial Basis Function Network.

At the core level, all the modules were evolutionary neural networks. We had intended to make 3 experts. Each of the experts had the data clustered into 3 clusters. This made a total of 3 x 3 modules. The modules were further partitioned by division of the input space. Hence the total number of evolutionary neural networks to be formed was 3 x 3 x 2 or 18. It is evident that independently evolving 18 neural networks, each with its own training data and model would have been a tedious task requiring a large amount of patience. We hence fixed the genetic parameters to be the same for all the networks. This was motivated from the face that all the networks were being trained on fragments from the same database. Hence the complexities of the various networks could be assumed to be similar and a single parameter set specified on the best practices guidelines may be optimal enough. Further the various networks in the same expert had the same network specifications, in case there was any required to be specified to the system. The various training data sets were given logical names at the time of division, mentioning their cluster numbers, module numbers and expert's names. This made it easy to reputedly call the evolutionary process for the different networks. The evolution was slightly different for the three experts, because of the difference in the models. However the evolution was exactly similar for the various networks within the same expert. There was only a difference in the training data set.

The first expert was Multi-Layer Perceptron, the maximum number of neurons h_{max} could be 20. The activation functions of the hidden and output layers were *tansig* and *logsig* respectively. The training algorithm was *traindg*. Back Propagation Algorithm was used as a local search strategy. The learning rate was 0.05 and momentum was fixed to be 0.3. The network was trained for 30 epochs with a goal of 10^{-2}. The various weights and biases could vary from -2 to 2. Further the crossover was fixed as 0.7 and Gaussian mutation had a scale and shrink of 1 each. Elite count was kept as 2. There were a total of 100 individuals and 50 generations. The connection penalty was fined as 0.01. The second expert was the Radial Basis function network. Here only the network parameters were optimized by GA and these could vary from 0 to 1. The third expert was also Multi-Layer Perceptron. This had exactly the same parameters. However h_{max} was given a value of 35.

The entire testing on the training and the testing data was done by writing separate codes that invoked the needed neural networks as per their logical names. The results were integrated by another complex integration mechanism as per the problem logic. This formed another part of the entire code.

The system gave a performance of 98.5075% on the training data and 95.8084% on the testing data. It may be noted that the same division of data gave a performance of 94.4020% for the training data and 91.4773% on the testing data when a conventional modular neural network was used and the input was clustered into 3 clusters in the input space. Further the performance was 98.2188% on the training data and 94.8864% on the testing data on the use of ensembles.

In this simulation three ensembles were used, all with a multi-layer perceptron architecture and back propagation training algorithm. These differenced in the number of neurons in the hidden layer which were 14, 18 and 20 for the three modules. The performance with the use of evolutionary neural networks was 96.2779% on the training data and

95.7831% on the testing data. Here a variable architecture connectionist approach was used for the multi-layer perceptron model. The various performances are summarized in table 1. It may be easily seen that the proposed algorithm is better than all the conventional approaches used for the diagnosis.

Table 1: Comparison of performances of various approaches

S. No.	Method	Training Accuracy	Testing Accuracy
1.	**Proposed Algorithm**	**98.5075%**	**95.8084%**
2.	Modular Neural Network	94.4020%	91.4773%
3.	Ensembles	98.2188%	94.8864%
4.	Evolutionary Neural Network	96.2779%	95.7831%

It would be further important to realize the distribution of this accuracy. We hence independently study the various experts and the various modules to independently identify their performances. The first expert was the Multi-Layer Perceptron. This expert had a performance of 98.0100% on the training data and 95.8084% on the testing data. This expert was a combination of three clusters. The performance of each of the clusters was 98.9362%, 97.5000%, and 96.8085% on the training data and 98.4848%, 92.0635%, and 97.3684% on the testing data. Each of the three clusters had two modules. We hence further analyze the performance of both these modules. For the 1st cluster the performance was 98.9362% for the training data and 95.4545% on the testing data for module 1. The same performance was 99.4681% and 98.4848% for module 2. The performance of the second cluster had 93.3333% and, 84.1270% as the performances of the first module; and 97.5000% and 93.6508% as the performance of the second module. For the third cluster the performances were 96.8085% and 97.3684% for 1st module and 100% and 96.8085% for the second module. The other experts had similar performance measures. The performance of the various modules is given in table 2.

Table 2: The individual performances of the various modules in the algorithm

Code	Expert Number	Cluster Number	Module Number	Training Accuracy	Testing Accuracy
A	**Entire System (all experts combined)**			**98.5075%**	**95.8084%**
E1	**Expert 1: Multi-Layer Perceptron (all clusters combined)**			**98.0100%**	**95.8084%**
E1.C1	*Multi-Layer Perceptron-1*	*1 (all modules combined)*		*98.9362%*	*98.4848%*
E1.C1.M1	Multi-Layer	1	1	98.9362%	95.4545%

	Perceptron-1				
E1.C1.M2	Multi-Layer Perceptron-1	1	2	99.4681%	98.4848%
E1.C2	*Multi-Layer Perceptron-1*	*2 (all modules combined)*		*97.5000%*	*92.0635%*
E1.C2.M1	Multi-Layer Perceptron-1	2	1	93.3333%	84.1270%
E1.C2.M2	Multi-Layer Perceptron-1	2	2	97.5000%	93.6508%
E1.C3	*Multi-Layer Perceptron-1*	*3 (all modules combined)*		*96.8085%*	*97.3684%*
E1.C1.M1	Multi-Layer Perceptron-1	3	1	96.8085%	97.3684%
E1.C1.M2	Multi-Layer Perceptron-1	3	2	100%	96.8085%
E2	**Expert 2: Radial Basis Function Network**			**98.0100%**	**95.8084%**
E2.C1	*Radial Basis Function*	*1 (all modules combined)*		*98.9362%*	*98.4848%*
E2.C1.M1	Radial Basis Function	1	1	97.8723%	93.9394%
E2.C1.M2	Radial Basis Function	1	2	99.4681%	96.9697%
E2.C2	*Radial Basis Function*	*2 (all modules combined)*		*96.6667%*	*93.6508%*
E2.C2.M1	Radial Basis Function	2	1	94.1667%	88.8889%
E2.C2.M2	Radial Basis Function	2	2	96.6667%	88.8889%
E2.C3	*Radial Basis Function*	*3 (all modules combined)*		*100%*	*97.3684%*
E2.C1.M1	Radial Basis Function	3	1	98.9362	97.3684%
E2.C1.M2	Radial Basis Function	3	2	100%	97.3684%
E3	**Expert 1: Multi-Layer Perceptron (all clusters combined)**			**98.7562%**	**95.8084%**
E3.C1	*Multi-Layer Perceptron-2*	*1 (all modules combined)*		*98.9362%*	*98.4848%*
E3.C1.M1	Multi-Layer Perceptron-2	1	1	98.9362%	95.4545%
E3.C1.M2	Multi-Layer Perceptron-2	1	2	98.9362%	98.4848%
E3.C2	*Multi-Layer Perceptron-2*	*2 (all modules combined)*		*98.3333%*	*90.4762%*
E3.C2.M1	Multi-Layer Perceptron-2	2	1	95.0000%	85.7143%
E3.C2.M2	Multi-Layer Perceptron-2	2	2	99.1667%	87.3016%
E3.C3	*Multi-Layer*	*3 (all modules*		*98.9362%*	*94.7368%*

	Perceptron-2	*combined)*			
E3.C1.M1	Multi-Layer Perceptron-2	3	1	100.0000%	97.3684%
E3.C1.M2	Multi-Layer Perceptron-2	3	2	98.9362%	97.3684%

8. Discussion

There is an extensive use of neural networks to solve numerous real life problems. These problems span across multiple domains and hence have different type of characteristics and traits. It is well known that the more complex a problem is the more complex the architecture of the system solving the problem becomes. Here complexity is the difficulty in solving, rather than the perceived complexity. The ultimate example is of the human brain that can do all the wonderful tasks, which are all highly complex problems. But this is possible due to the highly complex network architecture inside the human brain. Many times the problems may not be hard, but the excessive demand for high accuracy may make them hard to solve. This is the case with most of the bio-medical problems, where we attempt to solve problems in a manner different from the manner adopted by the doctors. With this we attempt to get additional functionalities and solve problems that may not be conventionally possible.

This paper is represents an attempt towards solving problems which in multiple ways represent a high complexity. The aim is to build a general framework that can be adjusted or tuned in a way so as to completely solve any problem, giving the best possible accuracy score. In this manner we aim in making the presently employed diagnostic systems more scalable to a large set of problems. Excessive increase in computation displays hope of having very large collection of biomedical databases that would store very large information in them. Excessively large databases are further sources of very valuable pieces of hidden information that may not be common. With this framework we further attempt to make systems diverse enough to capture these trends. Experimental results fully support that such a framework can be of use, displaying a very high recognition rate as compared to all other systems. Hence we may hope that trends to most of the other diseases may be similar in nature, where the complexity is high. The increase in accuracy using the proposed algorithm is also an indication towards capture of some more valuable diagnostic parameters by the intelligent system, based on the supplied parameters.

9. Conclusions

Scalability and robustness are important aspects of any system to be used in a real life application. It is important for whatever systems we develop to be able to digest large amounts of data. Highest degree of diagnosis accuracy is needed in the medical domain. This is especially important for the ethical issues confronting to the treatment of the humans by machines. Even a single wrong decision by the machine may lead to a fatal loss to the human which would further have large impact in the non-acceptance of these systems. Even though a large amount of research has been done in the development of various tools and techniques for the medical diagnosis, the performance is still not cent percent. One of the tasks that is done for the increase of accuracy is to pool the database with more attributes and more data. This may be due to the lack of expertise to work over better attributes, or the impossibility of the same. The larger amount of data requires the system to be more scalable. The training especially becomes very slow when there is a large amount of data and attributes. The attributes can naturally not be deleted as it would affect the recognition by the system. The large amount of attributes further makes the system more complex requiring more neurons and making learning and testing slow.

One of the most important characteristics of the proposed algorithm is its capability to better handle a large volume of data for the classificatory problems. In this characteristic it exceeds all the present modular approaches that can only handle modularity to some level. It is evident that if we divide the problem into too many modules, the problem becomes highly localized and the generality is completely lost. The resulting system can naturally not afford a large number of modules. Another task sometimes carried out is the division of the attributes. The attribute division would always demand every module to have enough attributes for a good overall performance. This again limits the amount of modularity. The discussed approach mixes both of these to attain a very large modularity level for highly complex problems with large data and large attributes. This all is built over a mixture of experts architecture for a more higher performance by redundant observations of the different experts.

The algorithm was tried over the Breast Cancer database. Using the approach we achieved a very high accuracy that exceeded all state-of-the-art methods used in literature. This clearly shows that the algorithm gains by the simultaneous introduction of the mixture of experts, modularity and evolutionary principles. All other methods making use of a single of such concept lagged at some or the other way. This makes the proposed algorithm exceed all other algorithms.

There are numerous limitations of the current work that may be worked upon into the future. The major limitation of the work is that all simulations are restricted to a single database of Breast Cancer database. This database is not very scalable in nature. Hence the experimentation needs to be done on different database with more attributes and data recordings. The difference between the different approaches would get magnified on these databases. Many methods may completely fail to do diagnosis because of the large data size. The validation may further be done of different data sets. Another limitation of the work is that it makes use of only the curve fitting neural models for the recognition. The approach may be extended to the classificatory models as well. Further the work is restricted to the use of a single integration technique. The usage of different integration techniques like polling, minimum, maximum, median, fuzzy integration etc. may provide different results and trends. The present approach uses random technique of division of attributes amongst the various modules. Both the number of type of attributes to be given to the various modules may be done by a computationally intelligent technique using some simple heuristics.

Complexity is a big problem in problems, which can never be predicted seeing any database. Adding too many hierarchies to the system would make the complete system very complex and would have its own disadvantages. Effective techniques hence need to be developed, that would automatically develop a system architecture that best uses the base techniques for problem solving. These techniques need to be as adaptable as possible, where the different parameters automatically attain their optimal values. This would be a big advantage over the present system that may be worked upon into the future.

Acknowledgement

This work has been supported and sponsored by Indian Institute of Information Technology and Management Gwalior.

References

[1] Abdalla , A. M., Deris. S., Zaki, N., Ghoneim, D. M. (2007) ‘Breast Cancer Detection Based on Statistical Textural Features Classification’, *proceedings of the 4th International Conference on Innovations in Information Technology*, pp 728-730.

[2] Alizad, A., Fatemi, M., Wold, L. E., and Greenleaf, J. F. (2004) 'Performance of Vibro-Acoustography in Detecting Microcalcifications in Excised Human Breast Tissue: A Study of 74 Tissue Samples', *IEEE transactions on medical imaging*, Vol. 23, No. 3, pp 307-312.
[3] Breastcencer.org (2010) *Understanding the Breast Cancer*, available at http://www.breastcancer.org [accessed Feb 2010]
[4] Bronzino, J. D. (2006) *Biomedical Engineering Fundamentals*, Boca Raton, FL: CRC Press
[5] Cheng, H. D., Lui, Y. M., and Freimanis, R. I. (1998) 'A Novel Approach to Microcalcification Detection Using Fuzzy Logic Technique', *IEEE transactions on medical imaging*, Vol. 17, No. 3, pp 442-450
[6] Hansen, L. K. and Salamon, P. (1990) 'Neural Network Ensembles', *IEEE Transaction on Pattern Analysis and Machine Learning*, Vol 12, No. 10, pp 993-1001
[7] Janghel, R. R., Shukla, A., and Tiwari, R. (2010) 'Decision Support system for fetal delivery using Soft Computing Techniques', *Proceedings of the Fourth International Conference on Computer Sciences and Convergence Information Technology*, *ieeexplore*, pp 1514-1519, Seoul, Korea.
[8] Janghel, R. R., Shukla, A., Tiwari, R., and Tiwari, P. (2009) 'Clinical Decision support system for fetal Delivery using Artificial Neural Network', *Proceedings of the 2009 International Conference on New Trends in Information and Service Science*, NISS 2009, pp 1070-1075, Gyeongju, Korea
[9] Kala, R., Shukla, A., and Tiwari, R. (2009a) 'Fuzzy Neuro Systems for Machine Learning for Large Data Sets', *Proceedings of the IEEE International Advance Computing Conference*, *ieeexplore*, pp 541-545, Patiala, India
[10] Kala, R., Shukla, A., Tiwari, R. (2009b) 'Comparative analysis of intelligent hybrid systems for detection of PIMA indian diabetes', *Proceedings of the IEEE 2009 World Congress on Nature & Biologically Inspired Computing*, NABIC '09, pp 947 - 952, Coimbatote, India
[11] Kala, R., Shukla, A., and Tiwari, R. (2010a) 'A Novel Approach to Classificatory problem using Neuro-Fuzzy Architecture', *International Journal of Systems, Control and Communications (IJSCC)*, Inderscience Publishers
[12] Kala, R., Shukla, A., and Tiwari, R. (2010b) 'A Novel Approach to Classificatory Problem using Grammatical Evolution based Hybrid Algorithm', *International Journal of Computer Applications*, Vol 1, No 28, pp 73-80
[13] Kala, R., Vazirani, H., Shukla, A., and Tiwari, , R. (2010) 'Fusion of Speech and Face by Enhanced Modular Neural Network', *Proceedings of*

the International Conference on Information Systems, Technology and Management, ICISTM 2010, CCIS 54, pp. 363-372, Bankok, Thailand
[14] Karahaliou, A. N. *et al.* (2008) ‘Breast Cancer Diagnosis: Analyzing Texture of Tissue Surrounding Microcalcifications’, *IEEE transactions on information technology in biomedicine*, vol. 12, no. 6, pp 731-738
[15] Konar, A. (2000) *Artificial Intelligence and Soft Computing: Behavioural and Cognitive Modelling of the Human Brain*, Boca Raton, FL: CRC Press
[16] Laufer, S. and Rubinsky, B (2009) ‘Tissue Characterization with an Electrical Spectroscopy SVM Classifier’, *IEEE transactions on biomedical engineering*, Vol. 56, No. 2, pp 525-528.
[17] Ma, Y. *et al* (2006) ‘Extracting Micro-calcification Clusters on Mammograms for Early Breast Cancer Detection’, *Proceedings of the 2006 IEEE International Conference on Information Acquisition,* pp 499-504, Weihai, Shandong, China
[18] Matera, F. (1998) ‘Modular Neural Networks’, *Substance Use and Misuse*, Vol. 33, No. 2, pp 307-315
[19] Melin, P. and Castillo, O. (2005) *Hybrid Intelligent Systems for Pattern Recognition Using Soft Computing*, Heidelberg: Springer
[20] Melin, P. *et al* (2006) ‘Modular Neural Networks and Fuzzy Sugeno Integral for Face and Fingerprint Recognition’, *In: Applied Soft Computing Technologies: The Challenge of Complexity,* Heidelberg: Springer
[21] Pedrajas, N. G., Martinez, C. H., and Perez, J. M. (2002) ‘Multi-objective cooperative coevolution of artificial neural networks (multi-objective cooperative networks)’, *Neural Networks*, Vol. 15, 2002, pp 1259–1278
[22] Revett, K., Gorunescu, F., Gorunescu, M., Darzi, E., Ene, M. (2005) ‘A Breast Cancer Diagnosis System: A Combined Approach Using Rough Sets and Probabilistic Neural Networks’, *Proceedings of the IEEE International Conference on Computer as a Tool, EUROCON 2005*, pp 1124-1127, Serbia & Montenegro, Belgrade.
[23] Sameti, M., Ward, R. K., Parkes, J. M., and Palcic, B. (2009) ‘Image Feature Extraction in the Last Screening Mammograms Prior to Detection of Breast Cancer’, *IEEE journal of selected topics in signal processing*, Vol. 3, No. 1, pp 46-52.
[24] Shukla, A. and Tiwari, R. (2011a) *Intelligent Medical technologies and Biomedical Engineering: Tools and Applications,* Hershley, PA: IGI Global Publishers
[25] Shukla, A. and Tiwari, R. (2011b) *Biomedical Engineering and Information Systems: Technologies, Tools and Applications*, Hershley, PA: IGI Global Publishers

[26] Shukla, A., Tiwari, R., and Kala, R. (2010) *Real Life Applications of Soft Computing*, Boca Raton, FL: CRC Press
[27] Shukla, A., Tiwari, R., Kaur, P. (2009) 'Intelligent System for the Diagnosis of Epilepsy', *Proceedings of the IEEE World Congress on Computer Science and Information Engineering (CSIE), ieeexplore*, pp 755-758, Los Angeles/Anaheim, USA
[28] Shukla, A., Tiwari, R., Meena, H. K., and Kala, R. (2009a) Speaker Identification using Wavelet Analysis and Modular Neural Networks, *Journal of Acoustic Society of India*, Vol 36, No. 1, pp 14-19
[29] Shukla, A., Tiwari, R., Kaur, P., and Janghel, , R. R. (2009b) 'Diagnosis of Thyroid Disorders using Artificial Neural Networks', *Proceedings of the IEEE International Advanced Computing Conference*, ieeexplore, pp 1016-1020, Patiala, India
[30] Tan, T. Z., Quek, C., Ng, G. S., and Ng, E. Y. K. (2007) 'A novel cognitive interpretation of breast cancer thermography with complementary learning fuzzy neural memory structure', *Expert Systems with Applications*, Vol. 33, pp 652–666.
[31] Wolberg, W. H., Mangasarian, O. L., and Aha, D. W. (1992) *UCI Machine Learning Repository* [http://www.ics.uci.edu/~mlearn/MLRepository.html], University of Wisconsin Hospitals.

About the Authors

Rahul Kala

Rahul Kala is an Integrated Post Graduate (BTech and MTech in Information Technology) student in Indian Institute of Information Technology and Management Gwalior. His areas of research are hybrid soft computing, robotic planning, biometrics, artificial intelligence, and soft computing. He has published about 35 papers in various international and national journals/conferences and is the author of 2 books. He also takes a keen interest toward free/open source software. He secured All India 8th position in Graduates Aptitude Test in Engineeging-2008 Examinations and is the winner of Lord of the Code Scholarship Contest organized by KReSIT, IIT Bombay and Red Hat.

R R Janghel

RR Janghel is a research scholar at Indian Institute of Information Technology and Management Gwalior. He did his BTech from Rungta college of Engineering and Technology, Bhiali (C.G) and MTech from National Institute of Technology, Raipur (C.G.). He secured first position in his post graduation from NIT Raipur. His areas of research include biomedical engineering, expert systems, neural networks, hybrid computing, and soft computing. He has numerous publications in various international journals and conferences.

Dr. Ritu Tiwari

Dr. Ritu Tiwari is serving as an Assistant Professor in Indian Institute of Information Technology and Management Gwalior. Her field of research includes Biometrics, Artificial Neural Networks, Speech Signal Processing, Robotics and Soft Computing. She has published over 50 research papers in various national and international journals/conferences. She has received Young Scientist Award from Chhattisgarh Council of Science & Technology and also received Gold Medal in her post graduation.

Prof. Anupam Shukla

Prof. Anupam Shukla is serving as a Professor in Indian Institute of Information Technology and Management Gwalior. He heads the Soft Computing and Expert System Laboratory at the Institute. He has 20 years of teaching experience. His research interest includes Speech processing, Artificial Intelligence, Soft Computing, Biometrics and Bioinformatics. He has published over 100 papers in various national and international journals/conferences. He is editor and reviewer in various journals. He received Young Scientist Award from Madhya Pradesh Government and Gold Medal from Jadavpur University.