Electrical Charge of Niamey City Modelisation by Neural Network

Mamane Moumouni Hamidou¹, Noma Talibi Soumaïla²,*, Boureima Seibou¹, Adekunlé Akim Salami³, Attoumane Kosso Moustapha¹, Madougou Saïdou²

¹Electrical Engineering Department, Mines, Industry and Geology School, Niamey, Niger
²Laboratory of Energy, Electronics, Electrotechnics, Automatics and Industrial Computing, Abdou Moumouni University (UAM), Niamey, Niger
³Electrical Engineering Department, National Engineering School, University of Lome, Lome, Togo

Abstract: In order to forecast consumption, electric power generation, transmission and distribution companies need model to predict short-term demand for electric power load so that they can use their electricity infrastructure efficiently, safely and economically. The short-term forecast of electrical energy demand is the forecast of consumption over time interval ranging from one hour to few days. For optimal use of electricity grid, energy production must keep pace with demand. To this end, prediction errors can lead to risks and shortcomings in the generation and distribution of electrical load to users. This paper is part of electrical charge prediction of Niamey city. Several are being carried out in this field, but prediction techniques based on artificial neural networks have recently been developed. This work focused on two (2) neural approaches such as the multilayer Perceptron (MLP) and the non-linear autoregressive network with exogenous inputs (NARX). Several configurations of these two models have been developed and tested on actual electrical load data. We carried out the short-term forecast (hourly basis) of electrical load of Niamey city. All configurations have been implemented in MATLAB software. The statistical indicators MAPE (Mean Absolute Average Error in Percent), $R^2$ (the correlation coefficient) and RMSE (Square Root of Mean Square Error) were used to evaluate the performance of the models. Thus, with MAPE of 5.1765%, $R^2$ of 95.3013% and RMSE of 5.6014%, the [ABCD] configuration of NARX model converges better compared to the MLP model with MAPE of 7.1874%, $R^2$ of 92.0622% and RMSE of 7.2199%. Where A is the data charge of the same time of the previous day, B is the charge data of the same time of the previous week, C is the charge data of same time of previous year and D is the average of last 24 charge values. So the NARX model is the most efficient and can be used for future predictions on Niamey city network.

Keywords: Short-term Forecast, Artificial Neural Networks, MLP, NARX, MAPE, $R^2$, RMSE
load for good balance of the power system [3].

Predicting the value of the electrical charge is an ideal way to reduce load shedding and ensure good supply of electrical energy [4]. As result, the quality of this forecast, which is essential element of preparation and anticipation, helps to ensure that the production-consumption balance is maintained at all times. It therefore has direct impact on the operational safety of electrical system. The prediction is made with knowledge of users’ consumption over previous years. Electricity consumption depends on activities of users and therefore on their daily, weekly or annual behavior [5]. Depending on this behavior, the load may increase or decrease from one hour to another, from one day to another or from one season to another [6].

Short-term forecasting of electricity consumption plays essential role in efficient management of resources allocated to electricity production. Forecast errors can lead to significant operational costs. The objective is therefore to provide short-term prediction (time horizon) of demand for electrical power. Short-term prediction helps to minimize errors, sources of risk and inadequacies in correct generation and distribution of electrical energy to users. There is a lot of research in this area [3].

Artificial Neural Networks are function estimators. They are considered as configurable black boxes, in order to find link between inputs and outputs through sample of data during the learning phase. In this paper, Artificial Neural Networks are applied for two modelling approaches [7, 8]. For both cases, similar parameters are used. We are talking about the type of network, the activation function and the learning rule.

2. Prediction with the Multi-Layer Perceptron (MLP)

The first model developed in this project is two-layer Perceptron Multi-Layer (MLP) with hidden layer and output layer [9]. This type of network is reliable tool for problems of approximation of functions. The choice of inputs is made using the correlation between the data. The activation function used to activate the neurons in the hidden layer is sigmoid function. The function provides output values belonging to interval [0,1]. For the neurons in output layer, activation function is of linear type. The procedure used for learning phase is error correction procedure (Backward Error Propagation). The principle is easy, we proceed to propagation of error calculated by network from the output layer to the input layer [10, 11]. The algorithm used to update weights is the Levenberg-Marquardt one. Its principle is based on a minimization of function. It calculates cost function, on which it decides whether or not the update will be accepted. It continues the calculation until the network converges. The calculation is done using the Jacobean weights and biases [12-14].

The output of our network is given by equation (1):

\[ y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i \]  

Table 1. Summarizes the different parameters of the selected MLP model.

| Model                        | Perceptron Multilayer (MLP) |
|------------------------------|-----------------------------|
| Number of layers             | 2                           |
| Number of hidden layers      | 1                           |
| Function to activate the neurons in the hidden layer | Sigmoid function |
| Function to activate the neurons of the output layer | Simple linear function |
| Learning algorithm           | Retro propagation of error   |
| Algorithm for updating synaptic weights | Levenberg-Marquardt |

3. Non-linear Autoregressive Network with Exogenous Inputs (NARX)

The recurring network has many applications. It can be used for modeling complex systems. As preacher, he can predict the next value of the output signal. In addition to the same parameters as the first model it has a number of delays [15, 16].

\[ y(t)=f(y(t-1),y(t-2),...,y(t-n_{y}),u(t-1),u(t-2),...,u(t-n_{u})) \]  

Table 2. Summarizes the different parameters of the selected NARX model.

| Model                              | Non-linear autoregressive network with exogenous inputs (NARX) |
|------------------------------------|---------------------------------------------------------------|
| Number of layers                   | 2                                                             |
| Number of hidden layers            | 1                                                             |
| Number of delays (nombre de retards) | 2                                                              |
| Function to activate the neurons in the hidden layer | Sigmoid function       |
| Function to activate the neurons of the output layer | Simple linear function |
| Learning algorithm                 | Retro propagation of error                                    |
| Algorithm for updating synaptic weights | Levenberg-Marquardt              |

4. Experimental Approach to Modelling

The neural network models we have built are two-layer feedforward models for MLP (Figure 1) and NARX for recurrent network (Figure 2). The neurons of the hidden layer have a sigmoid activation function and those of the output layer a linear function in both cases. This architecture is proposed in the Matlab “ntstool” library that we used.
To obtain the different models, the choice and methodical analysis of the explanatory variables is essential. These variables are used to assess the influence of each input parameter on the output of forecast model. Indeed, it is very important, for accuracy of the model, to choose appropriate input parameters. This step is very useful because it eliminates some variables that provide very little or no information to describe the output, or eliminates redundant variables [1, 4].

The variables that were chosen to model the electrical load of Niamey city are listed in Table 3.

| Data types | Mathematical Explanations | Code |
|------------|---------------------------|------|
| Load data from the same time of the previous day | Yh-24 | A |
| Load data from the same time of the previous week | Yh-168 | B |
| Load data from the same time of the previous year | Yh-8760 | C |
| Average of the last 24 hours' charges | $\text{mean}(\sum_{i=1}^{24} Yh_i)$ | D |

It is therefore necessary to predict the electrical charge with various combinations of these explanatory variables in order to determine the most efficient configuration case on the basis of well-defined criterion.

We tested different configuration cases that are summarized in Table 4, for total of 7 configuration cases.

### 5. Results and Interpretations

In this section, the main task is to present the results of research and then to choose the most appropriate model for predicting electrical load based on the MAPE, which is the main indicator chosen to evaluate the performance of these models.

To obtain the different results, programs are designed for each configuration case and for each learning. For given configuration case and neurons number in the hidden layer, each model was run 10 times for the learning and simulation phases.

Indeed, the synaptic weights change values with each execution, giving slightly different results from previous executions.

Neurons number is varied as follows: 10, 20, 30, 40 and 50. Two models, MLP and NARX, are studied. For each model there are seven (7) configuration cases. For each configuration case neurons number is varied in five steps (10, 20, 30, 40, and 50) and for each neurons number considered, ten (10) learning are performed.

In total there are $2 \times 7 \times 5 \times 10 = 700$ learnings, so 700 programs on MATLAB.

The learning time is 12 to 15 minutes for MLP and 3 to 6 minutes for NARX.

For each learning experience it is ensured that the results are automatically recorded by our program. In addition to the end that each learning of the curves is automatically traced also to graphically observe the results.

Finally, for each model and for each number of neurons given, the results of the best performance on the 10 learning outcomes are classified in tables 5 to 14.

#### 5.1. Performances of Perceptron Multilayer Models (MLP)

We have reported the minimum and maximum of considered performance criteria of different cases in tables.
Table 7. MLP performances - configuration [CD]: case_3.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN    |
| 10                                   | 8,5063   | 8,4678   | 7,9891| 90,1846|
| 20                                   | 8,4992   | 8,4652   | 7,9586| 90,1937|
| 30                                   | 10,2247  | 10,1670  | 9,1481| 86,9048|
| 40                                   | 10,1332  | 10,0956  | 9,1050| 87,0371|
| 50*                                  | 10,1168  | 10,0951  | 9,0995| 87,0538|

Table 8. MLP performances - configuration [ABC]: case_4.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN    |
| 10                                   | 8,4102   | 8,0942   | 8,1344| 89,8040|
| 20                                   | 8,0493   | 7,5222   | 8,0767| 89,9564|
| 30                                   | 7,9890   | 7,9361   | 8,0519| 90,0214|
| 40                                   | 7,9585   | 7,9149   | 8,0283| 90,0829|
| 50*                                  | 7,9651   | 7,9044   | 8,0190| 90,1071|

Table 9. MLP performances - configuration [ACD]: case_5.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN    |
| 10                                   | 8,4387   | 8,4020   | 8,3257| 89,2900|
| 20                                   | 8,3858   | 8,3666   | 8,3326| 89,2959|
| 30                                   | 8,3522   | 8,3232   | 8,3228| 89,4083|
| 40                                   | 8,3714   | 8,3054   | 8,2632| 89,4598|
| 50*                                  | 8,3287   | 8,2981   | 8,2500| 89,4953|

Table 10. MLP performances - configuration [BCD]: case_6.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN    |
| 10                                   | 8,2383   | 8,1431   | 7,7597| 90,7689|
| 20                                   | 8,1280   | 8,0758   | 7,6959| 90,9275|
| 30                                   | 8,0957   | 8,0318   | 7,6821| 90,9616|
| 40*                                  | 8,0530   | 8,0036   | 7,6308| 91,0080|
| 50*                                  | 8,0630   | 8,0081   | 7,6297| 91,0995|

Table 11. MLP performances - configuration [ABCD]: case_7.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN    |
| 10                                   | 7,5777   | 7,3877   | 7,4916| 91,6378|
| 20                                   | 7,3600   | 7,1321   | 7,3665| 91,8154|
| 30                                   | 7,2905   | 7,2628   | 7,3091| 91,8908|
| 40                                   | 7,2856   | 7,2152   | 7,2946| 92,0092|
| 50*                                  | 7,2850   | 7,1874   | 7,2041| 92,0622|

5.2. Performance of Non-linear Autoregressive Network Models with Exogenous Inputs (NARX)

The results of different configurations of this model are also presented in tables.

Table 12. NARX performances - configuration [AD]: case_1.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN    |
| 10                                   | 5,5741   | 5,3569   | 5,7758| 94,9960|
| 20                                   | 5,4585   | 5,2688   | 5,7246| 95,0888|
| 30                                   | 5,4469   | 5,2939   | 5,7161| 95,1017|
| 40*                                  | 5,4248   | 5,3123   | 5,7313| 95,0748|
| 50*                                  | 5,4725   | 5,2951   | 5,6998| 95,1304|
### Table 13. NARX performances - configuration [BD]: case_2.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN   | MAX   | MIN   |
| 10                                   | 5.5646   | 5.2267   | 5.9384| 5.6575| 94.9786| 94.7036|
| 20                                   | 5.4748   | 5.3173   | 5.8953| 5.7028| 95.1275| 94.7810|
| 30*                                  | 5.3557   | 5.2443   | 5.7202| 5.6550| 95.2107| 95.0943|
| 40                                   | 5.4181   | 5.2444   | 5.7912| 5.6428| 95.2294| 94.9686|
| 50                                   | 5.4791   | 5.2923   | 5.8346| 5.6911| 95.1453| 94.8909|

### Table 14. NARX performances - configuration [CD]: case_3.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN   | MAX   | MIN   |
| 10                                   | 5.4958   | 5.3378   | 5.8494| 5.7549| 95.0345| 94.8647|
| 20                                   | 5.4316   | 5.2654   | 5.8159| 5.6961| 95.1370| 94.9264|
| 30*                                  | 5.6953   | 5.3018   | 5.9454| 5.7043| 95.1230| 94.6921|
| 40                                   | 5.4395   | 5.2938   | 5.8081| 5.7269| 95.0827| 94.9388|
| 50*                                  | 5.4206   | 5.3438   | 5.7639| 5.7324| 95.0753| 95.0176|

### Table 15. NARX performances - configuration [ABC]: case_4.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN   | MAX   | MIN   |
| 10                                   | 5.6630   | 5.6036   | 6.0510| 6.0218| 94.5495| 94.4959|
| 20                                   | 5.7976   | 5.4336   | 6.0384| 5.9215| 94.7356| 94.5328|
| 30*                                  | 5.5895   | 5.4949   | 6.0086| 5.9083| 94.7583| 94.5734|
| 40                                   | 5.6012   | 5.4868   | 6.0045| 5.9452| 94.6903| 94.5805|
| 50*                                  | 5.6969   | 5.5375   | 6.1065| 5.9559| 94.6718| 94.3895|

### Table 16. NARX performances - configuration [ACD]: case_5.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN   | MAX   | MIN   |
| 10                                   | 5.3849   | 5.3087   | 5.8100| 5.7691| 95.0085| 94.9352|
| 20                                   | 5.4612   | 5.3056   | 5.8231| 5.7229| 95.0899| 94.9130|
| 30*                                  | 5.4193   | 5.2967   | 5.8090| 5.7090| 95.1167| 94.9372|
| 40                                   | 5.5729   | 5.3076   | 5.8784| 5.7470| 95.0478| 94.8167|
| 50                                   | 5.3849   | 5.3087   | 5.8100| 5.7691| 95.0085| 94.9352|

### Table 17. NARX performances - configuration [BCD]: case_6.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN   | MAX   | MIN   |
| 10                                   | 5.4221   | 5.2541   | 5.8240| 5.7016| 95.1270| 94.9104|
| 20                                   | 5.4227   | 5.3532   | 5.7818| 5.7477| 95.0469| 94.9862|
| 30                                   | 5.4607   | 5.1884   | 5.8188| 5.6707| 95.1813| 94.9195|
| 40                                   | 5.4216   | 5.3029   | 5.8056| 5.7182| 95.0979| 94.9431|
| 50*                                  | 5.3888   | 5.2862   | 5.7607| 5.6784| 95.1684| 95.0234|

### Table 18. NARX performances - configuration [ABCD]: case_7.

| Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|--------------------------------------|----------|----------|-------|
|                                      | MAX      | MIN      | MAX   | MIN   | MAX   | MIN   |
| 10                                   | 5.3803   | 5.3233   | 5.8215| 5.7609| 95.0232| 94.9212|
| 20                                   | 5.4484   | 5.1943   | 5.8254| 5.6472| 95.2221| 94.9076|
| 30*                                  | 5.3227   | 5.1765   | 5.7596| 5.6014| 95.3013| 95.0255|
| 40                                   | 5.4562   | 5.2638   | 5.7734| 5.7058| 95.1199| 95.0015|
| 50                                   | 5.4953   | 5.2874   | 5.8101| 5.7544| 95.0393| 94.9359|

### 5.3. Interpretations of Model Performance

The interpretation of the performances of different configurations of priori models has allowed us to identify for each case the neurons number in the hidden layer that gives better results as shown in Table 19 (* indicates the best performances).
Table 19. Models best performances summary.

| Model | Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|-------|--------------------------------------|----------|----------|-------|
|       | MAX | MIN | MAX | MIN | MAX | MIN | MAX | MIN | MAX | MIN |
| MLP   | 1   | 50  | 8,6701 | 8,6260 | 8,5790 | 8,5528 | 88,6599 | 88,5858 |
|       | 2   | 50  | 8,4992 | 8,4652 | 8,0124 | 7,9856 | 90,1937 | 90,1243 |
|       | 3   | 50  | 10,1168 | 10,0951 | 9,1202 | 9,0995 | 87,0538 | 86,9906 |
|       | 4   | 50  | 7,9651 | 7,9044 | 8,0649 | 8,0190 | 90,1071 | 89,9874 |
|       | 5   | 50  | 8,3287 | 8,2981 | 8,2900 | 8,2500 | 89,4953 | 89,3872 |
|       | 6   | 40  | 8,0530 | 8,0036 | 7,6711 | 7,6308 | 91,0880 | 90,9888 |
| NARX  | 1   | 40  | 5,4248 | 5,3123 | 5,7859 | 5,7313 | 95,0748 | 94,9831 |
|       | 2   | 30  | 5,3557 | 5,2443 | 5,7202 | 5,6550 | 95,2107 | 95,0943 |
|       | 3   | 50  | 5,4206 | 5,3438 | 5,7639 | 5,7324 | 95,0753 | 95,0176 |
|       | 4   | 30  | 5,5895 | 5,4949 | 6,0086 | 5,9083 | 94,7583 | 94,5734 |
|       | 5   | 30  | 5,4193 | 5,2967 | 5,8090 | 5,7090 | 95,1167 | 94,9372 |
|       | 6   | 50  | 5,3888 | 5,2862 | 5,7607 | 5,6784 | 95,1684 | 95,0234 |
|       | 7   | 30* | 5,3327 | 5,1765 | 5,7596 | 5,6014 | 95,3013 | 95,0255 |

The final choice of best performance for each model is made using the MAPE indicator and the correlation coefficient $R^2$. The results are shown in Table 20 (* refers to best performance of all models and configurations).

Table 20. Better performance of the different models.

| Model | Case | Number of neurons in the hidden layer | MAPE (%) | RMSE (%) | R (%) |
|-------|------|--------------------------------------|----------|----------|-------|
|       |      | MAX | MIN | MAX | MIN | MAX | MIN | MAX | MIN |
| MLP   | 7    | 50  | 7,2850 | 7,1874 | 7,3041 | 7,2199 | 92,0622 | 91,8678 |
| NARX  | 7    | 30* | 7,3327 | 5,1765 | 5,7596 | 5,6014 | 95,3013 | 95,0255 |

In addition, the MAPE, RMSE and $R^2$ values (tables 21, 22, 23) obtained yield the curves in Figure 3, Figure 4 and Figure 5 as function of neurons number under the hidden layer.

Table 21. MAPE values according to neurons number and case model7.

| Number of neurons in the hidden layer | MAPE(%): MLP, CASE 7 | MAPE(%): NARX, CASE 7 |
|--------------------------------------|----------------------|----------------------|
| 10                                   | 7,5077               | 5,3803               |
| 20                                   | 7,36                 | 5,4484               |
| 30                                   | 7,2905               | 5,3227               |
| 40                                   | 7,2856               | 5,4562               |
| 50                                   | 7,285                | 5,4953               |

Figure 3. Evolution of MAPE as function of neurons number in the case model7.

Table 22. RMSE values according to neurons number and the case model7.

| Number of neurons in the hidden layer | RMSE(%): MLP CAS 7 | RMSE(%): NARX CAS 7 |
|--------------------------------------|---------------------|---------------------|
| 10                                   | 7,4916              | 5,8215              |
| 20                                   | 7,3865              | 5,8254              |
| 30                                   | 7,3091              | 5,7596              |
| 40                                   | 7,2946              | 5,7734              |
| 50                                   | 7,3041              | 5,8101              |

Figure 4. Evolution of RMSE as a function of neurons number and the case model7.

Table 23. $R^2$ values according to neurons number and the case model7.

| Number of neurons in the hidden layer | $R^2$(%): MLP CAS 7 | $R^2$(%): NARX CAS 7 |
|--------------------------------------|---------------------|---------------------|
| 10                                   | 91,6378             | 95,0232             |
| 20                                   | 91,8154             | 95,2221             |
| 30                                   | 91,8908             | 95,3013             |
| 40                                   | 92,0092             | 95,1199             |
| 50                                   | 92,0622             | 95,0393             |
Figure 5. Evolution of $R^2$ as function of neurons number and the case model 7.

It has been shown that layer networks offer poor results for neurons number in the hidden layer that are insufficient or too large. Thus, figures 3, 4 and 5 analysis shows that neurons number in hidden layer of each model influences results. Indeed, for each case, the best results are obtained for models with 30, 40 or 50 neurons. Beyond 50 neurons, MAPE errors increase and the simulation time is very long, which forces us to limit the number of neurons for our tests.

6. Conclusion

Objective of this work is to develop a model for predicting electrical charge of Niamey city using artificial neural networks. To achieve this goal, two prediction models were tested: MLP and NARX. Several configurations of the two major models mentioned above have been developed and tested by varying the different explanatory variables. All configurations have been implemented on MATLAB. The statistical indicators MAPE (Absolute mean error in percent), $R^2$ (correlation coefficient) and RMSE (square root of mean square error) were used to evaluate performance of models. Thus with MAPE of 5.1765%, $R^2$ of 95.3013% and RMSE of 5.6014%, the [ABCD] configuration of the NARX model is chosen ahead of MLP with MAPE of 7.1874%, $R^2$ of 92.0622% and RMSE of 7.2199%. So the NARX model is the most efficient and can be used for future predictions on the Niamey city network.

References

[1] Binod Bhandari, Shree Raj Shakya, Ajay Kumar Jha, “Short Term Electric Load Forecasting of Kathmandu Valley of Nepal using Artificial Neural Network”, Kathford Journal of Engineering and Management, Volume 1, Issue 1, ISSN: 2661-6106, pp. 43-48, 2018.

[2] Rahul SG, Monah Pradeep I, Kavitha P, Dhivyasri G, “Prediction of Electricity Load Using Artificial Neural Network for Technology Tower Block of VIT University”, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 18, pp. 7717-7723, 2017.

[3] TCHONDA Malanguihewa, neural approach for the prediction of electrical charge on the grid of the Communauté Electrique du Benin (CEB), December 2016; 142p.

[4] Isaac Samuel, Tolulope Ojewola, Ayokunle Awelewa, and Peter Amaize, “Short-Term Load Forecasting Using The Time Series And Artificial Neural Network Methods”, IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE), e-ISSN: 2278-1676, p-ISSN: 2320-3331, Vol. 11, Issue 1, pp. 72-81, 2016.

[5] Anil Patel, Albert John Varghese, “Evaluate Hourly based Load Forecasting using NARX Neural Network in MATLAB Environment”, IJAREEIE, Vol. 6, Issue 3, March 2017.

[6] S. TATSA, ”Modeling and Forecasting of Hourly Electricity Consumption in Québec: Comparison of Time Series Methods,” Master’s Thesis in Econ. Laval University, Québec, Canada, pp. 1-10, 2013. Control, Signals, and Systems, Vol. 2, 1989, pp. 303-314.

[7] M. A. Arbib, The Hand Book of Brain Theory and Neural Networks, 2nd edition, 2002.

[8] L. P. J. Veenenturf, Analysis and Applications of Artificial Neural Networks, Book Prntice Hall Edition, 1995.

[9] Anamika Singh, Vinay Kumar Tripathi, “Load Forecasting Using Multilayer Perceptron Neural Network”, IJESC, Volume 6 Issue No. 5, pp. 4548-4551, May 2016.

[10] Mohsen Hayati, and Yazdan Shirvany, “Artificial Neural Network Approach for Short Term Load Forecasting for Illam Region”, IJCESE VOLUME 1 NUMBER 2, ISSN 1307-5179, pp. 121-125, 2007.

[11] Leo Dencelin X, Ramkumar T, “Analysis of multilayer perceptron machine learning approach in classifying protein secondary structures”, Biomedical Research; Special Issue: S166-S173, 2016.

[12] E. L. TAYLOR, “Short-term Electrical Load Forecasting for an Institutional/Industrial Power System Using an Artificial Neural Network,” Tennessee Res. Creat. Exch., pp. 1–97, 2013.

[13] Haykin S., “Neural Networks-A Comprehensive Foundation”, Prentice-Hall, New York, 1999.

[14] Cybenko G. "Approximation by Superpositions of a Sigmoidal Function", Mathematics of Control, Signals, and Systems, Vol. 2, 1989, pp. 303-314.

[15] Esra Eri¸sen Cem Iyigun Fehmi Tanrısever, "Short-term electricity load forecasting with special days: an analysis on parametric and non-parametric methods", Springer Science+Business Media, LLC, part of Springer Nature 2017.

[16] Abdullateef Ayodele Isqeel, Tijani Bayo Ismaeel, and Salami Momoh-Jimoh Eyiomika, “Consumer Load Prediction Based on NARX for Electricity Theft Detection”, IEEE-2016 International Conference on Computer & Communication Engineering, DOI 10.1109/ICCCE.2016.70, pp. 294-299, 2016.