SSRG International Journal of Computer Science and Engineering
ISSN: 2348 – 8387 /doi:10.14445/23488387/IJCSE-V8I1P106

Volume 8 Issue 1, 30-38, January 2021

Rainfall Prediction Using Time Series Nonlinear Autoregressive Neural Network

Urooj Kaimkhani[#1], Bushra Naz[*2], Sanam Narejo[#3]

[#]Student (Engineer), Institute of Information and Communication Technologies, Mehran University of Engineering and Technologies, Jamshoro, Pakistan.
*[*2]Assistant Professor, Institute of Information and Communication Technologies, Mehran University of Engineering and Technologies, Jamshoro, Pakistan.*
[#3]Assistant Professor, Institute of Information and Communication Technologies, Mehran University of Engineering and Technologies, Jamshoro, Pakistan.

Abstract *- Weather forecasting is important for the daily life plan of the person, but the agriculture sector is also dependent on the weather condition and several industries. This research work reflects the prediction of the metrological parameter in the time series, i.e., prediction of Rainfall by using ANN (Artificial Neural Network) based model NARX (Nonlinear Autoregressive with exogenous input). In this research paper, more than a few ANN models that rely on real-time sequence recurring NARX-based ANN techniques are initiated, trained, and tested with different parameter settings to find the network model's best possible output to its most-wanted prediction function. The model network's performance is evaluated on the basis of the Mean Squared Error (MSE) performance of the model when the data sets are trained, validated, and tested. Although one step ahead of prediction, multi-phase ahead prediction is more complicated and difficult to carry out because of its underlying added complication. Therefore, the findings found in this work provide useful and helpful suggestions for the NARX-ANN model parameters, particularly the choice of hidden layer size and self-regressive leg terms for an efficient predictor of multi-step time series. This study aims to build and train a network that can predict and predict the weather portion of precipitation by optimizing the parameters of the neural network.*

Keywords - *Artificial Neural network (ANN), NARX model, Outlier recognition, Time Series forecasting, Rainfall prediction.*

I. INTRODUCTION

Due to its changing, continuous, data-intensive, unpredictable, and erratic output patterns, forecasting the weather is a difficult time series prediction problem. Extensive time series forecasting algorithms and techniques currently exist and are widely adopted. However, innovative work continues to advance methods and techniques for reliable and precise forecasting results. Over the last decades, more than a few weather forecasting studies have been tracked to include various promising forecasting models.

Weather Forecasting predicts atmospheric conditions such as snowfall, storm tides, ambient temperature (minimum and maximum), air pressure, Humidity, Rainfall, and floods. Rainfall is one of the most critical parameters of the season. It plays a crucial role in forestry, the food production program, and water resources management. In the chaotic time series results, the unwanted noise signal may be included due to which the basic characteristics of the signal kill and decrease the accuracy of the prediction [4].

Weather forecasting is usually a neutral time series forecasting challenge in scientific meteorology science. The ordered sequence of data samples corrected over time intervals is the time series, as shown in Figure 1. Time series data is used in a variety of applications. For example, few data sequences have seasonality, and few show patterns, i.e., exponential or linear, and few are trendless, unpredictable in a few steps. In response to this scenario, some pre-processing needs to be done to monitor such volatile or fluctuating features. To derive any useful data from the data collection, additional uniqueness of real-time series data and prediction of future data are based on previously defined data, time series testing, forecasting, and predicting models are carried out and deployed.

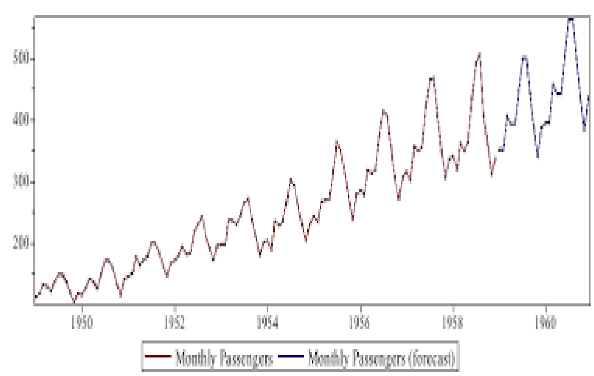

Figure 1 Time series data for monthly Air Passengers

The NARX network is known to delay the neural network (TDNN) [2]. The time series NARX models are important for time-dependent forecasting data and the dynamic system [3].

Rainfall weather forecasting parameter is one of the world's trending problems because it has multi-dimensional and non-linear data in various fields, such as seasonal change and agricultural production. Recently, climate change has created issues in forecasting. Here in our study, we have implemented a rainfall forecasting model using advanced neural network algorithms. Artificial Neural Network involves many inter-connected processors called neurons. Growing neuron generates a series of true valued results. The neurons are triggered through the input sensors, and other neurons are triggered through the previously activated neurons. Some neurons are affected by the triggering actions. And the input in our work is NARX (Nonlinear Auto Regressive Exogenous Data). The model is trained using different parameter settings such as hidden layer size, input delays, feedback delays.

Traditional linear statistical models, such as the ARIMA model or the Box-Jenkins [1-5], have been used in the forecasting field for a long time. Nevertheless, if the basic method is non-linear, these methods do not have the right results. The real-life approach is almost non-linear, taking into account the inadequacy of the conventional mathematical model. Subsequently, in the 1970s and early 1980s, it was found that non-linear models were ideal for real-life methods as opposed to linear models[6][7].

A. NARX-ANN:

The NARX neural network has a robust nonlinear mapping capability relative to others. Unlike other algorithms, NARX-ANN modeling with real-time and continuous data can give better predictive analysis so that NARX can be used extensively in multivariate irregular and continuous time series predictive models [8].

Nowadays, the ANN (Artificial Neural Network) model is used by researchers to predict erratic, chaotic, and continuous-time series results. ANN is divided into two categories, static ANN and dynamic ANN. Static neural networks are widely distributed in the BP and RBF neural networks [8], static neural networks do not have feedback as well as memory feature, for the network to simulate or learn that the current output depends only on the current input, there is no need for past input or past output. Static ANN does not have a memory and feedback function that is why the network's generalization capacity will be small, which is not suitable for the real-time data prediction model. On the other hand, the functional neural network is more fitting for the continuation of the prediction model of the time series. Throughout the chaotic real-time series, past input and past output are closely related to current output since the dynamic neural network has a feedback mechanism and memory mechanism such that we can use the dynamic neural network for chaotic time-series data to achieve reliable and accurate information. NARX regression is more suited to dynamic nonlinear-NN, offering better results than others.

Multi-step input and output delay in the NARX neural network model reflects the prior status of the data. The NARX network has a strong non-linear mapping capability to perform with memory function, adaptability, and accurate performance. It's easier and better for a chaotic time series system.

Previously rainfall prediction was made by the atmosphere's old physical and statistical chart, inaccurate and inaccurate for long periods. Nowadays, neural networks and machine learning techniques are more reliable, secure, stable, and robust. In our research, rainfall prediction technologies were explored to deliver more reliable rainfall results over long periods.

In recent years, several researchers have worked on various models and methods for predicting chaotic and dynamic time series structure, and some are linear predictive models, including AR [9], ARMA [10], ARIMA [11], etc. NN model including RBF neural network [12], echo state [13], extreme learning machines [14], Elman neural network [15], support vector machine [16], least-square support vector machine [17], adaptive filter [18], etc. Several other researchers used chaotic time-series data [19-21] to work on the prediction models.

II. LITERATURE REVIEW

They focused on linear regressive algorithms in their research and a variant on a practical regressive algorithm capable of capturing current weather or atmospheric patterns. Both of their algorithms were performed by specialized weather forecasting systems, so the gap between their algorithms and the specialized one time was quickly reduced for weather forecasting, and their model was able to perform specialized ones for longer time scales. Linear regression proved to be a model of low bias, moderate variance, while functional regression proved to be a model of moderate bias and low variance [9].

The Bayesian classification algorithm was used to recognize weather forecasting and prediction. They compared their algorithm with a variety of algorithms based on a genetic algorithm model, and, as a result, they achieved accurate and précised information [10].

They worked with recurrent neural networks on forecasting the Wind Speed. They have carried out a comparative analysis of wind speed forecasting accuracy between univariate and multivariate ARIMA, and the data set collection is from the Wind Engineering Research Field Laboratory. They compared five different heights for mean, median, temperature (maximum and minimum), and standard deviation for wind forecasting. Consequently, the non-linear

ARIMA models produce more efficient wind speed forecasting results than the linear ARIMA algorithm [11].

They used the non-linear ANN statistic technique to forecast weather parameters and returned the result that the NARMA algorithmic system and the ANN model area unit could not capture the intrinsic complexity of the non-linear time series [12].

This paper collects data for five years from January 2010 to January 2014 from the weather department at the Nagpur station. Temperature, pressure, and Humidity are typically the responsible parameters for the precipitation area unit. The usable datasets analyzed and calculated the Mean Absolute Error (MAE), Mean Square Error (MSE), and Standard Deviation (SD) variables in the Normal Trend Growth Algorithm [13].

Their proposed study is tested by comparing the models, the classic Box-Jenkins (AR) and other traditional neural networks (FTDNN and Elman). From these models, better performance is given by the NARX network [14].

III. DATA SET RESOURCE

Meteorological data in real-time were downloaded from the Mateo Weather Station mounted, as seen in Fig.2, at the Neuronica Laboratory, Politecnico Di Torino. Further research is carried out by adding certain pre-processing measures to data extraction, noise reduction, and outlier identification.

Figure 2 : Mateo Weather Station [14]

IV. PROPOSED METHODOLOGY

We used advanced neural network algorithms to forecast and model prediction. The use of neural networks is entirely focused on neural networks' potential to estimate nonlinear functions in time series forecasting. Consecutively predicting and forecasting rainfall values, the selection of ANN that we considered being NARX. NARX stands for a Nonlinear Auto-Regressive neural network model with exogenous inputs. The NARX algorithm is derived from self-regressive algorithms. The term self-regression means that the object regresses toward itself. In this respect, along with the auto-regressive inputs, the NARX algorithm uses the previous target values as outputs. ANN-NARX is the nonlinear Auto-Regressive exogenous-input network algorithm [14]. NARX neural regression network has features such as input layer, hidden layer neurons, an output layer, input delay, and output delay, estimating input delay and output delay order number, and pre-training the hidden layer size of neurons. Fig.3 displays the schematic diagram for the neural network.

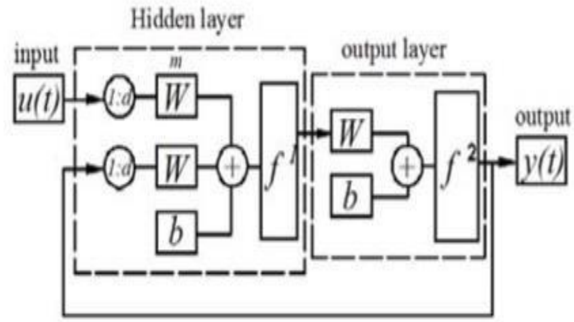

Figure 3 NARX Neural Network System

In above Fig.3, the neural network parameters consist of some features such as d is order delay, m is the hidden layer neurons, W is the weight matrix of the input vector and the feedback vector of the network, b is biased, f1 is the activation function of the hidden layer of the neural network with 'tansig' function, f2 is the activation function of the hidden layer of the neural network with 'purelin' transfer function. The NARX-ANN model is based on the time delay system, and the dynamic recurrent network is based on performance feedback. The neural network model includes both original input data and output data after training during the three steps of training, validation, and testing, so that the generalization capability of the neural network is better than before. Here's the mathematical representation of the NARX-ANN model:

$$y(t)=f[y(t-1),y(t-2),\ldots,y(t-ny),u(t-1),u(t-2),\ldots,u(t-nu)] \quad (1)$$

The function f(.) shows the ANN non-linear function. We observed that the input uses the previous one's output for the feedback function and enhances the network's efficiency and accuracy through the open-loop and closed-loop training. If we want to fulfill our requirements for better performance, it is important to set the appropriate network model parameters. The function f(.) of the above equation (1) depends on the number of hidden layer neurons. If we use more neurons, the output result will be closed to the expected target value and the prediction error, Mean Squared Error (MSE), autocorrelation, and input errors will be low in the error map NARX network training, making the NARX model better. If any of these errors are high, The parameter settings, such as the number of neurons in the hidden layer, the order of the input delay, and the delay of the feedback, should be changed until the output value and the target value are not in the same processor stage, and the error between the output value and the target value is in the correct range. To get better output analysis, the neural network parameters, just like a number of hidden layer neurons and input delays, need to be checked repeatedly.

A. MODEL ESTIMATION CRITERIA

In this research, I have used Mean Squared Error (MSE) to determine the prediction and forecasting errors. Mean Squared Error is the way to obtain the system errors, which estimate the deviation between the predicted and targeted value. The mathematical representation of the Mean Squared Error is given below:

$$MSE = \sqrt{\frac{\sum_{i-1}^{n}(Ytrue,i - Ypredict,j)^2}{n}} \quad \ldots\ldots\ldots\ldots (2)$$

In the above formula, the Ytrue, i is the targeted value and, the predicted value is Ypredict,j, and n is the number of test sets attempted.

A flowchart of the research is given below:

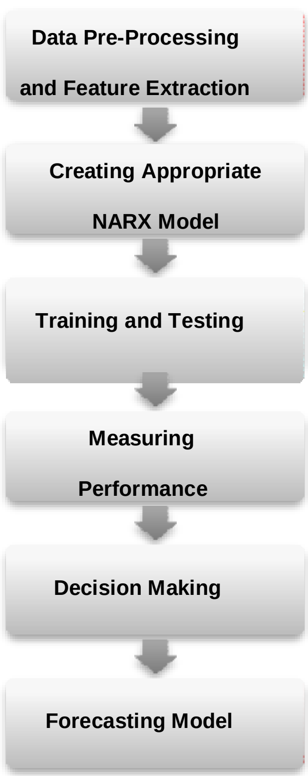

Figure 4 : Methodology

Prediction and estimation of precipitation involves a dynamic process that has not yet been completely understood. To order to overcome this, we have tried to put even more work into making rainfall forecasts.

a) DATA PRE-PROCESSING AND FEATURE EXTRACTION:

For getting accuracy in the rainfall predictions, we need to have meaningful and realistic datasets that include content participation and likely compact errors. Internal and exterior properties of the precipitation rainfall prediction sector depend on various factors, including wind, temperature, Humidity, previously obtained rainfall data and meteorological characteristics of the catchment area, and so on.

A valid dataset is filtered using a low-pass butter-value filter from a fluctuating, noisy and bulky dataset. Here in Figure 5, we can see the filtered dataset after pre-processing.

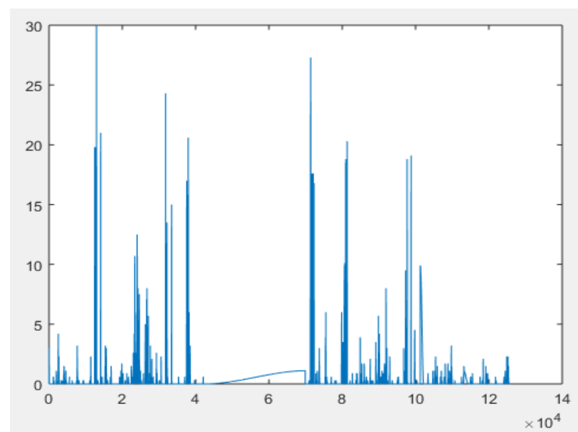

Figure 5 : Rainfall Filtered Data

b) CREATING NARX MODEL:

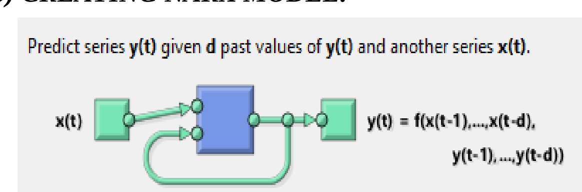

Figure 6 : NARX Model

The time series's potential forecast values from the past values of that time series and past values of the second time series x(t) are seen in the above figure y(t). This kind of model of prediction is called NARX and can be written mathematically as follows:

$$y(t)= f(y(t-1),., y(t-d), x(t-1), ..., (t-d))\text{--- } (3)$$

The typical NARX network is a two-layer feed-forward network, along with the hidden layer's sigmoid transfer function and the output layer's linear transfer function. NARX uses tapped delay lines to store previously obtained x(t) and y(t) series values. Notice that the NARX network output, y(t), is returned to the NARX network input using delays, as y(t) is a function of y(t – 1), y(t–2),...,y(t–d).

However, it is possible to open this feedback loop for competent training. Since the true output is usable when the network training occurs, in which the actual output is used despite feeding back the simulated output, it has two benefits. The first is that the feedback is more reliable and concise for the feed-forward network. The second is that there is a pure feed-forward architecture in the resulting network, and a more robust training algorithm can then be used. The default number of hidden layer neurons is 10, and the delay order is 2. For improved performance measurement, we should adjust these settings.

c) TRAINING AND TESTING:

Our research has used Matlab Neural Network nts toolbox to build up and test the NARX model. Input dataset for the Rainfall is the chaotic time-series data, which collects different weather parameters for better rainfall prediction performance analysis, such as Humidity, temperature, and previously obtained rainfall outputs. We determined the future values of the Rainfall; if the input is x(t), we determined x(t+1). This parameter is obtained by using the NARX NN model's proper parameter setting, such as the number of hidden neurons on the layer and the order of delay. We tested a lot of various function sets, first, by sequencing the meteorological parameters by adding and deleting these. Subsequently, find the necessary lagged terms of the chosen parameters to be used as characteristics to be included. This was further split into the training range (70%), validation (15%), and checking with a 15% ratio, respectively.

Table 1 Parameters Setting

Parameter	Setting
Maximum number of epochs for training	1000
MSE	0.00
Hidden Layers	Multiples
Input Delay	Multiples
Feedback Delay	Multiples

d) TRAINING ALGORITHM:

Levenberg Marquardt (trainlm) is preferred for most issues, except for certain loud and minor issues. It can take longer for Bayesian Regularization (trainbr).
The training continued until, for six iterations (validation stop), until the validation error failed to decrease.

V. RESULTS

After training the network 200 times with different parameter settings by changing input delays, feedback delays, and the number of hidden layer neurons. Then we get to know that by using delay order is 2 to 4, and hidden layer neurons are set to 35 to 65 then the training get better performance effect, and there is a small deviation between output and target values and MSE gets closer to 0 with 0.00483 value at input delays 2 and feedback delays 3, and the number of hidden layer neuron size is 55. The network model we obtained is shown in the figure.

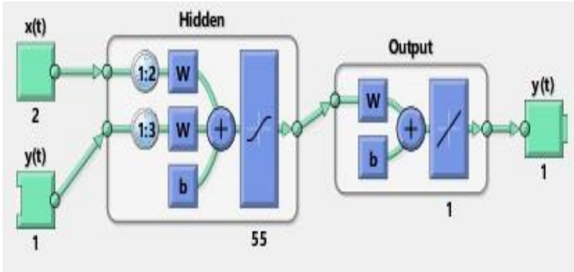

Figure 7 : Training Model

The following figure shows the relationship between output and real or target values, and R is the correlation factor between output and target values. It gave 0.9947 near the 1, which is the closest approximation of the results we have achieved. The figure is given below:

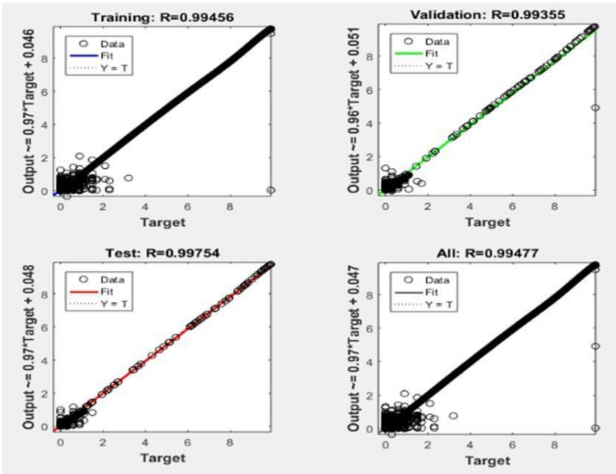

Figure 8 : Correlation between Target and Output Values

The figure below shows that errors in training, validation, and testing decreased until iteration 23. There does not seem to have been any overfitting, as neither testing nor validity error improved before iteration 23.

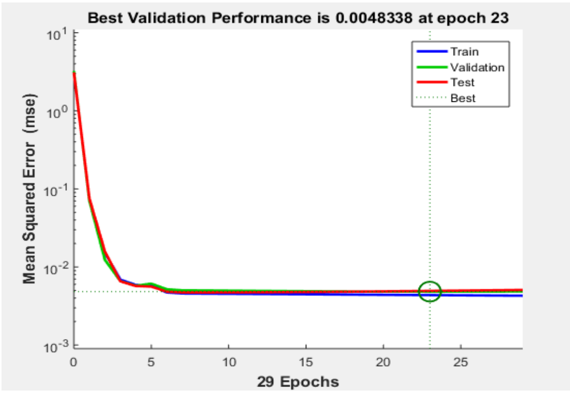

Figure 9 : MSE Performance

All preparation, including the validation and testing phases, is done in an open-loop (also called a series-parallel architecture). The typical workflow is to create a network in an open-loop entirely, and it can be transformed into a closed-loop for multi-step prediction only after it has been educated (including validation and checking steps). Similarly, depending on the open-loop training data, the R values in the Interface are calculated.

Close the NARX network loop, so a one-step prediction is made when the feedback loop is opened on the NARX network. Predict the next y(t) value from the previous y(t) and x(t) values. With the loop closed, multi-step forward predictions can be used. This is because the y(t) predictions are used instead of the true y(t) values of the future.

More calculation is needed to maximize the number of neurons and the number of delays, and this appears to over-fit the data when the numbers are set too high, but it helps the network solve more complex problems. More layers require more computation, but their use can result in more effective network resolution of complex issues. To use more than one hidden layer, insert hidden layer sizes in the fitness command as array elements.

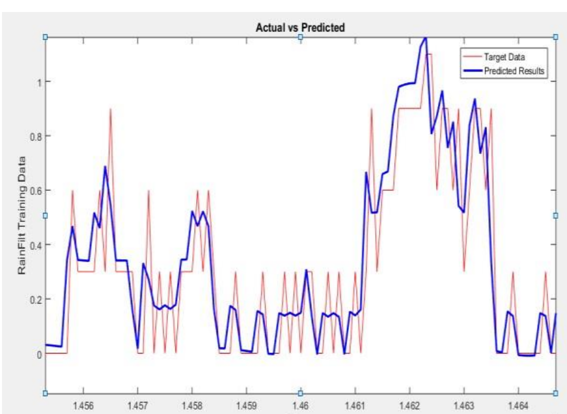

Figure 10 : Rainfall Prediction

Figures 10 and 11 show the deviation between the actual and targeted (predicted) value of the Rainfall, which is in the minor phase.

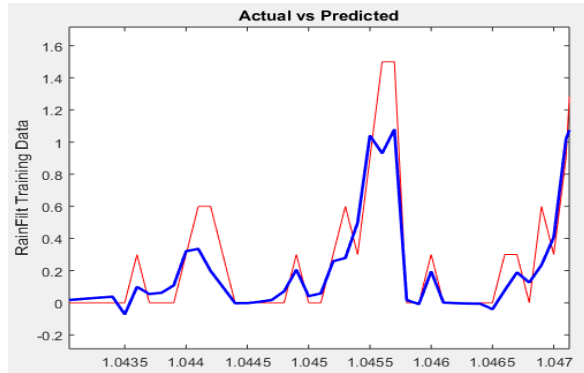

Figure 11: Rainfall Prediction

Figure 12 below the forecasting performance is measured for 4-steps ahead, with test planned for the next 1 hour onwards. This plot displays the 1-hour prediction errors for 4 separate test sets that were later simulated.

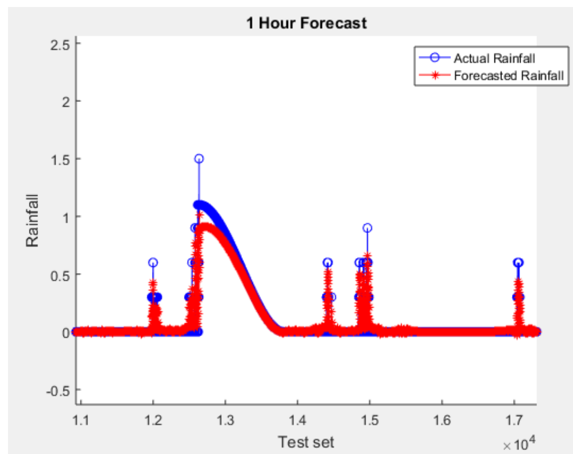

Figure 12 4 :Steps ahead forecasting

Figure 13 below the forecasting performance is measured for 8-steps ahead, 8-steps ahead, with test planned for the next 2 hours onwards. This plot displays the 2-hour prediction errors for 8 separate test sets that were later simulated.

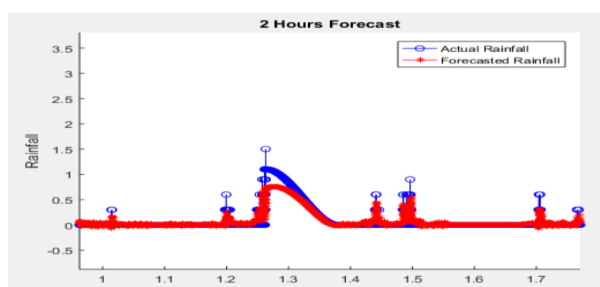

Figure 13 8 steps ahead forecasting

Quality forecasting review for 12-steps forward forecasting for each test range for the next 3 hours onwards shows that forecasting models can present higher MSE or lower precision instead of shorter forecasts for long-term future values. Figure 14 below is as follows:

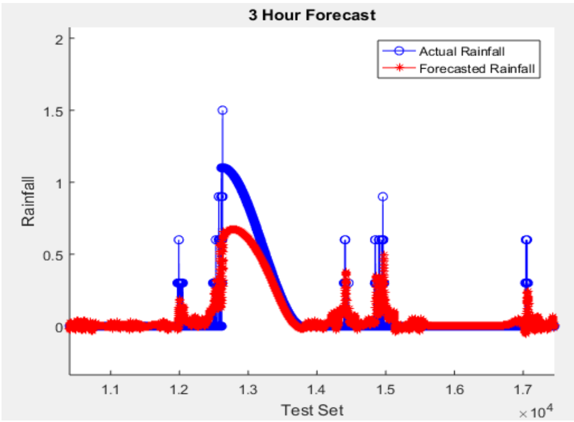

Figure 14 12 : steps ahead forecasting

Table 2 : Forecasting Analysis

Step Ahead Prediction	Hourly Prediction	MSE
4 Steps	1 Hour	0.0134
8 Steps	2 Hours	0.0341
12 Steps	3 Hours	0.0524

B. CONCLUSION:

In this research, prediction and forecasting of Rainfall were carried out using the algorithm to find the optimal network model according to their desired forecasting function. NARX models based on much different time series have been developed and trained with different parameter settings.

This work's primary goal was to contribute to the study and development of multi-step forward forecasting with NARX modeling. Because of the increasing phases 4, 8, and 12 for the next 1, 2, and 3 hours, forecasting was computed.

On short-term multi-step forecasting, the forecasting model can be better than longer models. For short-term future values, forecast models offer much higher accuracy than longer forecasts because the real-time series is essentially chronological data with temporal modifications.

This means that the correct selection of lagged terms increases the stability of the model for reliable predictions. This means that the right choice of lagging conditions increases the credibility of the model for detailed forecasts. The delay terms must, based on this, be chosen and taken into consideration.

C. FUTURE WORK:

From this work, the findings concluded that the proposed model was capable and could be further applied to the multi-step forward forecast of a complex range of weather parameters, such as Humidity, pressure, and temperature.

Table 3 : Different Parameter Settings To achieve desired Forecasting Analysis

S.No	Parameter Settings	Open loop training Results (MSE)		Closed loop training Results (MSE)	
1.	Hidden Layer Size = 3 Input Delay = 1:2 Feedback Delay = 1:2	Training	0.127	Training	2.118
		Testing	0.0035	Testing	0.0021
2.	Hidden Layer Size = 5 Input Delay = 1:2 Feedback Delay = 1:4	Training	0.241	Training	4.219
		Testing	0.0082	Testing	0.003
3.	Hidden Layer Size = 10 Input Delay = 1:2 Feedback Delay = 1:2	Training	0.192	Training	3.191
		Testing	0.0068	Testing	0.0081
4.	Hidden Layer Size = 15 Input Delay = 1:2 Feedback Delay = 1:2	Training	1.068	Training	3.192
		Testing	0.007	Testing	0.0052
5.	Hidden Layer Size = 25 Input Delay = 1:2 Feedback Delay = 1:4	Training	1.071	Training	5.052
		Testing	0.0071	Testing	0.0053
6.	Hidden Layer Size = 35 Input Delay = 1:2 Feedback Delay = 1:3	Training	0.1364	Training	2.037
		Testing	0.0094	Testing	0.003
7.	Hidden Layer Size = 48 Input Delay = 1:2 Feedback Delay = 1:3	Training	1.0622	Training	6.243
		Testing	0.0069	Testing	0.0051
8.	Hidden Layer Size = 55 Input Delay = 1:2 Feedback Delay = 1:3	Training	0.00562	Training	1.2627
		Testing	0.0012	Testing	0.0011
9.	Hidden Layer Size = 60 Input Delay = 1:2 Feedback Delay = 1:4	Training	0.1209	Training	1.2338
		Testing	0.0092	Testing	0.0086
10.	Hidden Layer Size = 65 Input Delay = 1:2 Feedback Delay = 1:3	Training	0.1231	Training	1.1281
		Testing	0.0057	Testing	0.0049

REFERENCES

[1] G.E. Box, G.M. Jenkins, G.C. Reinsel, and G.M. Ljung, " Time series analysis: forecasting and control. "Holden-Day, 1976.

[2] G.P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model.", Neuro-computing, vol. 50, pp. 159175, 2003.

[3] C. Brooks, "Univariatetime series modelling and forecasting.", Introductory Econometrics for Finance. 2nd Ed. Cambridge University Press. Cambridge, Massachusetts, 2008.

[4] J.H. Cochrane, Time series for macroeconomics and finance. Manuscript, University of Chicago, 2005.

[5] K.W. Hipel, and A.I. McLeod, Time series modelling of water resources and environmental systems, Elsevier. vol. 45, 1994.

[6] Taieb, Souhaib Ben, Gianluca Bontempi, Amir F. Atiya, and Antti Sorjamaa. "A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition." Expert systems with applications 39, no. 8 (2012): 7067-7083.

[7] De Gooijer, Jan G., and Rob J. Hyndman. "25 years of time series forecasting." International journal of forecasting 22, no. 3 (2006): 443-473.

[8] Xiu, Yan, and Wei Zhang. "Multivariate Chaotic Time Series Prediction Based on NARX Neural Networks." In 2017 2nd International Conference on Electrical, Automation and Mechanical Engineering (EAME 2017). Atlantis Press, 2017.

[9] Holmstrom, Mark, Dylan Liu, and Christopher Vo. "Machine learning applied to weather forecasting." (2016).

[10] Liu, James NK, Bavy NL Li, and Tharam S. Dillon. "An improved naive Bayesian classifier technique coupled with a novel input solution method [rainfall prediction]." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 31, no. 2 (2001): 249-256.

[11] Cao, Qing, Bradley T. Ewing, and Mark A. Thompson. "Forecasting wind speed with recurrent neural networks." European Journal of Operational Research 221, no. 1 (2012): 148-154.

[12] Ahmed Tealab a,*, Hesham Hefny a, Amr Badr b a Computer Science Department, Institute of Statistical Studies and Research, Cairo University, Giza, Egypt b Computer Science Department, Faculty of Computers and Information, Cairo University, Giza, Egypt Received 19 March 2017; accepted 14 May 2017 Available online 3 July 2017

[13] Taksande, Amruta A., and P. S. Mohod. "Applications of data mining in weather forecasting using frequent pattern growth algorithm." International Journal of Science and Research (2015).

[14] Narejo, Sanam, and Eros Pasero. "Meteonowcasting using deep learning architecture." (IJACSA) International Journal of Advanced Computer Science and Applications 8, no. 8 (2017).

[15] Math Works. Matlab R2016a Neural Network Toolbox user Guide. 2016

[16] Narasayya.Kamuju,Rainfall-Runoff Estimation and Comparative Analysis Using Advanced Geospatial Digital Hydrological Modelling Tools, ArcCN-Runoff & ArcSWAT., SSRG International Journal of Geoinformatics and Geological Science 2(3)(2015) 5-10.

[17] Bishwajit Roy, Maheshwari Prasad Singh. An Empirical-based rainfall-runoff modeling using optimization Technique. International Journal of River Basin Management 0:0, pages 1-19, 2019.

[18] Yaseen, Z.M.; Ebtehaj, I.; Kim, S.; Sanikhani, H.; Asadi, H.; Ghareb, M.I.; Bonakdari, H.; Wan Mohtar, W.H.M.; AlAnsari, N.; Shahid, S. Novel Hybrid Data Intelligence Model for Forecasting Monthly Rainfall with Uncertainty Analysis. Water 2019, 11, 502.

[19] Le TT., Pham B.T., Ly HB., Shirzadi A., Le L.M. (2020) Development of 48-hour Precipitation Forecasting Model using Nonlinear Autoregressive Neural Network. In: Ha-Minh C., Dao D., Benboudjema F., Derrible S., Huynh D., Tang A. (eds) CIGOS 2019, Innovation for Sustainable Infrastructure. Lecture Notes in Civil Engineering, vol 54. Springer, Singapore

[20] Esteves, João Trevizoli, Glauco de Souza Rolim, and Antonio Sergio Ferraudo. "Rainfall prediction methodology with binary multilayer perceptron neural networks." Climate Dynamics 52.3-4 (2019): 2319-2331.

[21] Arabeyyat, Omar, Nawras Shatnawi, and Mohammed Matouq. "Nonlinear Multivariate Rainfall Prediction in Jordan Using NARX-ANN Model with GIS Techniques." Jordan Journal of Civil Engineering 12.3 (2018).

[22] Geetali Saha Narendra Chauhan , C. "Analysis methods for numerical weather prediction." Quarterly Journal of the Royal Meteorological Society 112.474 (1986): " (2017)

[23] Cao, Qing, Bradley T. Ewing, and Mark A. Thompson. "Forecasting wind speed with recurrent neural networks." European Journal of Operational Research 221.1 (2012): 148-154."Forecasting wind speed with recurrent neural networks", Qing Cao a , Bradley T. Ewing a,b , Mark A. Thompson a,b, 2012.

[24] Mark Holmstrom, Dylan Liu, Christopher Vo, internal report from Stanford University, "Weather forecasting", (December 15, 2016),

[25] James N.K.Liu, Bavy N.L.Li, and Tharam S.Dillon, "Weather forecasting and prediction", (May 2001)

[26] Amruta A. Taksande, P. S. Mohod. An internal Report, "Rainfall prediction", 2003.

[27] Darji, M. P., V. K. Dabhi, and H. B. Prajapati. &quot, "Rainfall forecasting using neural network" A survey." In Computer Engineering and Applications (ICACEA), , pp. 706-713, 2015.

[28] Hossain, Moinul, et al. "Forecasting the weather of Nevada: A deep learning approach." 2015 international joint conference on neural networks (IJCNN). IEEE, 2015

[29] Hernández, Emilcy, et al. "Rainfall prediction: A deep learning approach, "International Conference on Hybrid Artificial Intelligence Systems. Springer, Cham, 2016.