Short Term Load Forecasting: One Week (With & Without Weekend) Using Artificial Neural Network for SLDC of Gujarat

Tejas Gandhi
M.Tech Student, Electrical Engineering Department
Indus University, Ahmedabad, Gujarat, India

Prof. Sweta Shah
Head of Department, Electrical Engineering Department
Indus University, Ahmedabad, Gujarat, India

Abstract **- This paper present for analysis of short term load forecasting: one week (with & without weekend) using ANN techniques for SLDC of Gujarat. In this paper short term electric load forecasting using neural network; based on historical load demand, The Levenberg-Marquardt optimization technique which has one of the best learning rates was used as a back propagation algorithm for the Multilayer Feed Forward ANN model using MATLAB.12 ANN tool box. Design a model for one week (with & w/o weekend) load pattern for STLF using the neural network have been input variables are (Min., Avg., & Max. load demands for previous week, Min., Avg., & Max. temperature for previous week & Min., Avg., & Max. humidity for previous week). And Nov-12 to Apr-13 (6 Months) historical load data from the SLDC, Gujarat are used for training, testing and showing the good performance. Using this ANN model computing the mean absolute error between the exact and predicted values, we were able to obtain an absolute mean error within specified limit and regression value close to one. This represents a high degree of accuracy.**

Keywords: **Short term load forecasting, Artificial Neural Networks based Levenberg-Marquardt Back Propagation Algorithm, ANN model**

I. INTRODUCTION

The most used thing in today's world is energy. We use energy in various forms in our day to day life like solar energy, wind energy, thermal energy, chemical energies in form of batteries and many other forms of energies. Sometimes we are extravagant and sometimes we are careful. But to provide users uninterrupted supply of electricity there must be proper evaluation of present day and future demand of power. That's why we need a technique to tell us about the demand of consumers and the exact capability to generate the power and this need load forecasting technique because Electrical energy cannot be stored. It has to be generated whenever there is a demand for it. It is, therefore, imperative for the electric power utilities that the load on their systems should be estimated in advance. This estimation of load in advance is known as load forecasting [1].

Load forecasting helps an electric utility to make important decisions including decisions on purchasing and generating electric power, load switching, and infrastructure development. Load forecasts are extremely important for energy suppliers, financial institutions, and other participants in electric energy generation, transmission, distribution, and markets [4].

Load forecasts can be divided into three categories: i) Short-term forecasts which are usually from one hour to one week, ii) Medium forecasts which are usually from a week to a year, and iii) Long-term forecasts which are longer than a year. The forecasts for different time horizons are important for different operations within a utility company. The natures of these forecasts are different as well.

For these three categories of load forecasting are depend on various factors like for: i) For Short-term load forecasting several factors should be considered as: Time factors, Weather data (Temperature & Humidity) and Customer classes and ii) For The medium- and long-term forecasts take into account: The historical load, Weather data (Temperature & Humidity), The number of customers in different categories, The appliances in the area and their characteristics including age, The economic and demographic data and their forecasts and The appliance sales data and other factors [3].

STLF can be performed using many techniques such as similar day approach, various regression models, time series, statistical methods, fuzzy logic, artificial neural networks, expert systems, etc. But application of artificial neural network in the areas of forecasting has made it possible to overcome the limitations of the other methods mentioned above used for electrical load forecasting [2].

The use of artificial neural networks (ANN) has been a widely studied electric load forecasting technique since 1990. NNs are able to give better performance in dealing with the non-linear relationships among the input variables by learning from training data set.

In this paper involves the design of an ANN STLF model for the SLDC, Gujarat in order to obtain accurate system that predicted for one week (with & w/o weekend) load demand pattern. As inputs we took the previous week Min., Avg., &

Max. Load demand as well as temperature and humidity for Min., Avg., & Max. for previous week. Load forecast which is necessary for the operational planning of the power system utility company. And in order to determine the connection weights between the neurons, the Levernberg Marquardt back-propagation algorithm available from MATLAB.12 ANN tool box was used. The network was trained with load data of Nov-12 to Apr-13 (6 Months) period which was obtained from the SLDC, Gujarat [5].

The paper begins with an introduction to STLF followed by for a description of the designed neural network model. The paper concludes with a discussion of the results and a comparison between ANN error and Analytical error for load data of Nov-12 to Apr-13 (6 Months) period.

II. ARTIFICAL NEURAL NETWORK

Neuron is an electrically excitable cell that processes and transmits information through electrical and chemical signals. Synapse is a structure that permits a neuron to pass an electrical or chemical signal to another neuron. Neurons can connect to each other to form Neural Networks.

A neural network is a machine that is designed to model the way in which the brain performs a particular task. The network is implemented by using electronic components or is simulated in software on a digital computer.

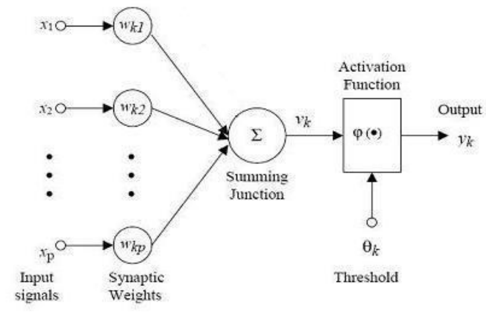
The outputs of an artificial neural network are some linear or nonlinear mathematical function of its inputs. In practice network elements are arranged in a relatively small number of connected layers of elements between network inputs and outputs. Feedback paths are sometimes used.

In applying a neural network to electric load forecasting, one must select one of a number of architectures (e.g. Hopfield, back propagation, Boltzmann machine), the number and connectivity of layers and elements, use of bi-directional or uni-directional links, and the number format (e.g. binary or continuous) to be used by inputs and outputs, and internally. The most popular artificial neural network architecture for electric load forecasting is back propagation [8].

A. Mathematical Model of Neural Network

A neuron is an information processing unit that is fundamental to the operation of a neural network. The three basic elements of the neuron model are:

i. A set of weights, each of which is characterized by a strength of its own. A signal x_j connected to neuron k is multiplied by the weight w_{kj}. The weight of an artificial neuron may lie in a range that includes negative as well as positive values.
ii. An adder for summing the input signals, weighted by the respective weights of the neuron.
iii. An activation function for limiting the amplitude of the output of a neuron. It is also referred to as squashing function which squashes the amplitude range of the output signal to some finite value.



(Fig.1 Simple model of Neural Network)

B. Benefits of ANN

i. They are extremely powerful computational devices.
ii. Massive parallelism makes them very efficient.
iii. They can learn and generalize from training data.
iv. They are particularly fault tolerant.
v. They are very noise tolerant.

C. Network Architecture

There are two fundamental different classes of network architectures:

i. Single layer feed forward network: It has only one layer of computational nodes (output layer). It is a feed forward network since it does not have any feedback. The single layer feed-forward network consists of a single layer of weights, where the inputs are directly connected to the outputs, via a series of weights. The synaptic links carrying weights connect every input to every output, but no other way. The sum of products of the weights and the inputs is calculated in each neuron node, and if the value is above some threshold (typically 0) the neuron fires and takes the activated value (typically 1); otherwise it takes the deactivated value (typically -1). [6].

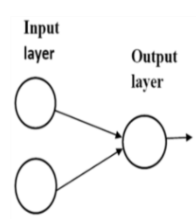


Fig. 2(a)

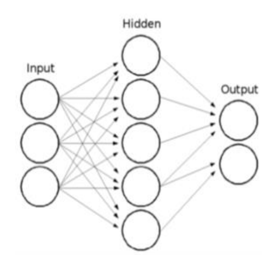


Fig.2(b)

(Fig 2(a) Single-layer Feed forward Network & Fig. 2(b) Multi-layer Feed forward Network of ANN)

ii. Multi-layer feed forward network: It is a feed forward network with one or more hidden layers. The source nodes in the input layer supply inputs to the neurons of the first hidden layer. The outputs of the first hidden layer neurons are applied as inputs to the neurons of the second hidden layer and so on. If every node in each layer of the network is connected to every other node in the adjacent forward layer, then the network is called fully connected. If however some of the links are missing, the network is said to be partially connected. Recall is instantaneous in this type of network.

D. Learning Processes of ANN

By learning rule we mean a procedure for modifying the weights and biases of a network. The purpose of learning rule is to train the network to perform some task. They fall into three broad categories:

i. Supervised learning: The learning rule is provided with a set of training data of proper network behavior. As the inputs are applied to the network, the network outputs are compared to the targets. The learning rule is then used to adjust the weights and biases of the network in order to move the network outputs closer to the targets.
ii. Reinforcement learning: It is similar to supervised learning, except that, instead of being provided with the correct output for each network input, the algorithm is only given a grade. The grade is a measure of the network performance over some sequence of inputs.
iii. Unsupervised learning: The weights and biases are modified in response to network inputs only. There are no target outputs available. Most of these algorithms perform some kind of clustering operation. They learn to categorize the input patterns into a finite number of classes [5].

III. BACK PROPAGATION ALGORITHM

The back propagation algorithm is used to find a local minimum of the error function. Error back-propagation learning consists of two passes through the different layers of the network: a forward pass and a backward pass. In the forward pass, an input vector is applied to the nodes of the network, and its effect propagates through the network layer by layer. Finally, a set of outputs is produced as the actual response of the network. During the forward pass the weights of the networks are all fixed. During the backward pass, the weights are all adjusted in accordance with an error correction rule. The actual response of the network is subtracted from a desired response to produce an error signal. This error signal is then propagated backward through the network, against the direction of synaptic connections. The weights are adjusted to make the actual response of the network move closer to the desired response [9].

Let us consider the three layer network with input layer having *'l'* nodes, hidden layer having *'m'* nodes, an output layer with *'n'* nodes. We consider sigmoidal functions for activation functions for the hidden and output layers and linear activation function for input layer. The number of neurons in the hidden layer may be chosen to lie between *'l'* and *'2l'*.

Algorithm illustrates the step by step procedure of the back propagation algorithm

Step 1: It is proved that the neural networks better if input and outputs lie between 0-1. For each training pair, assume there are 'l' inputs given by $\underset{l*1}{\{I\}_I}$ and 'n' outputs $\underset{n*1}{\{O\}_o}$ in normalized forms.

Step 2: Assume the number of neurons in the hidden layer to lie between l<m<2l.

Step 3**:** [V] represents the weight of synapses connecting input neurons and hidden neurons and [W] represents weights of synapses connecting hidden neurons and output neurons. the threshold values can be taken as 0.

$$[V]^0 = [\text{random weights}]$$
$$[W]^0 = [\text{random weights}]$$
$$[\Delta V]^0 = [\Delta W]^0 = [0] \quad (1)$$

Step 4: For the training data, present one set of inputs and outputs. Present the pattern to the input layer $\{I\}_I$ as inputs to the input layer. By using linear activation function, the output of the input layer may be evaluated as

$$\underset{l*1}{\{O\}_I} = \underset{l*1}{\{I\}_I} \quad (2)$$

Step 5: Compute the inputs to the hidden layer by multiplying corresponding weights of synapses as

$$\underset{m*1}{\{I\}_H} = \underset{m*l}{[V]^T} \underset{l*1}{\{O\}_I} \quad (3)$$

Step 6: Let the hidden layer units evaluate the output using the sigmoidal function as

$$\{O\}_H = \underset{m*1}{\left\{ \begin{array}{c} \vdots \\ \frac{1}{(1+e^{-I_{Hi}})} \\ \vdots \end{array} \right\}} \quad (4)$$

Step 7: Compute the inputs to the output layer by multiplying corresponding weights of synapses

$$\underset{n*1}{\{I\}_O} = \underset{n*m}{[W]^T} \underset{m*1}{\{O\}_H} \quad (5)$$

Step 8: Let the output layer units evaluate the output using the sigmoidal function as

$$\{O\}_O = \left\{ \begin{array}{c} \vdots \\ \frac{1}{(1+e^{-I_{Oj}})} \\ \vdots \end{array} \right\} \quad (6)$$

Step 9: Calculate the error and the difference between the network output and the desired output as for the i^{th} training set as

$$E^p = \frac{\sqrt{\Sigma\,(Tj - Ooj)^2}}{n} \tag{7}$$

Step 10: Find $\{d\}$ as

$$\underset{n*1}{\{d\}} = \left\{(T_K - O_{ok})O_{ok}(1 - O_{ok})\right\} \tag{8}$$

Step 11: Find $[Y]$ matrix as

$$\underset{m*n}{\{Y\}} = \underset{m*1}{\{O\}_H}\ \underset{1*n}{\{d\}} \tag{9}$$

Step 12: Find

$$\underset{m*n}{[\Delta W]^{t+1}} = \alpha\,\underset{m*n}{[\Delta W]^t} + \eta\,\underset{m*n}{[Y]} \tag{10}$$

Step 13:

$$\underset{m*1}{\{e\}} = \underset{m*n}{[W]}\ \underset{n*1}{[d]} \tag{11}$$

$$\underset{m*1}{\{d^*\}} = \underset{m*1}{\left\{e_i(O_{Hi})\,(1 - O_{Hi})\right\}} \tag{12}$$

Find [X] matrix as

$$\underset{1*m}{[X]} = \underset{l*1}{\{O\}_I}\ \underset{1*m}{[d^*]} = \underset{l*1}{\{I\}_I}\ \underset{1*m}{[d^*]} \tag{13}$$

Step 14: Find

$$\underset{1*m}{[\Delta V]^{t+1}} = \alpha\,\underset{1*m}{[\Delta V]^t} + \eta\,\underset{1*m}{[X]} \tag{14}$$

Step 15: Find

$$[V]^{t+1} = [V]^t + [\Delta V]^{t+1}$$

$$[W]^{t+1} = [W]^t + [\Delta W]^{t+1} \tag{15}$$

With the updated weights [V] and [W], error is calculated again and next training set is taken and error will be adjusted

Step 16: Find error rate as

$$\text{Error rate} = \frac{\Sigma\, E_p}{nset} \tag{16}$$

Step 17: Repeat steps 4-16 until the convergence in the error rate is less than the tolerance value. Once weights are adjusted the network is ready for inference.

IV. LOAD FORECASTING USING ANN

The learning function used in the training process is a gradient descent with momentum weight/bias function, which allows calculating the weight change for a given neuron. It is expressed as

$$dW = m_c * dW_{prev} + (1 - m_c) * l_r * gW \tag{17}$$

Where dW_{prev} is the previous weight change, g_W is the weight gradient with respect to the performance, l_r is the learning rate, and mc is the momentum.

A. ANN Based LF Flow Chart

The STLF procedure for the chosen ANN model is shown in Fig. 3 [8].

i. Input Variable Selection: Input variables such as load, day type, temperature and spot prices of the previous day, and day type, temperature and spot prices of the forecasting day are initially chosen.

ii. Data Pre-processing: Improperly recorded data and observation error are inevitable. Hence, bad and abnormal data are identified and discarded or adjusted using a statistical method to avoid contamination of the model.

iii. Scaling: Since the variables have very different ranges, the direct use of network data may cause convergence problems. Two scaling schemes are used and compared.

iv. Training: Each layer's weights and biases are initialized when the neural network is set up. The network adjusts the connection strength among the internal network nodes until the proper transformation that links past inputs and outputs from the training cases is learned. Data windows are used for training and moved one day ahead.

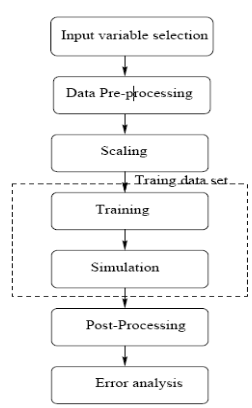
v. Simulation: Using the trained neural network, the forecasting output is simulated using the input patterns.

vi. Post-Processing: The neural network output need de-scaling to generate the desired forecasted loads. If necessary, special events can be considered at this stage.

vii. Error Analysis: As characteristics of load vary, error observations are important for the forecasting process. Hence, the following Mean Absolute Percentage Error (MAPE) ε and Root Mean Square Error (RMSE) σ are used here for after-the-fact error analysis

$$\varepsilon = \frac{1}{N}\sum_{i=1}^{N}\frac{|X_t - X_f|}{X_t} * 100 \tag{18}$$

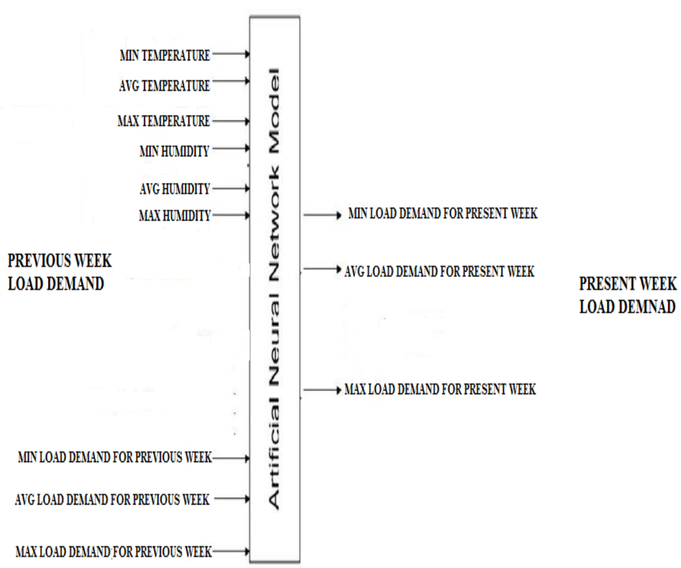
$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_t - X_f)^2} \tag{19}$$



(Fig.3 ANN Based Load Forecasting Flow chart)

B. Approach of STLF Using ANN

A broad spectrum of factors affect the system's load level such as trend effects, cyclic-time effects, and weather effects, random effects like human activities, load management and thunderstorms. Thus the load profile is dynamic in nature with temporal, seasonal and annual variations. In this paper we developed a system that predicted for one week (with & w/o weekend) load demand pattern. As inputs we took the previous week Min., Avg., & Max. Load demand as well as temperature and humidity for Min., Avg., & Max. for previous week. The inputs were fed into our Artificial Neural Network (ANN) and after sufficient training were used to predict the load. A schematic model of our system is shown in Fig 4. The inputs given are: (i) Min, Avg and Max Temperature of Previous week (ii) Min, Avg and Max Humidity of Previous week (iii) Min, Avg and Max Load Demand of Previous week And the output obtained was the predicted Min, Avg and Max load demand for the next week. The flow chart is shown below [11].



(Fig.4 Input-Output Schematic for Short Term Load Forecasting)

V. SIMULATION RESULT

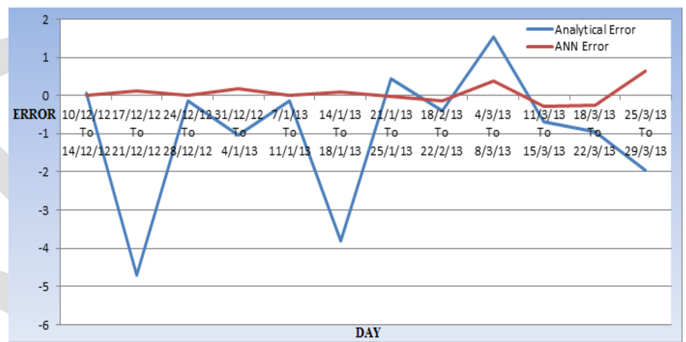
Without Weekend (5 Days)		
Date	Analytical Error	ANN Error
10/12/12 To 14/12/12	0.0776	0.013
17/12/12 To 21/12/12	-4.710	0.108
24/12/12 To 28/12/12	-0.143	-0.00015
31/12/12 To 4/1/13	-1.033	0.177
7/1/13 To 11/1/13	-0.133	0.0022
14/1/13 To 18/1/13	-3.804	0.090
21/1/13 To 25/1/13	0.446	-0.022
18/2/13 To 22/2/13	-0.396	-0.151
4/3/13 To 8/3/13	1.544	0.389
11/3/13 To 15/3/13	-0.689	-0.270
18/3/13 To 22/3/13	-0.945	-0.244
25/3/13 To 29/3/13	-1.958	0.634

(Table 1: ANN Error v/s Analytical Error of w/o weekend for Nov-12 to Apr-13 for SLDC, Gujarat)

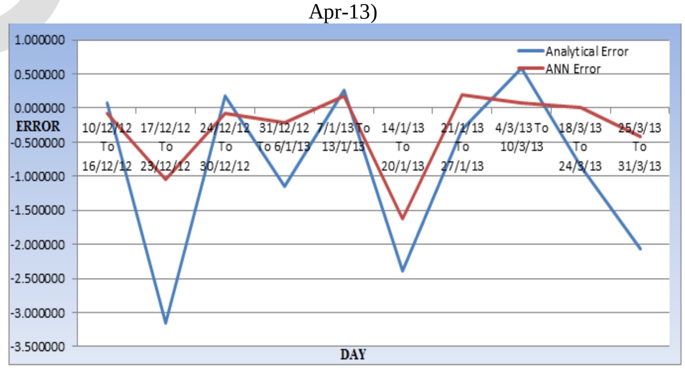
With Weekend (7 Days)		
Date	Analytical Error	ANN Error
10/12/12 To 16/12/12	0.077	-0.073
17/12/12 To 23/12/12	-3.159	-1.041
24/12/12 To 30/12/12	0.173	-0.077
31/12/12 To 6/1/13	-1.142	-0.219
7/1/13 To 13/1/13	0.257	0.180
14/1/13 To 20/1/13	-2.394	-1.631
21/1/13 To 27/1/13	-0.308	0.187
4/3/13 To 10/3/13	0.586	0.081
18/3/13 To 24/3/13	-0.866	0.011
25/3/13 To 31/3/13	-2.063	-0.412

(Table 2: ANN Error v/s Analytical Error of with weekend for Nov-12 to Apr-13 for SLDC, Gujarat)

VI. ANALYSIS OF SIMULATION RESULT FOR STLF



(Fig.5 Analytical Error v/s ANN Error for w/o weekend for Nov-12 to Apr-13)



(Fig.6 Analytical Error v/s ANN Error for with weekend for Nov-12 to Apr-13)

VII. CONCLUSION

The results obtained from testing the trained neural network for one week (w/o and with weekend) data for Nov-12 to Apr-13 (6 Months) period using ANN STLF model for SLDC, Gujarat. It shows that the ANN Model has been given good performance and reasonable prediction accuracy was achieved for this model.

The absolute mean error (%AME) between the 'Analytical' and 'ANN' loads for w/o weekend and weekday for Nov-12 to Apr-13 (6 Months) period have been calculated and presented in the table. 1 & 2 and fig. 5 & 6. This represents a high degree of accuracy in the ability of neural networks to forecast electric load and Regression value close to one.

The results suggest that ANN model with the developed structure can perform good prediction with least error and finally this neural network could be an important tool for short term load forecasting.

ACKNOWLEDGMENT

I wish to express my profound sense of deepest gratitude to my motivator Prof. Sweta Shah, HOD, Electrical Engineering Department, Indus University, Ahmedabad for her valuable guidance, sympathy and co-operation during the entire period of this paper. I wish to convey my sincere gratitude to all the faculties of Electrical Engineering Department, who have enlightened me during my studies.

REFERENCES

[1]. K Geetha and Sk. Mohiddin, "Artificial Neural Network Approach for Short Term Load Forecasting for IJARCSSE Region", International Journal of Computer, and Soffware Engineering Volume 3 , Number 4, 2007 ISSN 2277-128X.

[2]. K.Y. Lee, Y.T. Cha and J.H. Park, "Short Term Load Forecasting Using An Artificial Neural Network", IEEE Transactions on Power Systems, Vol 1, No 1, February 1992.

[3]. G.A. Adepoju, S.O.A. Ogunjuyigbe and K.O. Alawode, "Application of Neural Network to Load Forecasting in Nigerian Electrical Power System", Volume 8, Number 1, May 2007 (Spring).

[4]. "Load Forecasting" Chapter 12, E.A. Feinberg and Dora Genethlio, Page 269 – 285, from links: www.ams.sunysb.edu and www.usda.gov

[5]. P. Werbos, "Generalization of backpropagation with application to recurrent gas market model", Neural Networks, vol.1,pp.339 – 356,1988

[6]. P. Fishwick, "Neural network models in simulation: A comparison with traditional modeling approaches," Working Paper, University of Florida, Gainesville, FL,1989.

[7]. Dr. John A. Bullinaria, "Introduction to Neural Networks - 2nd Year UG, MSc in Computer Science: Lecture Series".

[8]. Yasser Al-Rashid and Larry D. Paarmann, "Short –Term Electric Load Forecasting Using Neural Network Models", 0-7803-3636-4/97, 1997 IEEE.

[9]. Khotanzad, A., Afkhami-Rohani, R., and Maratukulam, D.ANNSTLFArtificial neural network shortterm load forecastergeneration three, IEEE Trans. on Power Syst., 13, 4, 1413–1422, November, 1998.

[10]. I. Moghram and S. Rahman, "Analysis and evaluation of five short termload forecasting techniques," IEEE Trans. Power Syst., vol. 4, no. 4, pp. 1484–1491, Nov. 1989.

[11]. http://www.wunderground.com/history/airport/VAAH/2014/1/1/DailyHistory.html?req_city=Ahmedabad&req_statename=India&reqdb.zip=00000&reqdb.magic=1&reqdb.wmo=42647.