NSE TECHNICAL TRANSACTION VOL.45, NO. 3, JULY – SEPTEMBER, 2010
M. O. OMOIGUI, T. O. ATOKI AND M. S. BAMIGBOYE

SHORT-TERM LOAD FORECASTING OF THE POWER HOLDING COMPANY OF NIGERIA ELECTRIC NETWORK USING ARTIFICIAL NEURAL NETWORK

BY

M.O. OMOIGUI, T.O. ATOKI, BAMIGBOYE
Department of Electronic and Electrical Engineering
Obafemi Awolowo University Ile-Ife, Nigeria

Abstract

This paper is concerned with the application of Artificial Neural Network (ANN) technique to forecast the short-term electrical load of the Power Holding Company of Nigeria (PHCN) network. Electric power load forecasting is essential for the purpose of optimal planning and operation of a large system. It is also a useful tool for electric utilities in several applications such as economic allocation of generation, energy transaction, unit commitment, maintenance scheduling and optimal energy interchange between utilities.

Important factors considered in this investigation include past load historical data, season and weather information and effects.

Multi-Layer Perceptron (MLP) was used in the implementation of the Feed Forward Backward Propagation in order to create the ANN algorithm using Matlab-based software. Historical load data from National Control Center (NCC) Osogbo was used in the training of this network for Morning and Evening load patterns. ANN algorithm was trained with different hidden neurons ranging from 2-10 neurons. The best forecast was obtained using 2 hidden neurons for Morning and Evening short term load forecast and the average percentage errors were 1.68% and 3.90% as compared to the ones obtained from PHCN load forecasting techniques which were 50.56% and 70.42% respectively.

1.0 Introduction

Load Forecasting can be defined as the systematic procedure for quantitatively defining future electrical power requirement.

Load forecasting is essential to the operating and planning of a utility company, and therefore plays a dominant part in the economic optimization and secure operation of electric power systems. Load forecasts are extremely important for energy suppliers, financial institutions, and other participants in electric energy generation, transmission, distribution, and marketing.

Load forecasts can be divided into three categories: short- term which are usually from one hour to one week, medium forecasts which are usually from a week to a year, and long- term forecasts which are longer than a year.

The nature of these forecasts is different as well. For example, for a particular region, it is possible to predict the next day with an accuracy of approximately 1-3%. However, this paper focused on short term forecasting where the prediction time varies between a few hours and about one week. Short-term power system load is mainly dependent on nonlinear combinations of variables that can be classified according to their dependence on weather, social occasions and seasonal factors [1].

Short Term Load Forecasting (STLF) is a very useful tool for electric utilities in several applications such as economic allocation of generation, energy transaction, and optimal energy interchange between utilities, unit commitment and maintenance scheduling.

Due to the shrinking spinning reserve and the seasonal high-energy demand, utilities are dependent more than ever on short term forecasting for daily transactions. STLF activities include forecasting the daily peak load, total daily energy, and daily load curve as a series of 24 hourly forecasted electrical loads. Load forecasting is somewhat a difficult task. The reasons are: (i) electric load is complex and exhibits several levels of seasonality: the load at a given hour is dependent not only on the load at the previous hour, but also on the load at the same hour on the previous day, and on the load at the same hour on the day with the same denomination in the previous week [2], and (ii) there are many important exogenous variables that must be considered, especially weather and occasion related variables.

2.0 Load Forecasting Technique

2.1 Approach Used By National Control Center

Conventionally, load forecasting has been based on series of empirical formulae derived from several operating and prevailing condition and assumptions. This technique which is a quasi-time series/regression method relied on a few highly experienced in-house human experts to perform judgmental forecasts manually employing similar day data. The Power Holding Company of Nigeria (PHCN) adopts this approach at the National Control Centre Osogbo, to forecast the electrical load for Nigeria. The steps adopted by PHCN may be summarized as follows:

a) The seasonal weather load model based on historical data is determined by plotting scatter diagrams of daily peaks versus an appropriate weather variable using regression analysis or curve fitting.

b) Separation of historical weather sensitive and non-weather sensitive components of peak demand performed using weather load model.

c) The mean and variance of non-weather sensitive components of demand is forecasted.

d) Extrapolate weather load model and the forecast mean and variance of weather sensitive components.

e) Determine mean, variance and density function of total load forecast.

f) Finally, calculate density function of monthly or annual forecast.

Each of these factors has to be taken into consideration when load forecasting is to be performed. The steps are cumbersome and highly susceptive to errors without employing appropriate algorithm.

2.2 Proposed Scheme

Several approaches to load forecasting have been presented in the literature [1 – 8].

Majority of recently reported approaches are based on Artificial Neural Network (ANN) techniques. The attraction of the method lies in the assumption that neural networks are able to learn properties of the load which would otherwise require tedious analysis to obtain.

An ANN can be defined as a highly connected array of elementary processors known as neurons. The block diagram of the proposed scheme employing a Neural

Network model is shown in figure 1.0. The model is composed of many computing elements that are usually denoted as neurons working in parallel. The elements are connected by synaptic weights, which are allowed to adapt through a learning process. In this scheme, the neural network uses supervised learning in which the actual numerical weights assigned to element inputs are determined by matching historical data (such as time and weather) to desired outputs (such as historical electric loads) in a pre-operational "training session". The network weights are then adapted in order to minimize the difference between network outputs and desired outputs. The algorithm employs the back propagation technique to predict the load after considering all necessary parameters including weather and recent load history.

Apart from providing reinforcement on how it is performing on the task, the technique also provides information about errors which are filtered back through the system and is used to adjust connections between layers to improve performance. An ANN algorithm incorporated in Matlab 7.0 is employed for training data from which the network learns.

The flow diagram for the simulation process is shown in the figure 2.0.

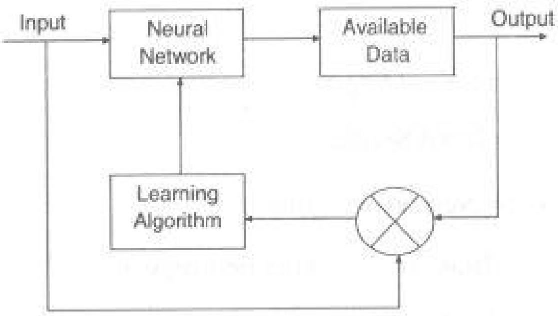


Figure 1.0: ANN Input and Output Target

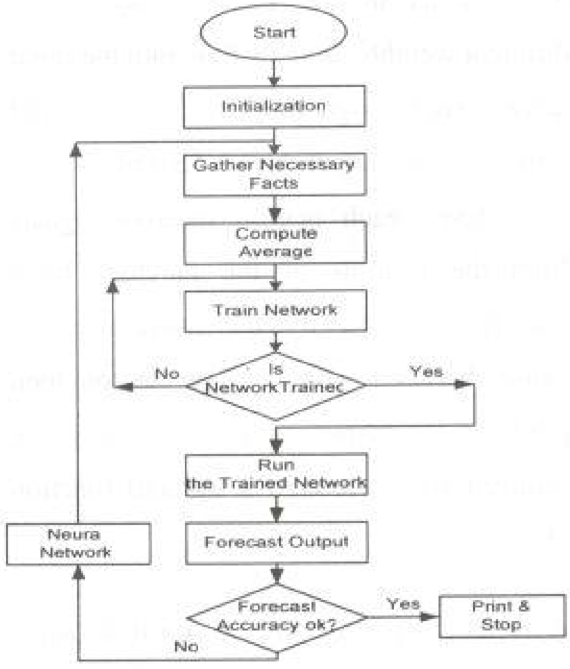


Figure 2.0 Simulation Flow Diagrarr

3.0 Algorithm

One of the most important developments of recent network research is the discovery of learning algorithm to adjust the weight in Multi Layer Perception (MLP). The algorithm is known as back propagation since the weights are adjusted from the output layer backwards layer-by-layer to reduce output errors [3, 4, 9].

In general terms, an artificial neural network consists of a large number simple processing unit's linked by weighted connections. The MLP type ANN consists of one input layer, one or more hidden layers and one output layer. Each layer employs several neurons and each neuron in a layer is connected to the neurons in the adjacent layer with different weights. Signals flow into the input layer, pass through the hidden layers, and arrive at the output layer. Except for the input layer, each neuron receives signals from the neurons of the previous layer linearly weighted by the interconnecting values between neurons. The neuron then produces its output signal by passing the summed signal through a sigmoid function [3].

A certain network may be tuned to some particular problem by varying the connection topology and values of the connection weight between units. Once a neural network has analyzed the given dataset (i.e. trained), it is able to make prediction based on these found hidden dependencies.

3.1 Multi-Layer Perceptron (MLP)

Multi-layer Perceptron is the most popular neural network and most of the reported short-term load forecasting model are based on it. The basic unit (neuron) of the network is a perceptron. This is a computation unit, which produces its output by taking a linear combination of the input signals and by transforming this by a function called activity function [10]. The output of the perceptron on a function of the signals can thus be written as:

$$Y = \sigma\left(\sum_{i}^{n} W_i X_i - \theta\right)$$

(1)

Where Y = Output

X_i = Input Signals

W_i = Neuron Weights

θ = Bias term (another neuron weight)

σ = Activity function

The MLP network may have several layers of neurons. Each neuron in a certain layer is connected to each neuron of the next layer.

There are no feedback connections. A three layer MLP network is shown in figure 3.0.

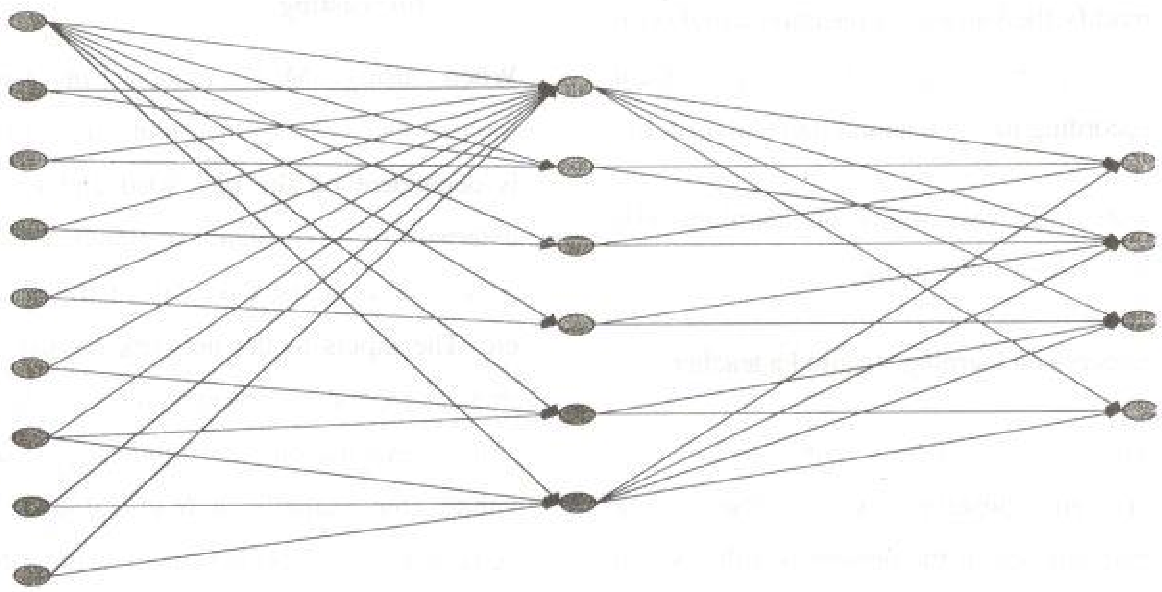


Figure 3.0: A three Layer MLP network

As an N-dimensional input vector is fed to the network, an M-dimensional output vector is produced. The function can be written as:

$$y = f(x;W) = \sigma(W_n\sigma(W_{n-1}\sigma(W_1x)...))) \quad (2)$$

where

y = output vector

x = input vector

W_i = matrix containing the neuron weights of the i^{th} hidden layer.

The most often used MLP network consists of the three layers: an inputs layer, hidden layer, and an output layer.

The activation function used in the hidden layer is usually nonlinear (logistic sigmoid or hyperbolic tangent) and the activation function in the output layer is a linear network. The input layer nodes are designed to contain pertinent data that are considered to have major relation to the desired forecast value. These could include past two days average temperature, past 4 hours temperature, next hour temperature, and hour of the day, day of the week, and electric load for corresponding periods.

3.2 Learning Procedure

The purpose of learning function is to modify the variable connection weights on the inputs of each processing element according to some neural based algorithm.

There are two types of learning, viz: Supervised and Unsupervised Learning.

Supervised learning required a teacher.

The teacher may be a training set of data or an observer who grades the performance of the network results. When there is no external teacher, the system must organize itself by some internal criteria designed into the network. The most common network type using supervised learning is a feed forward (signal transfer) network.

For the supervised learning, the network weights are adjusted by training the network and making it learn through examples. The basic principle is to give the network input signal and desired outputs. To each input signal, the network produces an output signal and the learning aims at minimizing the sum of squared errors between the desired and actual outputs until the weights stabilize and the sum of the squared errors converges to a specified minimum value.

3.3 Basic MLP model for load forecasting

When using MLP models in load forecasting, it is assumed that future load is dependent on the past load and some external factors such as temperature, season, time of the day, day of the week etc. The inputs to the network consist of those temperature values and past load values, and the output is the target load values (for example, a load value of a certain hour, the peak load of a day, the total load of a day, etc). Consequently, the building of a MLP load forecasting model is a nonlinear system identification problem.

3.3.1 Formulating the Algorithm

The objective of short-term load forecasting is to predict the short-term (24 hours or 7 days) load, $L(i)$, i = 1, 2, …24 (or 168 hours for 7 days). There are four main steps involved in the process [5].

Step 1: Read in the day to be forecasted and the weather forecast and other variables.

Step 2:. Identify the day type (weekday or weekend) and obtain the normalized hourly loads, $L_n(i)$ for that day type. These had been previously stored in data base to represent the load pattern for each day type.

Step 3: Compute the forecasted daily peak and valley load using the following equations:

$$L_p = g_p x T_p + h_p \tag{3}$$

$$L_v = g_v x T_v + h_v \tag{4}$$

where

L_p = peak load of the day

L_v = valley load of the day

T_p = equivalent forecasted high temperature of the day

T_v = equivalent forecasted low temperature of the day

$g_p, h_p, g_v, and \quad h_v$ =coefficients determined by least-square error technique using the load and weather data in the data base.

Step 4: Compute the forecasted hourly loads $L(i)$ as follows:

$$L(i) = L_n(i)(L_p - L_v) + L_v$$

$$i = 1, 2, \ldots 24 \text{ (or 168)} \tag{5}$$

All necessary data was gathered from several sources, particularly from National Control Centre (NCC), PHCN, Osogbo.

The data that were either in spreadsheet or database are used for training the network.

The Matlab-based program allows the user to define variable in binary code, which is 0 and 1.

3.3.2 Training The Network

This involves repeatedly presenting a set of facts to the network. The actual power generated were used as the input load to train the network which takes in each input and makes a "guess" as to the output, checks this "guess" against the output supplied and makes corrections to the internal connections if this "guess" is incorrect. This process is repeated for each fact in turn until the network learns the fact well enough to be useful. The Matlab-based program is designed to take care of the training; however it may be necessary for the user to interact with the network during the training process to influence results in the correct direction.

3.3.3 Testing The Network

The network may be tested interactively on the screen or through the network testing file from which the Matlab-based program runs and compares its output with the known results. If the answers are good enough, the network is ready to be used but if not, it is necessary to reassess the data, change some learning parameter or redesign the network.

The actual generated power (MW) was used as the input values while the average peak load (MW) was used as the target. The loads were grouped into "Morning Load Forecast" and "Evening Load Forecast". The historical load data obtained from NCC Osogbo were used to train with Matlab-base algorithm (Matlab 7.0 version) using the actual generated power and average peak loads as the targets. The hour by hour forecasting model was applied for each test day recursively 24 times. After forecasting the load patterns for each test day, the forecasts were compared with the actual load data and the average percentage were calculated. The average percentage error forecasts were used as a measure of performance.

4.0 Test Results and Discussions

The results of the training and testing of the morning and evening load forecasting are shown in Table1. The network was trained on the past record taken from 28th March to 4th April 2008. All load and weather condition were taken into consideration during the training so that the model could adapt to all conditions.

In table 1, the "Input" is denoted "p", while the "Target" is denoted "t".

Table 1: Input and Target for the short term forecasting using the hourly generated peak power a week.

(a) Morning Load Forecast

p = [2906 2838 2851 2882 2892 2947 2950 2920 2804 2703 2700 2749;

2977 2971 2965 2969 3034 3315 3186 3016 3056 2972 2921 2924;

2939 2971 3024 3014 3133 3263 3242 3051 2998 2812 2832 2805;

3082 3054 3056 3058 3136 3229 3110 3007 3017 2952 2934 2875;

3001 3003 3006 3014 3036 3106 3012 2979 2993 2851 2660 2777;

3043 3029 3002 2988 3002 3063 3075 2979 2811 2679 2712 2629;

3185 3059 3054 3059 3091 3156 3122 3053 2971 2927 2995 2873]

t= [3019 2989 2994 2998 3050 3154 3100 3000 2950 2842 2822 2805];

(b) Evening Load Forecast

p = [2796 2894 2894 2893 2871 2971 3214 3246 3240 3227 3008 2994;

2907 2896 2847 2842 2799 2924 3148 3300 3269 3268 3216 3146;

2760 2786 2726 2885 2854 2919 3084 3185 3193 3157 3137 3094;

2913 2849 2914 2795 2798 2805 2997 3129 3106 3138 3050 2989;

2764 2791 2903 2915 2916 2923 3050 3294 3221 3230 3115 3055;

2655 2697 2857 2939 2945 3027 3194 3230 3260 3209 3006 3092;

2784 2793 2821 2919 2939 2964 3144 3182 3259 3252 3025 3010]

t = [2797 2815.1 2851.7 2884 2874.6 2933.3 3118.7 3223.7 3221.1 3211.6 3079.6 3054].

The percentage errors were calculated as:

$$\% \text{ error calculated} = \frac{Actual\ \ Load - ANN\ \ Forecast\ \ x\ \ 100}{Actual\ \ Load} \qquad (6)$$

4.1 Hidden Neurons

The number of Hidden Neurons (2 – 10) for Morning and Evening Load Forecast were found to be of significant importance in overall performance of the network. The results for Morning and Evening Load Forecasts when the algorithm was run with two, four, six, eight and ten hidden neurons are plotted in figures 4 and 5 respectively.

The percentage errors calculated using equation (6) for both Morning and Evening Load Forecasts showed that the condition with two hidden neurons has the lowest error. The values of the percentage errors were found to be 1.68% and 3.90% for both morning and evening forecasts respectively. The corresponding plots and pictorial representation of the target and forecast with 2 hidden neurons are shown in figures 6 (a) & (b) and 7 (a) & (b) respectively.

These results were compared with the results of the average % error calculated from load forecast technique which is presently adopted by PHCN as:

% error calculated =

$$\frac{ActualLoad - PHCNForecast \times 100}{ActualLoad} \quad (7)$$

The percentage errors calculated using equation (7) for both Morning and Evening Load Forecast were 50.56% and 70.42%

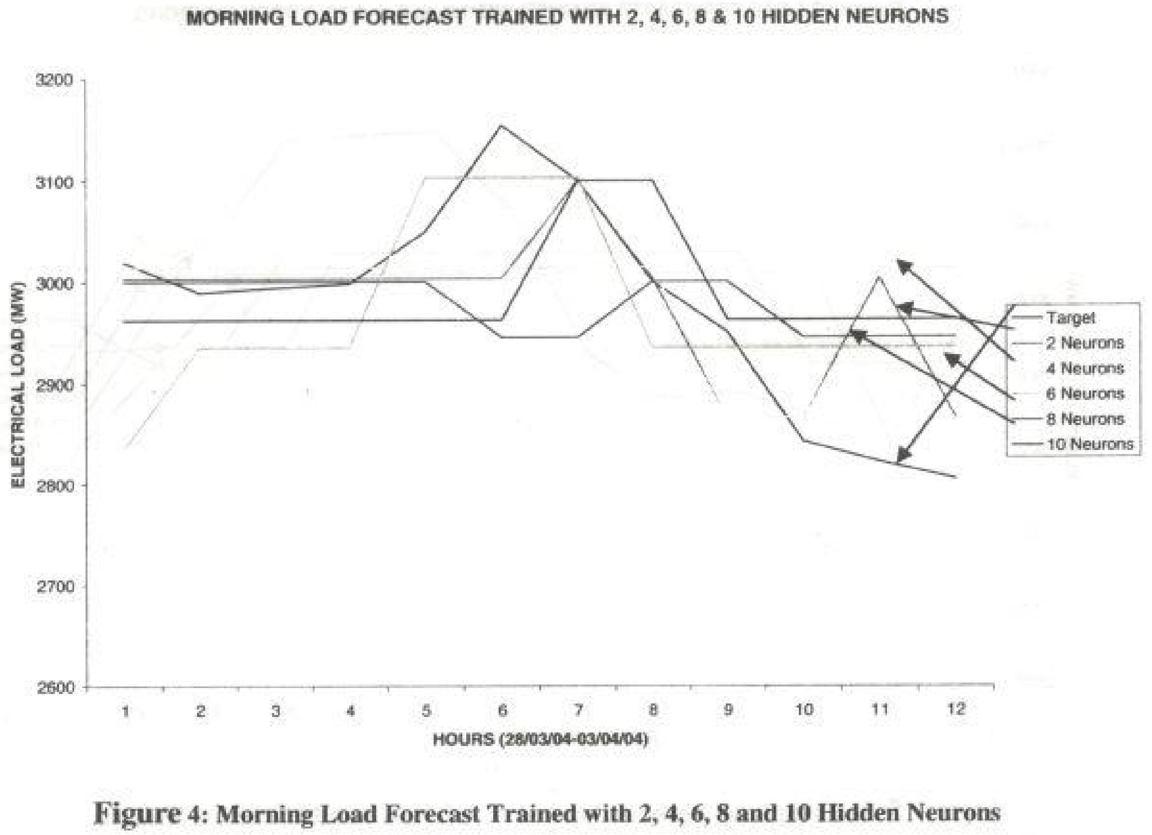


Figure 4: Morning Load Forecast Trained with 2, 4, 6, 8 and 10 Hidden Neurons

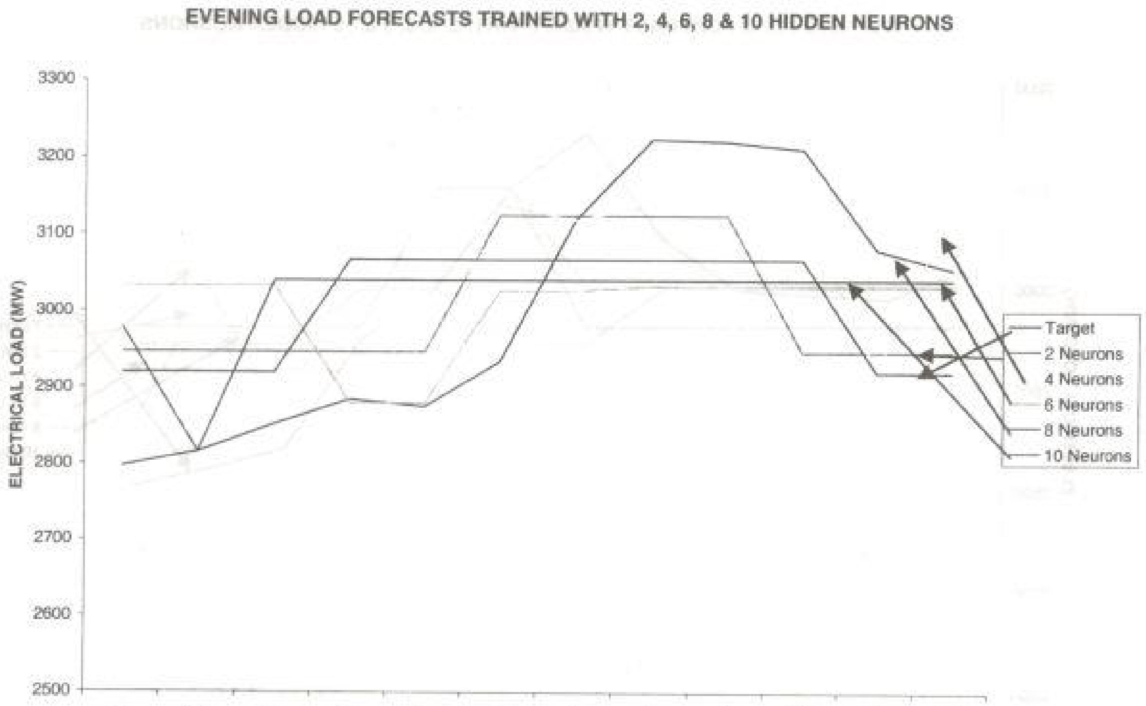
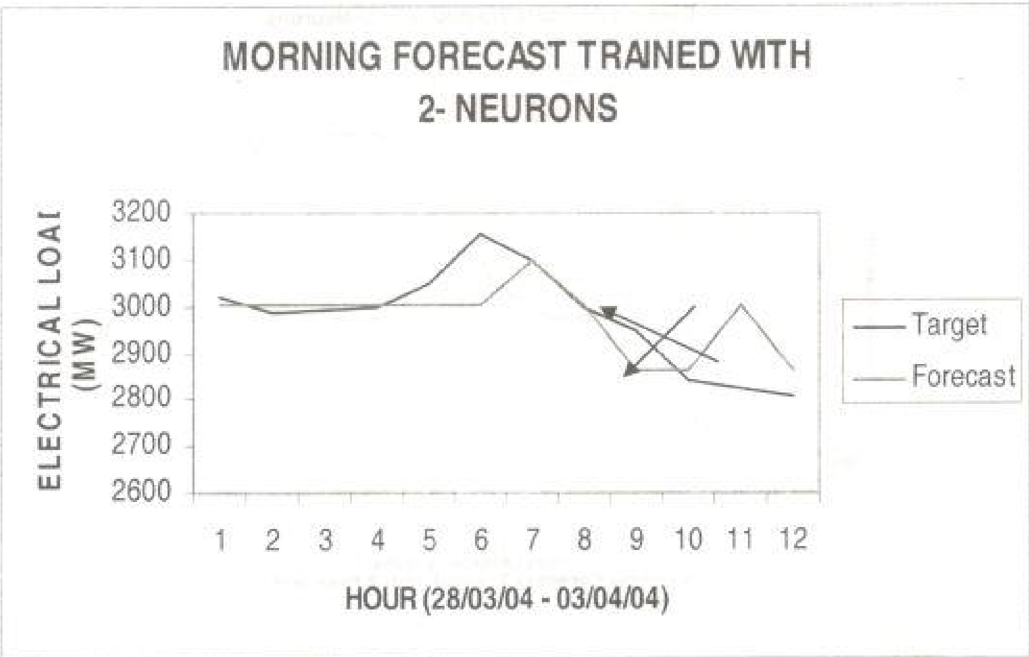


Figure 5: Evening Load Forecast Trained with 2, 4, 6, 8 and 10 Hidden Neurons


MORNING FORECAST TRAINED WITH 2- NEURONS
ELECTRICAL LOAD (MW)
3200
3100
3000
2900
2800
2700
2600
1 2 3 4 5 6 7 8 9 10 11 12
HOUR (28/03/04 - 03/04/04)
Target
Forecast

(a)

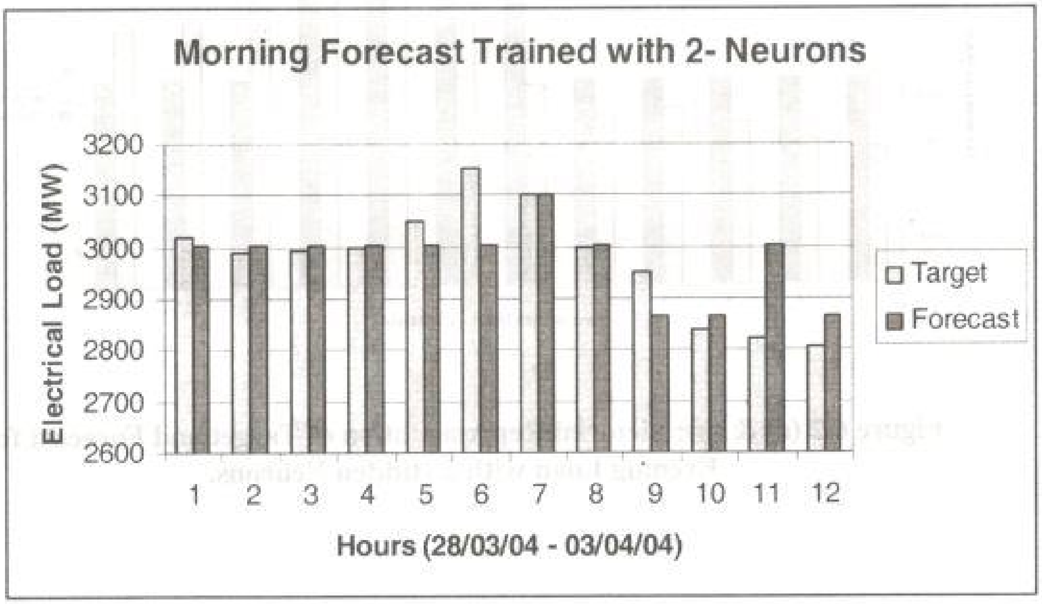

Morning Forecast Trained with 2- Neurons
Electrical Load (MW)
3200
3100
3000
2900
2800
2700
2600
1 2 3 4 5 6 7 8 9 10 11 12
Hours (28/03/04 - 03/04/04)
Target
Forecast

Figure 6.1 (a) & (b): Pictorial Representation of Target and Forecast for Morning Load with 2 Hidden Neurons.

(b)

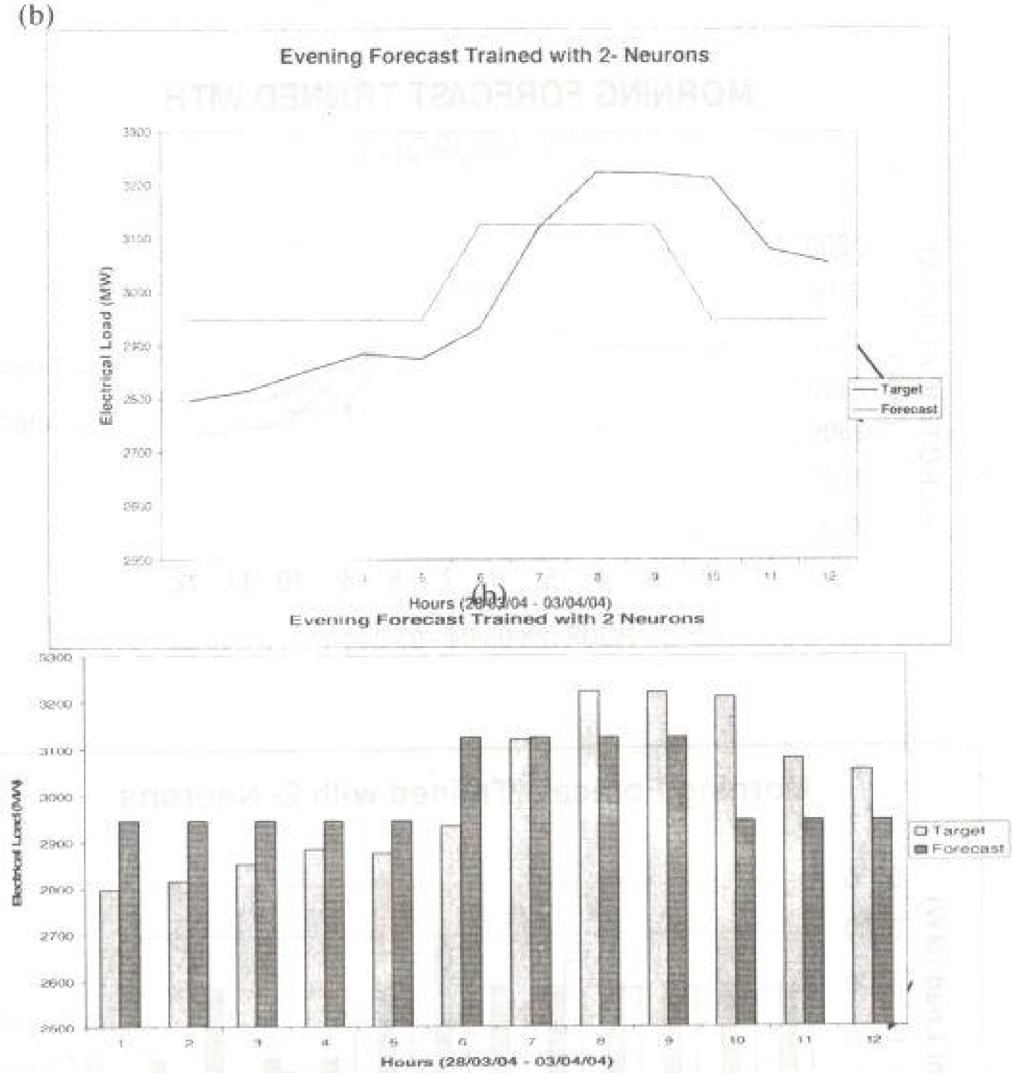


Figure 6.2 (a) & (b): Pictorial Representation of Target and Forecast for Evening Load with 2 Hidden Neurons.

5.0 Conclusion

Matlab-based ANN technique has been successfully employed for Short-Term Load Forecasting using real data obtained from the Power Holding Company of Nigeria (PHCN). The network was trained on the past load and weather records taken from 28th March to 4th April 2008. Load and weather conditions were taken into consideration during the training so that the model could adapt to all conditions. The number of Hidden Neurons for Morning and Evening Load Forecast were found to be of significant importance in overall performance of the network. ANN with Two, Four, Six, eight and ten hidden neurons was tested. The case with two hidden neurons was found to have the least percentage error, being 1.68% and 3.90% for both Morning and Evening forecasts respectively. The percentage errors for Morning and Evening Load Forecasting using PHCN's forecasting technique were 50.56% and 70.42% respectively. Both results show that Short-Term load forecasting employing ANN technique can yield better and more accurate load forecast.

6.0 REFERENCES

1. AlFuhaid A. S, El-Sayed M. A and Mahmoud M. S, "Cascaded Artificial Neural Networks for Short-Term Load Forecasting", IEEE Transactions on Power Systems, Vol. 12, No. 4, November 1997, pp 1524 – 1529.

2. Hippert Henrique Steinherz, Pedreira Carlos Eduardo and Souza Reinaldo Castro, "Neural Networks for Short-Term Load Forecasting: A Review and Evaluation", IEEE Transactions on Power Systems, Vol. 16, No. 1, February 2001, pp 44 – 55.

3. Park, D. C, El-Sharkawi, M.A, Marks II R. J,Atlas, L. E, and Damborg, M. J, "Electric Load Forecasting Using An Artificial Neural Network", IEEE Transactions on Power Systems, Vol. 6, No. 2, May 1991, pp 442 – 449.

4. Lee, K. Y, Cha, Y. T and Park, J. H, "Short-Term Load Forecasting Using an Artificial Neural Network", IEEE Transactions on Power Systems, Vol. 7, No. 1, February 1992, pp 124 - 132.

5. Ho, Kun-Long, Hsu, Yuan-Yih and Yang, Chien-Chuen, "Short Term Load Forecasting Using A Multilayer Neural Network With An Adaptive Learning Algorithm", IEEE Transactions on Power Systems, Vol. 7, No. 1, February 1992, pp 141 - 149.

6. Peng,T. M, Hubele, N. F and Karady G. G, "Advancement in the application of Neural Networks for Short-Term Load Forecasting", IEEE Transactions on Power Systems, Vol. 7, No. 1, February 1992, pp 250 - 256.

7. Bakirtzis, A. G, Theocharis, J. B, Kiartzis, S. J and Satsios, K. J, "Short Term Load Forecasting Using Fuzzy Neural Networks", IEEE Transactions on Power Systems, Vol. 10, No. 3, August 1995, pp 1518 – 1524.

8. Khotanzad Alireza, Hwang Rey-Chue, Abaye Alireza, and Maratukulam Dominic, "An Adaptive Modular Artificial Neural Network Hourly Load Forecaster and its Implication at Electric Utilities", IEEE Transactions on Power Systems, Vol. 10, No. 3, August 1995, pp 1716 – 1722.

9. Lu, C. N, Wu, H. T and Vemuri, S, 1993, "Neural Network Based Short Term Load Forecasting", IEEE Transactions on Power Systems, Vol. 8, No. 1, February 1993, pp 336 - 342.

10. Pauli Murto, 1998, "Neural Network Models for Short-Term Load Forecasting", A Master's Thesis, Helsinki University of Technology.

11. George J. Tsekouras, Nikos D. Hatziargyriou and Evangelous N. Dialynas, 2006, "An Optimized Adaptive Neural Network for Annual Midterm Energy Forecasting" IEEE Transactions on Power Systems, Vol. 21, No. 1, February 2006, pp 385 – 391.

12. Shu Fan and Luonan Chen, 2006, Short-Term Forecasting Based on an Adaptive Hybrid Method", IEEE Transactions on Power Systems, Vol. 21, No. 1, February 2006, pp 392 - 401.

13. Ascar, A. J. R. Mc Donal 1994, "A Specification of Neural Network Applications in the Load Forecasting Problem", IEEE Transaction on Control Systems Technology, Vo1.2 no 2, June 1994, pp 135 -141