Asian Journal of Information Technology 10 (5): 165-169, 2011
ISSN: 1682-3915

Comutational Performance of GRNN in Weather Forecasting

[1]Tiruvenkadam Santhanam and [2]A.C. Subhajini
[1]Department of Computer Science, D.G. Vaishnav College, Chennai, India
[2]Department of Software Engineering, Noorul Islam Engineering College,
Kumaracoil, Kanyakumari, India

Abstract: Accurate weather forecasting plays a vital role for planning day to day activities. Neural network has been use in numerous meteorological applications including weather forecasting. A neural network model has been developed for weather forecasting, based on various factors obtained from meteorological experts. This study evaluates the performance of Generalized Regression Neural Networks (GRNN) model with Radial Basis Function (RBF) with Back Propagation (BPN) neural network. The back propagation neural network, radial basis function neural network and generalized regression neural networks are used in this study to test the performance in order to investigate which technique for weather forecasting most, effective. The prediction accuracy of GRNN is 96.80%. The results indicate that proposed generalized regression neural networks is better than back propagation neural network and radial basis function.

Key words: Multilayer perception, weather forecasting, rainfall prediction, Radial Basis Function (RBF), back propagation, Artificial Neural Network (ANN), Numerical Weather Prediction (NWP), Regression Neural Networks (GRNN)

INTRODUCTION

Weather simply refer to the condition of air on earth at a given place and time. The application of science and technology are to predict the state of the atmosphere in future time for a given location is so important due to its effectiveness in human life (Cheng *et al*., 2010). Today, weather forecasts are made by collecting quantitative data about the current state of the atmosphere and using scientific understanding of atmospheric processes to project how the atmosphere will evolve. The chaotic nature of the atmosphere implies the need of massive computational power required to solve the equations that describe the atmospheric conditions. This is resulted from incomplete understanding of atmospheric processes which mean that forecasts become less accurate as the difference in time between the present moment and the time for which the forecast is being made increases. Weather is a continuous, data-intensive, multidi-mensional, dynamic and chaotic process and these properties make weather forecasting a big challenge. Generally, two methods are used for weather forecasting the empirical approach and the dynamical approach. The first approach is based on the occurrence of analogs and is often referred by meteorologists as analog forecasting. This approach is useful for predicting local-scale weather if recorded data's are plentiful. The second approach is based on equations and forward simulations of the atmosphere and is often referred to as computer modeling. The dynamical approach is only useful for modeling large-scale weather phenomena and may not predict short-term weather efficiently. Most weather prediction systems use a combination of empirical and dynamical techniques (Moro *et al*., 1994).

Artificial Neural Network (ANN) provides a methodology for solving many types of nonlinear problems that are difficult to be solved by traditional techniques (Baboo and Shereef, 2010). Most meteorological processes often exhibit temporal and spatial variability. They are suffered by issues of nonlinearity of physical processes, conflicting spatial and temporal scale and uncertainty in parameter estimates. The ANN has capability to extract the relationship between the inputs and outputs of a process without the physics being explicitly provided (Veisi and Jamzad, 2009). Thus, these properties of ANN are well suited to the problem of weather forecasting (Abd, 2009). The main purpose is to develop the most suitable ANN architecture and its associated training technique for weather prediction. This development will be based on using two different neural network architecture to demonstrate the suitable one for this application. Back Propagation (BPN) feed forward network and radial basis function network which were trained by differential evolution algorithm are

Corresponding Author: Tiruvenkadam Santhanam, Department of Computer Science, D.G. Vaishnav College, Chennai, India

the selected architectures in this study. The basic architecture of the both Radial Basis Functions (RBF) neural network and multilayer feed forward neural networks are given. These neural network architectures are used as a prediction tools for the weather forecasting.

Weather forecasting system structure: Components of a modern weather forecasting system include the following modules: data collection, data assimilation and numerical weather prediction.

Data collection: Observations of atmospheric pressure, temperature, wind speed, wind direction, humidity and precipitation are made near the earth's surface by trained observers, automatic weather stations. The World Meteorological Organization acts to standardize the instrumentation, observing practices and timing of these observations worldwide.

Data assimilation: During the data assimilation process information gained from the observations is used in conjunction with a numerical model most recent forecast for the time that observations were made to produce the meteorological analysis. This is the best estimate of the current state of the atmosphere. It is a three dimensional representation of the distribution of temperature, moisture and wind. The features considered in this study are bar temperature, bar reading, sea level pressure, mean sea level pressure, dry bulb temperature, wet bulb temperature, due point temperature, vapor pressure, wind speed, humidity, cloudiness, precipitation, wind direction, wind speed and for prediction of rain. It is easy to implement and produces desirable forecasting result by training the given data set.

Numerical weather prediction: Numerical Weather Prediction (NWP) uses the power of computers to make a forecast. Complex computer programs, also known as forecast models, run on supercomputers and provide predictions on many atmospheric variables such as temperature, pressure, wind and rainfall. A forecaster examines how the features predicted by the computer will interact to produce the day's weather.

MATERIALS AND METHODS

ANN is an information processing paradigm that is inspired by the way biological nervous systems such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) to solve specific problems. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. Neural networks have broad applicability to real world problems. In fact, they have already been successfully applied in many industries (Dastorani *et al.*, 2010).

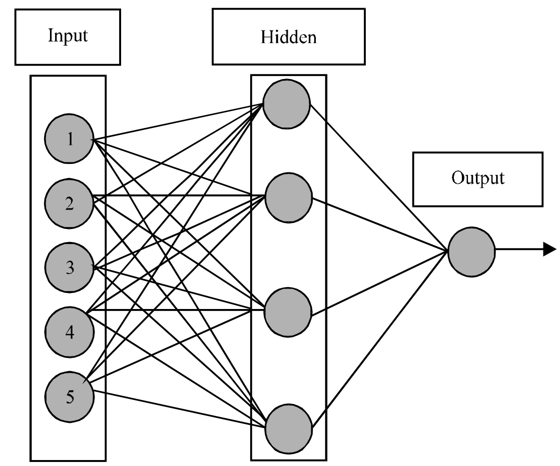

Fig. 1: Neural network architecture

The configuration of the neural network depends highly on the problem. Therefore, it is left with the designer to choose an appropriate number of input, output and hidden layer nodes based on experience. Thus, an appropriate architecture is determined for each application using the trial and error method. The learning rate parameter and momentum term were adjusted intermittently to speed up the convergence.

Feed forward back propagation network: A typical neural network consists of layers. In a single layered network there is an input layer of source nodes and an output layer of neurons. A multi-layer network has in addition one or more hidden layers. A multi-layer neural network is shown in Fig. 1. Extra hidden neurons raise the network's ability to extract higher-order statistics from (input) data. Furthermore a network is said to be fully connected if every node in each layer of the network is connected to every other node in the adjacent forward layer (Khan and Coulibaly, 2010). The network learns by adjusting the interconnections (called weights) between layers. When the network is adequately trained, it is able to generalize relevant output for a set of input data. A valuable property of neural networks is that of generalization whereby a trained neural network is able to provide a correct matching in the form of output data for a set of previously unseen input data.

Learning typically occurs by example through training where the training algorithm iteratively adjusts the connection weights (synapses). Back propagation is one of the most famous training algorithms for multilayer perceptions (Omaima, 2010). BP is a gradient descent technique to minimize the error E for a particular training pattern. For adjusting the weight w_{ij} from the ith input unit to the jth output in the batched mode variant the descent is based on the gradient:

$$\nabla E\left(\frac{\delta E}{\delta w_{ij}}\right)$$

for the total training set:

$$\Delta w_{ij} = -\Sigma^* + \alpha * \Delta w_{ij}(n-1) \tag{1}$$

The gradient gives the direction of error E. The parameters ε and α are the learning rate and momentum respectively (Omaima, 2010). Normally, the learning rate is held constant throughout the training. If the learning rate is too high, the algorithm may oscillate and become unstable. If the learning rate is too small, the algorithm is slow to converge. The performance of the steepest descent algorithm can be improved by using an adaptive learning rate which will keep the learning step size as large as possible while keeping learning stable. The learning rate is made adaptive to the complexity of the local error surface. If the new error exceeds the old error by more than a predefined ratio, the new weights are discarded. In addition, the learning rate is decreased. Otherwise the new weights are kept. If the new error is less than the old error, the learning rate is increased. This architecture was used in the field of forecasting in many fields (Luo and Zhou, 2010).

Basic architecture of RBF: A Radial Basis Function (RBF) network is a special type of neural network that uses a radial basis function as its activation function. RBF networks are very popular for function approximation, curve fitting, time series prediction, control and classification problems (Park and Sandberg, 1991). In RBF networks, determination of the number of neurons in the hidden layer is very important because it affects the network complexity and the generalizing capability of the network. In the hidden layer, each neuron has an activation function. The Gaussian function which has a spread parameter that controls the behavior of the function is the most preferred activation function (Ajeel, 2010). The training procedure of RBF networks also includes the optimization of spread parameters of each neuron. Afterwards, the weights between the hidden layer and the output layer must be selected appropriately.

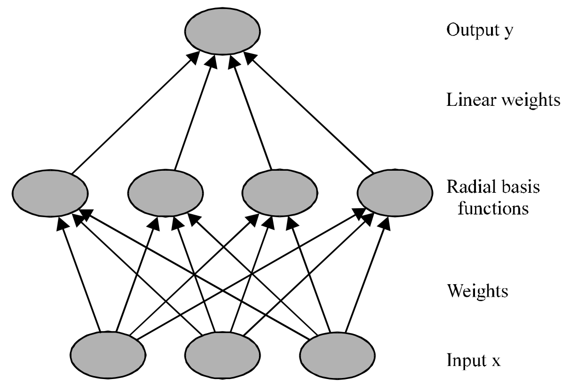

Fig. 2: Block diagram of a RBF network

Finally, the bias values which are added with each output are determined in the RBF network training procedure. RBF network is a type of feed forward neural network composed of three layers, namely the input layer, the hidden layer and the output layer. A general block diagram of an RBF network is shown in Fig. 2. A RBF network with m outputs and hidden nodes can be expressed as:

$$y_i(t) = w_{io} + \sum_{j=1}^{n_h} w_{ij}\Phi\left(\|v(t) - c_j(t)\|\right), \ i = 1,...m \tag{2}$$

Considering this argument, the RBF network with additional linear input connections is used. The proposed network allows the network inputs to be connected directly to the output node via weighted connections to form a linear model in parallel with the non-linear standard RBF model (Ghods and Kalantar, 2010). The new RBF network with m outputs, n inputs, hidden nodes and n_1 linear input connections can be expressed as:

$$y_i(t) = w_{io} + \sum_{j=1}^{n_h} \lambda_{ij} vl(t) + \sum_{j=1}^{n_h} w_{ij}\varphi\left(\| v(t) - c_j(t) \|\right) \tag{3}$$

$$i = 1, 2, \ldots m$$

Where the λ's and vl's are the weights and the input vector for the linear connections may consist of past inputs and outputs (Goriz and Punithori, 2011).

General Regression Neural Network (GRNN): Robert Hecht-Nielsen developed the Generalized regression neural networks. The General Regression Neural Network (GRNN) is one of the most popular neural networks. They have a parallel structure where the learning is one fold that is input to structure to output there is no iterative learning present such as in the case of Multi Layer Perceptrons (MLP) making them fast to some extents.

That is one of the reasons the GRNN are being used in weather forecasting. GRNN is also very unswerving and as the size of the dataset increases the error approaches towards zero. The GRNN works quite accurately with prediction of weather forecasting. The GRNN infrastructure consists of four layers input, hidden, summation and output layer: a basic GRNN comprising of an input, hidden, summation and output layer. A general block diagram of an GRNN network is shown in Fig. 3:

- GRNN has a very parallel structure and generalizes almost better than any other neural network
- GRNN is a non iterative neural network that is data flows from input layer to the output layer so it consumes less memory than the iterative neural networks like back propagation neural networks, etc.
- These qualities make GRNN very feasible for real time applications such as weather forecasting

Steps involved in working of GRNN

Step 1 (Input layer): There is one neuron in the input layer for each predictor variable. In the case of categorical variables, N-1 neurons are used where N is the number of categories. The input neurons (or processing before the input layer) standardizes the range of the values by subtracting the median and dividing by the interquartile range. The input neurons then feed the values to each of the neurons in the hidden layer.

Step 2 (Hidden layer): This layer has one neuron for each case in the training data set. The neuron stores the values of the predictor variables for the case along with the target value. When presented with the x vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies sigma value (s). The resulting value is passed to the neurons in the pattern layer.

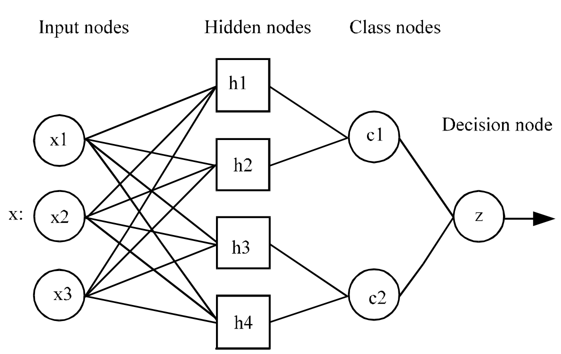

Fig. 3: Block diagram of a GRNN network

Step 3 (Pattern layer/Summation layer): For GRNN networks, there are only two neurons in the pattern layer. One neuron is the denominator summation unit the other is the numerator summation unit. The denominator summation unit adds up the weight values coming from each of the hidden neurons. The numerator summation unit adds up the weight values multiplied by the actual target value for each hidden neuron.

Step 4: The decision layer divides the value accumulated in the numerator summation unit by the value in the denominator summation unit and uses the result as the predicted target value.

RESULTS AND DISCUSSION

BPN, RBF and GRNN are trained with sample of thousand patterns. The performance of the GRNN is compared with performance of the BPN and RBF for weather forecasting. The training of GRNN is faster compared that of BPN and RBF. The classification to predict rainfall is enhanced by GRNN which is shown in Table 1. Performance of classification of rainfall prediction by BPN, RBF and GRNN is shown in Fig. 4 and 5.

To experiment the proposed system a sample dataset is taken from metrological department. These data sets contain real time observation of the weather for a particular period of time. For this experiment, an observation of the complete previous 10 years of data is

Table 1: Percentage of classification of rainfall by BPN, RBF and GRNN

		Percentage of correct classification		
Weather forecasting	No of patterns	BPN	RBF	GRNN
Rain	500	80.30	86.60	97.40
No rain	500	83.60	90.30	96.20
Overall percentage	-	81.90	88.40	96.80

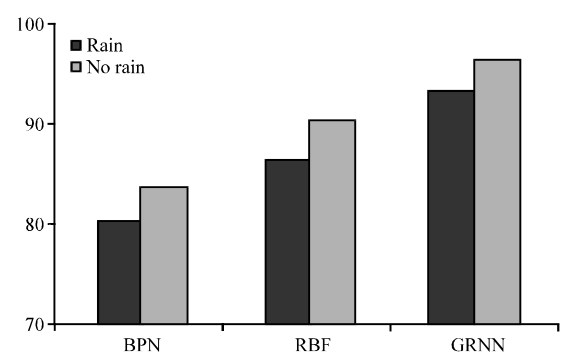

Fig. 4: Performance of classification of rainfall prediction by BPN, RBF and GRNN

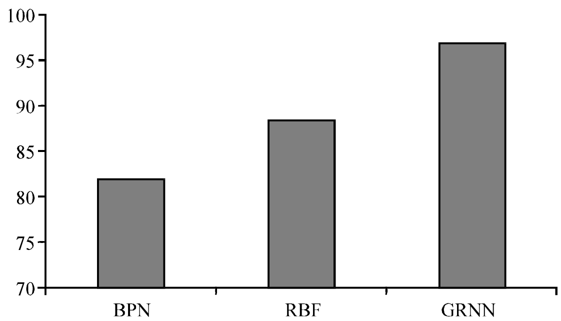

Fig. 5: Overall percentage of classification by BPN, RBF and GRNN

collected from Meteorological department, Kanya Kumari district. The data set contain many attribute. The basic input data for classification needs preprocessing and the above attributes are processed for weather forecasting using BPN, RBF and GRNN. Improvement of classification accuracy in weather forecasting is an important issue. The factors temperature, air pressure, humidity, cloudiness, precipitation, wind direction wind speed of weather forecasting are consolidated from meteorological experts. GRNN are trained with samples and outputs are namely no rain and rain.

CONCLUSION

Neural network has gained great popularity in weather prediction because of their simplicity and robustness. In this study, the performance of Back Propagation Neural Network (BPN), Radical Basis Functioned neural network (RBF) and Generalized Regression Neural Network (GRNN) is compared. Back propagation algorithm and radical basis functioned neural network are too time consuming and the performance is heavily dependent on the network parameters. Compared to BPN and RBF, GRNN gives the overall best results in terms of accuracy and fastest training time. GRNN are much faster and more reliable for the weather forecasting. These proportions make it more effective for fast real time weather forecasting.

REFERENCES

Abd, M.K., 2009. Electricity load forecasting based on framelet neural network technique. Am. J. Applied Sci., 6: 970-973.

Ajeel, S.A., 2010. A novel carbon steel pipe protection based on radial basis function neural network. Am. J. Applied Sci., 7: 248-251.

Baboo, S.S. and I.K. Shereef, 2010. An efficient weather forecasting system using artificial neural network. Int. J. Environ. Sci. Dev., 1: 321-326.

Cheng, C.S., G. Li, Q. Li and H. Auld, 2010. A synoptic weather typing approach to simulate daily rainfall and extremes in Ontario, Canada: Potential for climate change projections. J. Applied Meteorol. Climatol., 49: 845-866.

Dastorani, M.T., H. Afkhami, H. Sharifidarani and M. Dastorani, 2010. Application of ANN and ANFIS models on dryland precipitation prediction (case study: Yazd in Central Iran). J. Applied Sci., 10: 2387-2394.

Ghods, L. and M. Kalantar, 2010. Long-term peak demand forecasting by using radial basis function neural networks. Iran. J. Electr. Electron. Eng., 6: 175-182.

Goriz, M. and T. Punithori, 2011. A new model for time-series forecasting using radial basis functions. Int. J., 181: 903-916.

Khan, M.S. and P. Coulibaly, 2010. Assessing hydrologic impact of climate change with uncertainty estimates: Back propagation neural network approach. J. Hydrometeorol., 11: 482-495.

Luo, L. and L. Zhou, 2010. Application of radial basis function neural network in modeling wastewater sludge recycle system. Commun. Comput. Inform. Sci., 98: 117-122.

Moro, Q.I., L. Alonso and C.E. Vivaracho, 1994. Application of neural networks to weather forecasting with local data. Proceedings of the 12th IASTED International Conference on Applied Informatics, May 1994, Annecy, France, pp: 68-70.

Omaima, N.A., 2010. Improving the performance of backpropagation neural network algorithm for image compression/decompression system. J. Comput. Sci., 6: 1347-1354.

Park, J. and I.W. Sandberg, 1991. Universal approximation using radial-basis-function networks. Neural Comput., 3: 246-257.

Veisi, H. and M. Jamzad, 2009. A complexity-based approach in image compression using neural networks. Int. J. Sign. Process., 5: 82-92.