American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-6, Issue-9, pp-155-165
www.ajer.org
Research Paper Open Access

Predictive and Comparative Analysis of NARX and NIO Time Series Prediction

Omolaye O. Philip[1]; Badmos T. Adeleke[2]

[1]*Department of Electrical and Electronics Engineering, University of Agriculture, P. M. B 2373, Makurdi, Nigeria.*

[2]*Material Engineering Research Institute, Art Computing Engineering Science, Sheffield Hallam University, Sheffield, United Kingdom*

Corresponding Author: Omolaye O. Philip

ABSTRACT: *Telecommunication is one of the most unavoidable utilities of daily life of human being. It provides data records of its services from call logs, port activities to internet usage or consumption. This paper investigates and explored the methodologies for modeling, simulation and controls in ANN based of time series application of telecommunication. To show and prove efficiency, simulated and operational data sets are employed to demonstrate the capability of neural networks in capturing complex nonlinear dynamics where NARX and NIO models are set up to explore and compare both steady-state and transient features on daily internet usage activities. The structures were configured, generated and run in MATLAB to create and train platform, validation, testing and results demonstrate that the techniques can be applied accurately which means that both models successfully capture dynamics of the system up to a certain degree of acceptance. The related parameters for the design and simulation are tuned and set up according to the requirements which show that ANN can perform even better than conventional methods. Finally, it was deduced that NARX model outperform more than the NIO model.*

Keyword*: NARX, NIO, Neural network, Time series prediction, Regression, Levenberg-Marquardt*

Date of Submission: 11-08-2017 Date of acceptance: 13-09-2017

I. INTRODUCTION

Telecommunication is one of the most unavoidable utilities of daily life of human being. It provides data records of its services from call logs, port activities to internet usage or consumption. Prediction is a kind of dynamic filtering where past values of one or more time series such as Nonlinear Autoregressive Network with Exogenous Inputs (NARX), Nonlinear Autoregressive (NAR), Recurrent Neural Networks (RNN), and The Nonlinear Input- Output (NIO) are either employed to predict future values. The dynamic RNN, NAR, NARX and NIO are all neural network structures that can be useful but have distinct merits and demerits depending on the type of application to be used (Cao et al., 2012; Mohanty et al., 2015). This is because they can accept dynamic inputs represented by time series sets, forms the major component include tapped delay lines for nonlinear filtering and prediction. Predictive models also known as dynamic models are of great import for analysis, simulation, monitoring and control of a variety of applications in robotics, manufacturing systems, chemical processes and aerospace systems. A professional such as financial analyst, biologist, geologist and engineer tend to discuss some common factor of interest by having a better understanding of past values to predict the future (Omolaye et al., 2015). The numerous intelligent applications for prediction with the likes of financial analyst to predict the future value of a financial instrument of stock, dividend or bond; an engineer might be interested in the signal strength evaluation and prediction of a particular sub-station or to predict the impending failure of a jet engine; a biologist might be interested to predict the population of an amphibian species in the forests and on and on. All these professionals must work with data to support their design or model propounded in one way or the other. In this research work, the NIO and NARX neural network was focused on to predict internet optimization. Our goal is to provide a methodology framework to analyze the time series historical data of internet usage, and observe if the mid-term prediction of such can be achieved with the models by turning the number of neurons and delays or require further external variables to increase the prediction accuracy or performance.

II. MOTIVATION

The motivation comes from the fact that most time series approach usually contains some unwanted characteristics of high noise and non-stationary that tend to make the classical statistical system not competent and intelligent as such. It is in this view that, we try to adopt an advanced technique such as NARX and NIO to salvage the shortcomings of classical time series methods. Therefore, in this study, neural network based NARX and NIO are employed as forecasting tool.

III. LITERATURE REVIEW

There are numerous research literature related to time series prediction with multitude of models, techniques and methodologies in different and related field of studies. The statistical methods such as autoregressive conditional heteroskedasticity (ARCH), autoregressive integrated moving average model (ARIMA) or Box-Jenkins model, and Smooth Transition Autoregressive (STAR) model are used for prediction of time series data but processes involved could not adequately handle a noisy and nonlinear data (Lahmiri, 2011). Therefore, soft computing techniques such as fuzzy system, genetic algorithm, and artificial neural networks were largely adopted for prediction (Omolaye et al., 2017). Artificial Neural networks (ANNs) coupled with additional flavours are seen to be of one of the most popular techniques based on excellent results and successes recorded in real application and hybridization techniques (Doucoure et al., 2016; Omolaye et al., 2017). Siti and Antoni, (2012) buttress point to report ANN in the area of time series patterns, nonlinear characteristics, and better accuracy over the others techniques. Vadiraj and Narahari (2014) presented and apply linear regression analysis to estimation of average revenue per user in telecom service. Parsapoor and Bilstrup (2013) proposed chaotic time series prediction using brain emotional learning based recurrent fuzzy system and recorded a high accuracy results. Miranian and Abdollahzade, (2013) developed a forecasting model of local least-squares support vector machines-based neuro-fuzzy Model for nonlinear and chaotic time series prediction. Time-series prediction model was proposed by Manish and Salankar (2015) for the packet loss rate (PLR) which is helpful in congestion control mechanisms. Parveen and Srivastava (2016) investigated the import of applicability and capability of ANFIS techniques for time series analysis of network traffic response data prediction. All these numerous predictive models designed or proposed by researchers had addressed high number of real world problems in one way or the other. The predictive flowchart diagram for both NARX and NIO is shown in fig 1.

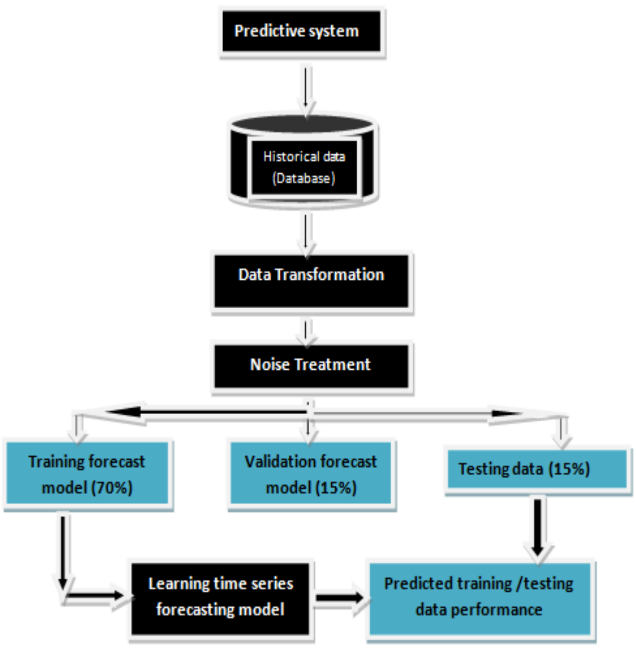

Fig. 1: Predictive Flowchart Diagram for both NARX and NIO

IV. OVERVIEW OF NIO AND NARX

Time series are found to be focused of several overlapping disciplines of information theory, dynamical systems theory and digital signal processing. A system or model is said to be linear if all objective functions and constraints of the system are represented by linear equations. Otherwise, it is considered to be known as a nonlinear model (Ljung and Glad, 1994). However, the description of time series is analyzed either discrete or continuous phenomena; deterministic or stochastic, depend on condition.

The Nonlinear Input- Output (NIO) time series problem is defined as

$$y(t) = f\big(x(t-1), \dots, x(t-d)\big) \qquad (1)$$

This is has a striking similarity with NARX, because it involves two series of an input series *x(t)* and an output/target series *y(t)*. The interpretation of eqn. 1 gives insight on predicting values of *y(t)* from previous values of *x(t)* without prior knowledge of previous values of *y(t)*. The snapshot of NIO network is shown in fig. 2 with tapped delay lines and a layer feedforward network, a sigmoid transfer function in the hidden layer and a linear transfer function in the output layer.

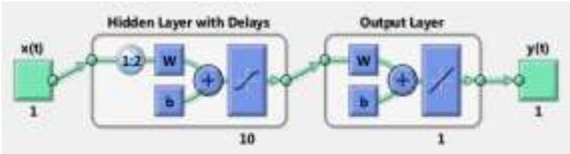

Fig. 2: depicted the standard NARX network

Nonlinear Autoregressive Network with Exogenous Inputs (NARX) is a recurrent dynamic network, with feedback connections enclosing several layers of the network (Caswell, 2014). The snapshot of NARX network is shown in fig. 3 with tapped delay lines and two-layer feedforward network, a sigmoid transfer function in the hidden layer and a linear transfer function in the output layer.

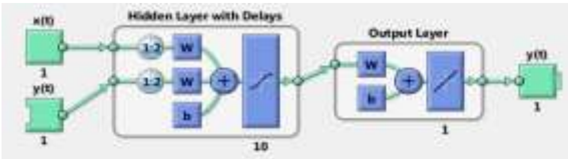

Fig. 3: depicted the standard NARX network

By considering a scalar (or vector) with time *t* and output *P*

$$\{\, x(t_0), x(t_1), \dots x(t_{i-1}), x(t_i), x(t_{i+1}), \dots \} \qquad (2)$$

t is known to be real-valued while *x(t)* is a continuous signal. The discrete signal can be deduced by uniform sampling as shown in eqn. (3) where sampling period Δt is introduced according to the Nyquist sample theorem (Levesque, 2014).

$$\{x[t]\} = \{x(0), x(\Delta t), x(2\Delta t), x(3\Delta t), \dots\} \qquad (3)$$

Estimation of *x* at some future time

$$\hat{x}[t+s] = \ f(x[t], x[t-1], x[t-2], \dots) \qquad (4)$$

where *s* is called the horizon of prediction. In a situation where *s* is set the value of 1 (*s* = 1), then it is termed as *one time step* ahead prediction; otherwise, it is called *multi-step* ahead prediction.

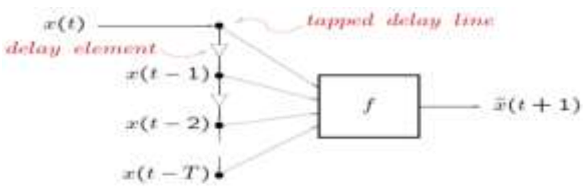

Fig. 4: Employ the past to predict the future

A typical NARX problem of prediction is defined in eqn. 5, which can be further be interpreted as predict series output or target, *y(t)* given that *d* stores past values of *y(t)* and another series of input, *x(t)*

$$y(t) = \ f\big(y(t-1), \dots, y(t-d), x(t-1), \dots, (t-d)\big) \qquad (5)$$

where *d* is the tapped delay lines to store previous values of the input, *x(t)* and output or target, *y(t)* sequences.

V. METHODOLOGY

Levenberg-Marquardt (LM) Optimization was employed because it from the Newton family methods of second order, which is a virtual standard in nonlinear optimization that works with only function evaluations and gradient information. It estimates the Hessian matrix using the sum of outer products of the gradients as shown in eqn. (6).

$$W_k = W_{k-1} - [H_{k-1} + \lambda_{k-1}.I]^{-1}.j_{k-1} \qquad (6)$$

In this research work, random data division using *dividerand* command, training with LM (*trainlm*) and Mean squared error (*mse*) were employed. The input vectors and target vectors randomly divided into three set values of training, validation and testing with assigned values of 70%, 15% and 15% respectively. This is amounted to 949 data sample for training, 204 each for validation and testing. The number of neurons and delays were tuned up to further vary the performance of the network at 5, 10, 15, 20; and 1, 2, 3 4 respectively. The network stops when the target MSE was achieved or after the maximum number of epochs was reached.

Therefore, the matrix m by n with m rows and n columns is called the size of the matrix. Thus matrix is of the form

$$A = [a_{ij}] = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ . & . & \cdots & . \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix} \qquad (7)$$

A column vector is of the form

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \qquad (8)$$

VI. RESULTS AND DISCUSSIONS

This section describes the entire series of tests performed, simulation output and other experimental set-up

A. SIMULATED NARX RESULTS

The simulated NARX results with various varying parameters are presented in Table 1 to Table 4 and figure 5 to figure 11. The LM algorithm converged upon a solution after maximum 12 iterations with no significant error crosscorrelation or autocorrelation issues identified. Therefore, there are highly significant ($p < 0.001$) correlations between output and target data at good fit (R values that greater than 0.9).

Table 1: The result of NARX with parameter: n =5, d = 1 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.55746e-2	9.64893e-1	12	00.00.00	0.0155	0.00012	1.00e-05	6
Validation	2.79891e-2	9.34701e-1						
Testing	1.71727e-2	9.59978e-1						

Table 2: The result of NARX with parameter: n =10, d = 2 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.18511e-2	9.73864e-1	9	0.00.01	0.0115	0.00397	1.00e-5	6
Validation	2.20784e-2	9.38266e-1						
Testing	4.14051e-2	9.13119e-1						

Table 3: The result of NARX with parameter: n =15, d = 3 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.47505e-2	9.68085e-1	10	0.00.01	0.0121	0.00628	1.00e-05	6
Validation	3.38421e-2	9.26714e-1						
Testing	1.70682e-2	9.63112e-1						

Table 4: The result of NARX with parameter: n =20, d = 4 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.51303e-2	9.65287e-1	11	0.00.03	0.0121	0.0395	1.00e-05	6
Validation	1.86372e-2	9.62154e-1						
Testing	2.18148e-2	9.52936e-1						

The graphical representations of the simulation of NARX are depicted in figure 5 to figure 11.

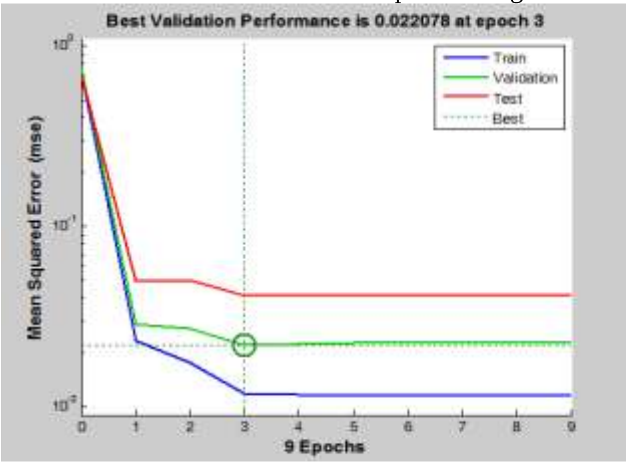

Fig 5: Performance plot of NARX (*plotperform*)

Fig. 5 shows that there are decreases in errors in training, validation and testing until iteration 10 is attained which depicted that there is no element of occurrence of over fitting. The training, validation and testing are done in open loop likewise, the R values which are also computed based on the open-loop training results.

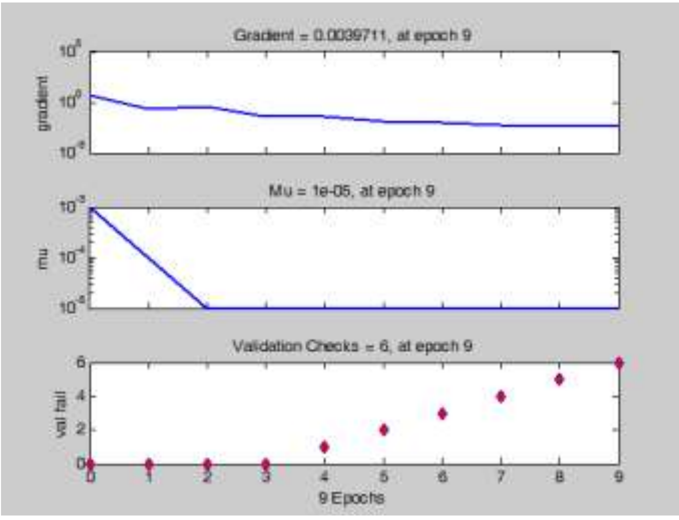

Fig. 6: Training state of NARX (*plottrainstate*)

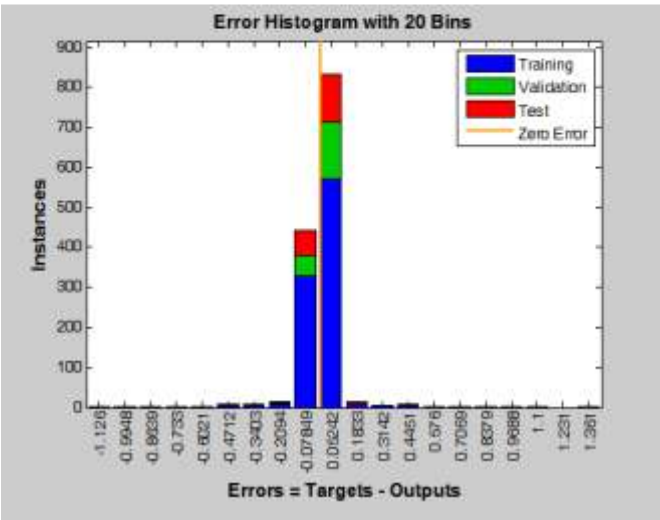

Fig. 7: Error histogram of NARX (*ploterrhist*)

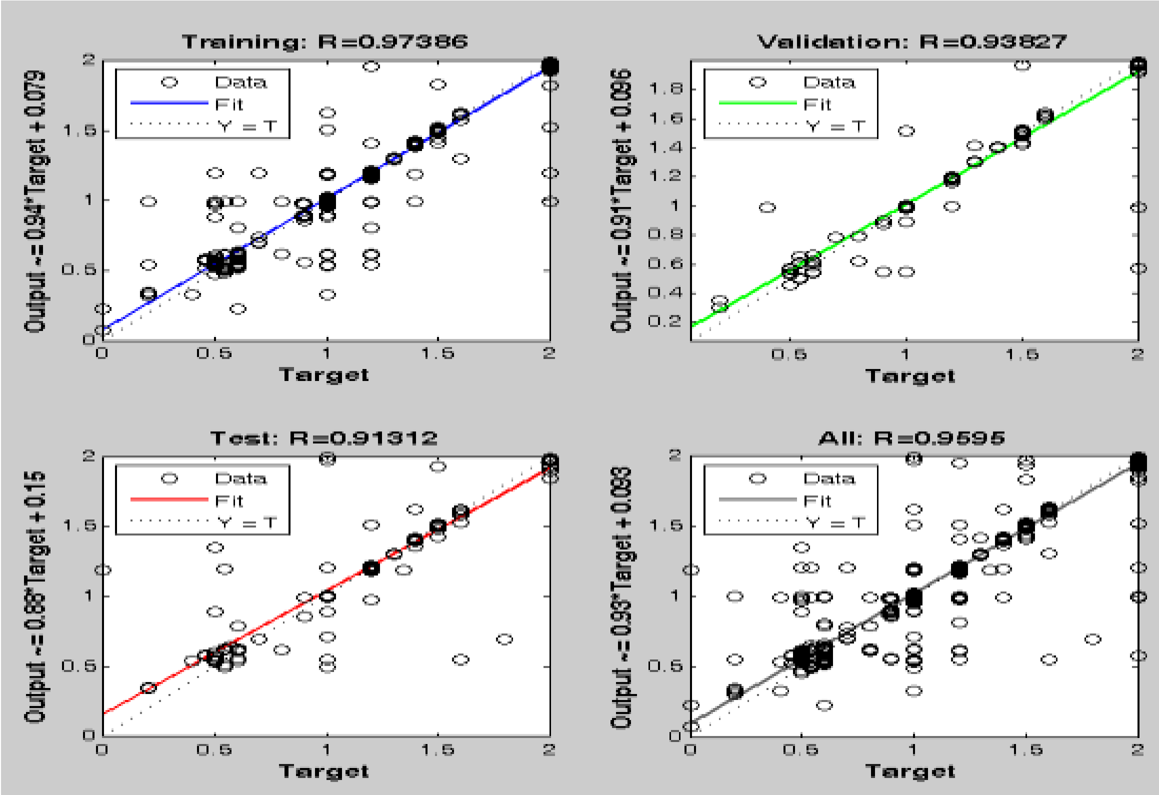

Fig. 8: Regression of NARX plot (*plotregression*)

Fig 8 showed the regression of NARX plot with four different plots represent the training, validation, and testing and output data. This scatter plot of NARX is helpful in showing that certain data points have good or poor fits. The solid straight lines represent the best fit linear regression line between outputs and targets of training (blue), validation (green), testing (red) and output of all (black) while the dashed line in each plot represents the perfect result – outputs = targets. The regression (R) value indicates the relationship between the outputs and targets. Actually, If R is close to zero means there is no linear relationship between outputs and targets but, if R = 1, then there is an indication that there is an exact linear relationship between outputs and targets. In this research work, it was shown that, the training data (R=0.97386) indicates a good fit, validation (R=0.93827) and test (0.91312) results also show R values that greater than 0.9.

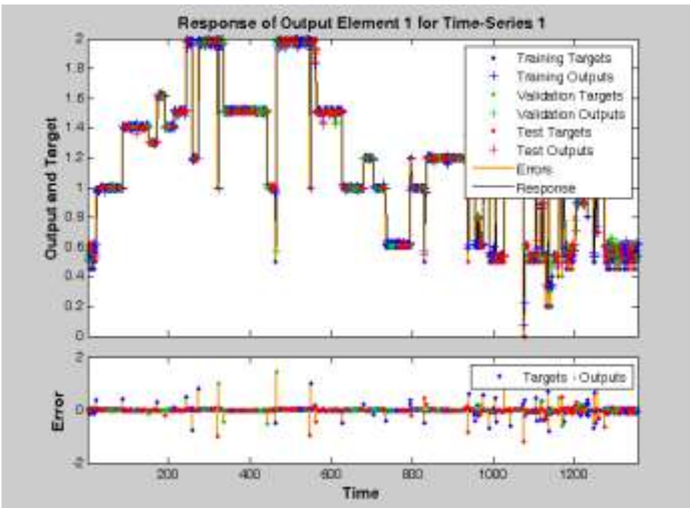

Fig. 9: Time-series response of NARX (*plotresponse*)

Fig. 9 shows the Time Series Response of NARX which gives a clear indication where time points were selected for training, testing and validation. Time Series Response helps in displaying of the inputs, targets and errors versus time of a given problem.

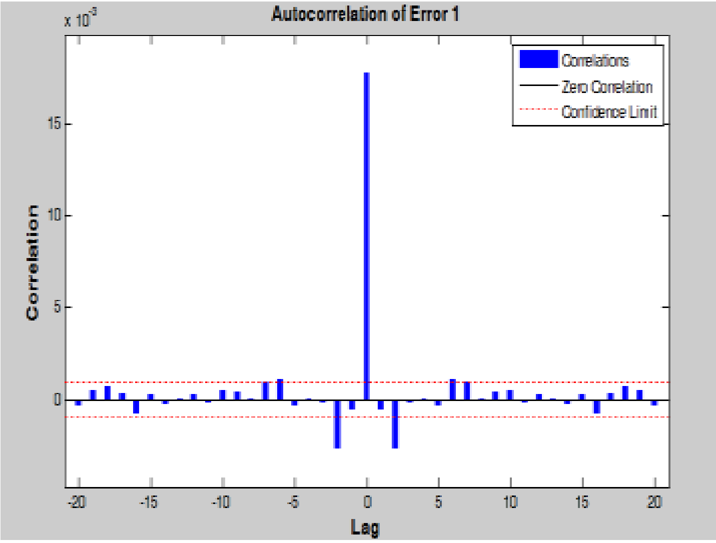

Fig. 10: Error autocorrelation of NARX (*ploterrcorr*)

Fig. 10 shows the error autocorrelation of NARX using error autocorrelation function to validate the performance of the trained network to gives description on how the prediction errors are related in time. It was observed that there was significant correlation in the prediction errors, the most of the trained network falls within the red confidence limits, which can create rooms for possible improvement (maybe by increasing the number of neurons and delays further). If the network has been trained well all the other lines will be much shorter, and most if not all will fall within the red confidence limits

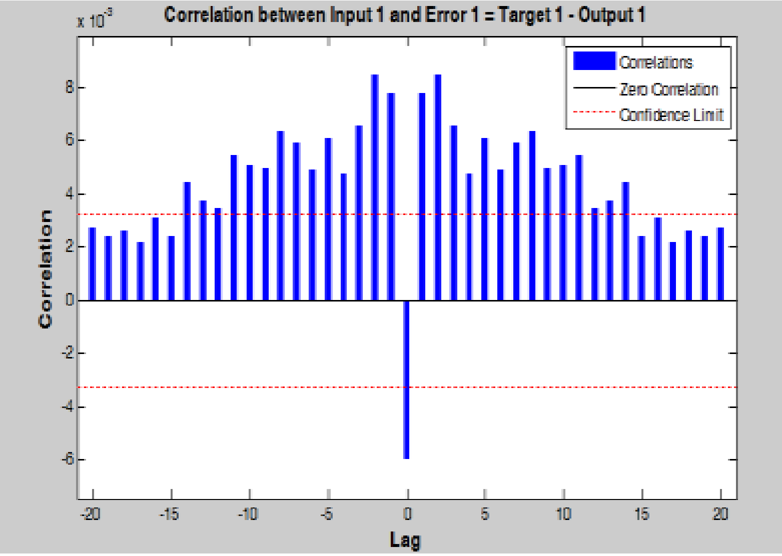

Fig. 11: Input-Error Cross-Correlation of NARX (*plotinerrcorr*)

Fig 11 shows the input-error cross-correlation function that assist in to obtaining additional verification of network performance in the sense that it investigates how the errors are correlated with the input sequence *x(t).* It was observed that the input is correlated with the error which also indicated that there is room for prediction improvement by increasing the number of delays in the tapped delay lines

B. SIMULATED NIO RESULTS

The simulated NIO results with various varying parameters are shown in Table 5 to Table 8. Again, the LM algorithm converged upon a solution after the maximum of 10 iterations with no significant error crosscorrelation or autocorrelation issues identified. Therefore, there are highly significant ($p < 0.001$) correlations between output and target data at bad fit (R values is far lesser than 0.9).

Table 5: The result of NIO with parameter: *n* =5, *d* = 1 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.14762e-1	6.85351e-1	8	00.00.02	0.0114	0.000126	1.00e-06	6
Validation	1.55331e-1	6.21594e-1						
Testing	1.19218e-1	6.97511e-1						

Table 6: The result of NIO with parameter: *n* =10, *d* = 2 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.13444e-1	7.04550e-1	8	0.00.05	0.0113	0.00208	0.0001	6
Validation	1.11942e-1	7.26325e-1						
Testing	1.04210e-1	7.01761e-1						

Table 7: The result of NIO with parameter: *n* =15, *d* = 3 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.04122e-1	7.41547e-1	10	0.00.13	0.101	0.00693	1.00e-05	6
Validation	1.08582e-1	6.95327e-1						
Testing	1.06980e-1	6.86780e-1						

Table 8: The result of NIO with parameter: *n* =20, *d* = 4 (Train and Retrain)

	MSE	R	Epoch	Time	Performance	Gradient	MU	Validation check
Training	1.12107e-1	7.33226e-1	9	0.00.17	0.0994	0.0200	0.0001	6
Validation	1.12248e-1	6.86381e-1						
Testing	9.76763e-2	7.27694e-1						

The graphical representations of the simulation of NIO are showcased in figure 12 to figure 15

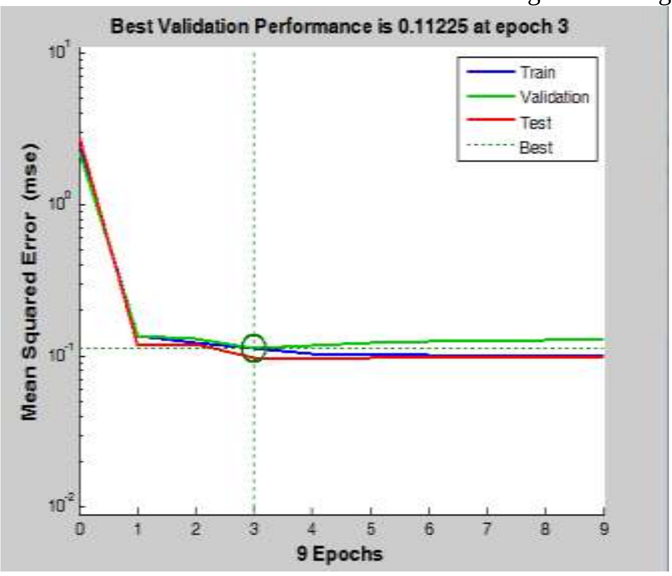

Fig 12: Performance plot of NIO (*plotperform*)

Fig. 12: shows the performance plot of NIO with a problem of overfitting with the training. It was observed that the validation and test curves are similar and the test curve had increased tremendously before the validation curve increased, therefore, overfitting is suspected

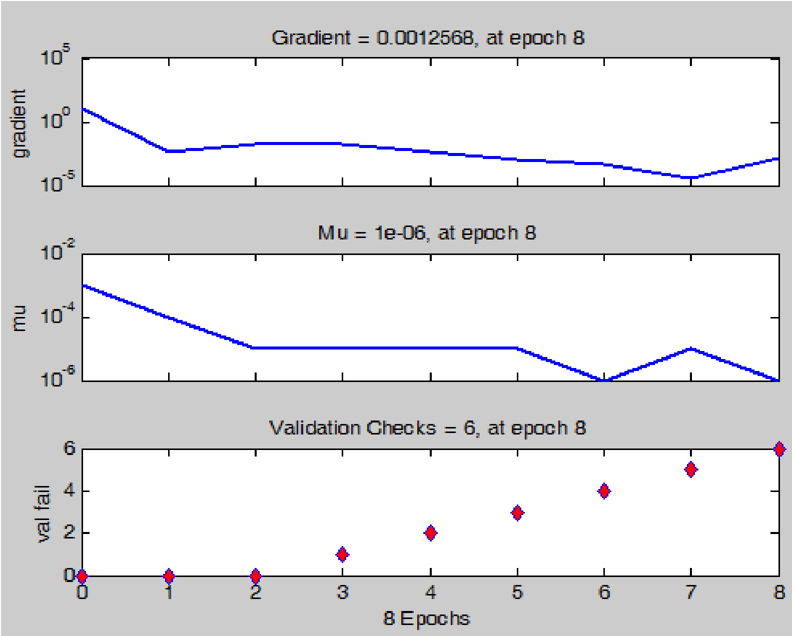

Fig. 13: Training state of NIO (*plottrainstate*)

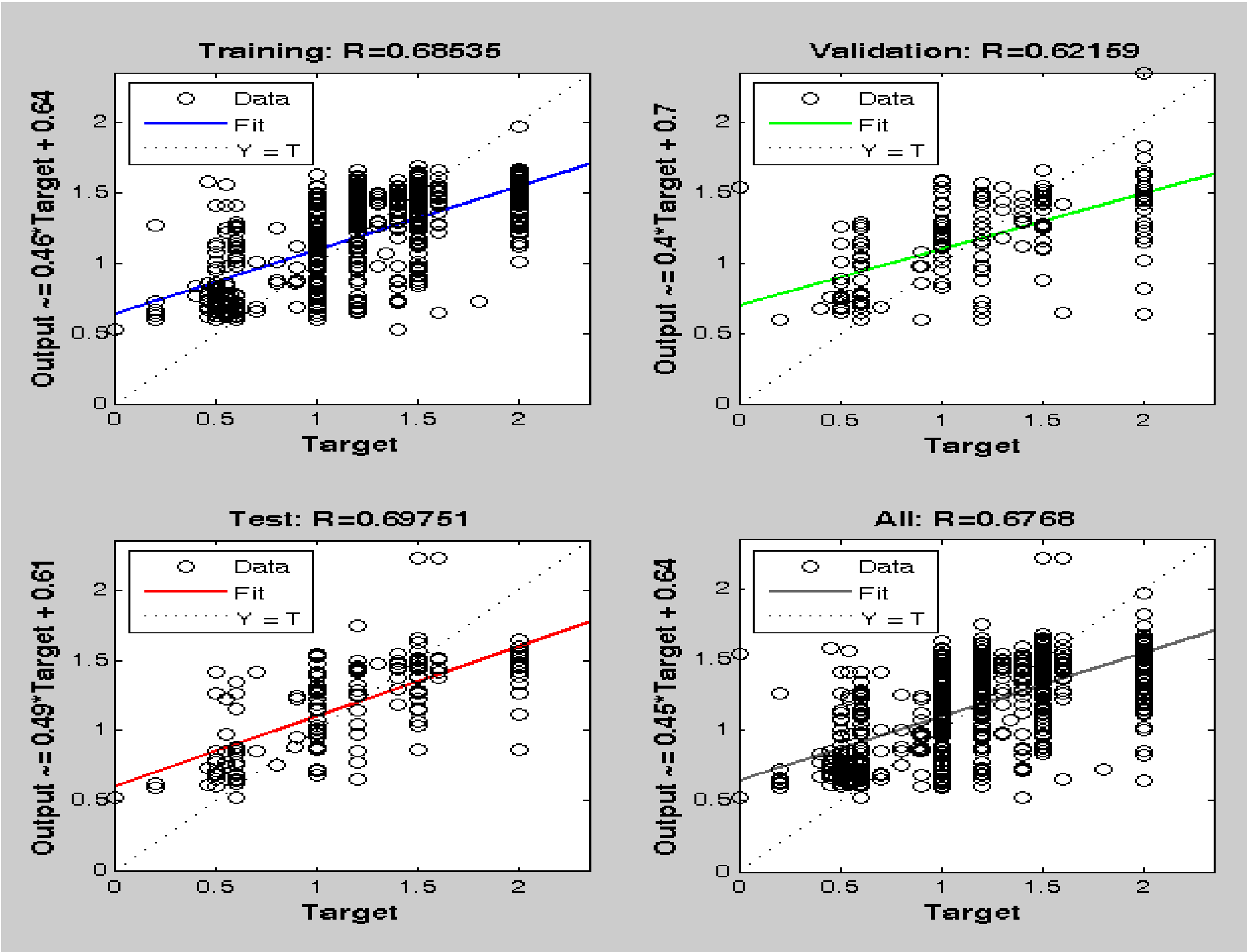

Fig. 14: Regression of NIO plot (*plotregression*)

Fig 14 showed the regression of NIO with scatter plots that helps in showing certain data points with good or poor fits. In this research work, it was shown that, the training data (R=0.68535) indicates a very bad fit, validation (R=0.62159) and test (0.69751) results also show R values that less than 0.9.

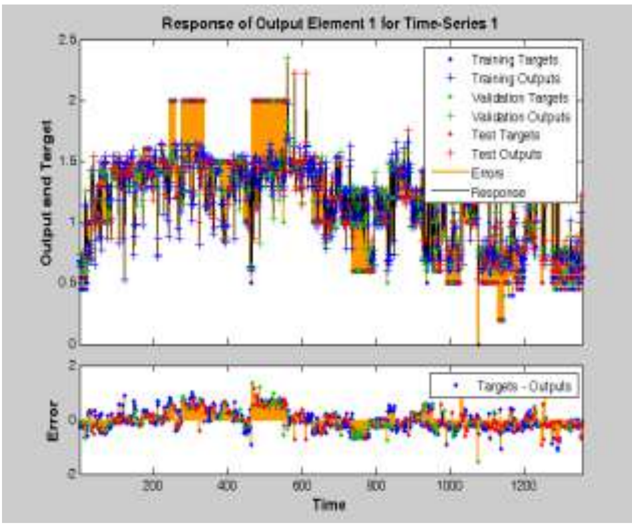

Fig. 15: Time-series response of NIO (*plotresponse*)

C. Average Accuracy (NARX and NIO)

Overall measures of performance error (target-output) were obtained and summarized in column 2 of Table 1 through Table 8 for both models (MSE). These results also suggest that NARX model produces a greater predictive capacity for both fit and accuracy while NIO yield a moderate predictive capacity for accuracy but bad fit.

VII. CONCLUSION

We have presented models based on ANN concepts for telecommunication network services and evaluate merits and demerits of the models in question through results comparison. Overall, in spite of all the controversial issues regarding ANN, the research work found that NARX model possesses high and strong potential to be considered as a reliable alternative to the conventional techniques. The NIO results were not close with NARX may be some applications in which the previous values of *y(t)* would not be available. However, those are the only cases where you would want to use the input-output model instead of the NARX model. The NARX model discussed and investigated in this research, had provided better predictions than NIO model, this is owning to the fact that NARX employs the use of additional information contained in the previous values of *y(t)*. In conclusion, it was observed that both NARX and NIO can effectively learn complex sequences and outperform some well-known, existing models but NARX is better off than NIO of the same class.

FUTURE DIRECTION

LM (*trainlm*) is frequently employed and recommended for most problems, but for situation where some occurrence of noisy and small problems, Bayesian Regularization (*trainbr*) can be employed but takes longer time to proffer better solution. It is further recommended that Scaled Conjugate Gradient (*trainscg*) should be recommended for larger problems because it uses gradient calculations that are more memory efficient and convenient than the Jacobian.

REFERENCES

[1] Omolaye, P. O., Igwue G. A., and Akpakwu G. A. *Okumura-Hata: A Perfect Model for Driving Route UHF Investigation*. American Journal of Engineering Research (AJER), Volume-4, Issue-9, pp-139-147, 2015, e-ISSN: 2320-0847, p-ISSN: 2320-0936.

[2] Omolaye, P. O., Mom, J. M. and Igwue, G. A (2017). A Holistic Review of Soft Computing Techniques. Applied and Computational Mathematics; 6(2): 93-110. http://www.sciencepublishinggroup.com/j/acm. doi: 10.11648/j.acm.20170602.15. ISSN: 2328-5605 (Print); ISSN: 2328-5613 (Online).

[3] L. Ljung and T. Glad, Modelling of Dynamic Systems, Englewood Cliffs, New Jersey: PTR Prentice Hall, Prentice Hall Information and System Sciences Series, 1994, 361pp.

[4] Doucoure, B.; Agbossou, K.; Cardenas, A. Time series prediction using artificial wavelet neural network and multi-resolution analysis: Application to wind speed data. Renew. Energy 2016, 92, 202–211.

[5] Siti H. A. and Antoni, W. Neural Networks Based Nonlinear Time Series Regression for Water Level Forecasting of Dungun River. Journal of Computer Science 8 (9): 1506-1513, 2012. ISSN: 1549-3636. © 2012 Science Publications.

[6] Joseph M. Caswell. (2014). A Nonlinear Autoregressive Approach to Statistical Prediction of Disturbance Storm Time Geomagnetic Fluctuations Using Solar Data. Journal of Signal and Information Processing, 2014, 5, 42-53. http://www.scirp.org/journal/jsip. http://dx.doi.org/10.4236/jsip.2014.52007

[7] Cao, Q.; Ewing, B.T.; Thompson, M.A. Forecasting wind speed with recurrent neural networks. Eur. J. Oper. Res. 2012, 221, 148–154.

[8] Mohanty, S.; Patra, P.K.; Sahoo, S.S. Prediction of global solar radiation using nonlinear auto regressivenetwork with exogenous inputs (narx). In Proceedings of 2015 39th National Systems Conference (NSC), Odisha, India, 14–16 December 2015.

[9] Levesque, L (2014). Nyquist sampling theorem: understanding the illusion of a spinning wheel captured with video camera. IOP publishing, Physics Education, vol. 49, No. 6

[10] Parsapoor, M., Bilstrup, U. (2013). Chaotic Time Series Prediction Using Brain Emotional Learning Based Recurrent Fuzzy System (BELRFS). Intl Journal of Reasoning-based Intelligent Systems, 5(2): 113-126. http://dx.doi.org/10.1504/IJRIS.2013.057273

[11] Miranian, A. and Abdollahzade, M. (2013), 'Developing a Local Least-Squares Support Vector Machines-Based Neuro-Fuzzy Model for Nonlinear and Chaotic Time Series Prediction', Neural Networks and Learning Systems, IEEE Transactions on, Vol.24, no.2, pp. 207-218

[12] Manish, P. G, and Salankar, S. S.(2015), "Time Series Forecasting of Packet Loss Rate Using Artificial Neural Network Based on Particle Swarm Optimization", International Journal of Engineering Research and General Science. Vol 3, Issue 2, pp. 466-472

[13] Parveen, H and Srivastava, R. (2016). Traffic Congestion Prediction using Soft computing Technique. Intl Journal of Research and Development in Applied Science and Engineering (IJRDASE) ISSN: 2454-6844. Vol 10, Issue 1.

[14] Lahmiri, S. (2011). A comparative study of Backpropagation Algorithms in Financial Prediction. International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.1, No.4, August 2011. DOI: 10.5121/ijcsea.2011.1402 15

[15] Vadiraj, D. and Narahari, N.S. (2014). Statistical Modelling Approach to Estimation of Average Revenue per User in Telecom Service – A Case Study. International Journal of Scientific Engineering and Technology (ISSN: 2277-1581). Vol. No.3 Issue No.5, pp: 557-560.

AUTHORS

Engr. Philip Omohimire Omolaye graduated from Electrical and Electronics Engineering Department at Federal University of Technology Akure, Ondo State of Nigeria and a Masters degree in Information Technology Systems and Industrial Electronics from Federal University of Agriculture Makurdi, Benue State of Nigeria. Presently, the author is pursing PhD in Telecommunication System Engineering. He is a registered member of the Council for the Regulation of Engineering in Nigeria, Nigeria Society of Engineers (NSE), Nigeria Institute of Electrical and Electronic Engineers (NIEEE). The author has written several books, technical manuals and many journal articles.

Tajudeen Adeleke Badmos is a graduate of Mathematics/Physics from Federal College of Education Kontagora, Nigeria in 1992. He also obtained his first degree in Mathematics/computer science from Federal University of Technology Minna, same in Niger state in 1997. The author had a Masters Degree in Information Technology Systems and Industrial Electronics from Federal University of Agriculture Makurdi, Benue State, Nigeria in 2013. Presently, He is undertaking his PhD in Robotics (Path Retraction) at Sheffield Hallam University, United Kingdom. The author is a lecturer in the Department of Computer science, Federal Polytechnic Nasarawa, Nigeria and also a former director of ICT in the same school. The author has authored several books and articles. He is happily married with children.

Omolaye O. Philip "Predictive and Comparative Analysis of NARX and NIO Time Series Prediction" American Journal of Engineering Research (AJER), vol. 6, no. 9, 2017, pp. 155-165