Rainfall Forecasting Models Using Focused Time-Delay Neural Networks

Kyaw Kyaw Htike and Othman O. Khalifa
Department of Electrical and Computer Engineering
International Islamic University Malaysia
ali.kyaw@gmail.com, khalifa@iiu.edu.my

Abstract—**Rainfall forecasting is vital for making important decisions and performing strategic planning in agriculture-dependent countries. Despite its importance, statistical rainfall forecasting, especially for long-term, has been proven to be a great challenge due to the dynamic nature of climate phenomena and random fluctuations involved in the process. Artificial Neural Networks (ANNs) have recently become very popular and they are one of the most widely used forecasting models that have enjoyed fruitful applications for forecasting purposes in many domains of engineering and computer science. The main contribution of this research is in the design, implementation and comparison of rainfall forecasting models using Focused Time-Delay Neural Networks (FTDNN). The optimal parameters of the neural network architectures were obtained from experiments while networks were trained to perform one-step-ahead predictions. The daily rainfall dataset, obtained from Malaysia Meteorological Department (MMD), was converted to monthly, biannually, quarterly and monthly datasets. Training and testing were performed on each of the datasets and corresponding accuracies of the forecasts were measured using Mean Absolute Percent Error. For testing data, results indicate that yearly rainfall dataset gives the most accurate forecasts (94.25%). As future work, more parameters such as temperature, humidity and sunshine data can be incorporated into the neural network for superior forecasting performance.**

Keywords-rainfall; forecasting; neural networks; dynamic systems; focused time delay neural networks; statistical forecasting

I. INTRODUCTION

Rainfall is one of the most important components of water resource management for decision making and planning especially in agricultural sectors. The ability to predict and forecast rainfall quantitatively can help crop planting decisions, reservoir water resource allocation, traffic control, the operation of sewer systems [1] and confronting water-related problems such as flood and draught [2] especially in countries such as Malaysia where agriculture contributes much to the wealth and economy of the country. Therefore, an accurate forecast of rainfall will help in natural disaster mitigation.

However, rainfall is one of the most complex and challenging components of the hydrology cycle to comprehend and to forecast due to the various dynamic environmental factors and random variations both spatially and temporally. There are several reasons why ANNs are valuable and appropriate for use in such forecasting systems. Firstly, they are data-driven methods which have the ability to model both linear and non-linear systems without needing to make priori assumptions which are implicit in most classical statistical approaches such as the Box–Jenkins or ARIMA which assume that the time series under study are generated from linear processes, which is not the case in most real-world situations [2]. Secondly, they are capable of generalization. After learning the data that have been given to them during the training, they can often correctly estimate the unseen part of a population which is not part of the training data. Finally, they have been shown to be universal functional approximators and can approximate any continuous function to any desired accuracy.

II. RELATED WORKS

There have been many attempts to forecast rainfall. Rainfall forecasting can apply to many time horizons such as short term [3], medium term, and long term periods [4] [5]. Some authors design systems which can forecast yearly data, some try to forecast monthly data [5] whereas some try to forecast daily data [6]. Most of them concentrate on one-step-ahead prediction. If multi-step prediction is then required, many iterations of one-step-ahead can be performed. The accuracy of the forecasts would of course decrease with the number of such iterations.

The traditional techniques for statistical weather forecasting include ARMA models, Box-Jenkins Models and Multivariate Adaptive Regression Splines [7]. When the machine learning became popular, there have been many attempts to build rainfall forecasting models using recurrent neural networks [8], feed-forward neural networks with input delays [2] and NARX networks. Many attempts have also been made to incorporate extra weather parameters in the rainfall forecasting model for improved predictions [9].

In [10], ANNs were used to forecast the spatial distribution of rainfall for an urban catchment. The types of ANNs implemented were multi-layer feed-forward neural networks (MLFN), partial recurrent neural networks and time-delay neural networks (TDNN) to perform one-step-ahead predictions. The authors found out that MLFN and TDNN could capture the dynamic structure of the rainfall process. In

978-1-4244-6235-3/10/$26.00 ©2010 IEEE

[5], the authors developed a rainfall prediction model for monthly precipitation mapping of Myanmar. They experimented with 3-layer neural network models using different network architectures to find the optimal parameters. In [11], the authors have validated that neural network models outperformed all comparative methods by comparing the neural network models with linear regression, multiple regression, stepwise polynomial regression, discriminant analysis, logic models, and rule-based systems. In [7], the authors compared the performance of Multivariate Adaptive Regression Splines (MARS) and ANNs for one-month-ahead prediction of rainfall using 84 years of rainfall data in Kerala state in India. Their results reveal that MARS is a better and more robust forecaster than neural networks in terms of performance time and lowest RMSE. They concluded that network performance could be improved by incorporating more training data.

III. Methodology & Implementation

A. Datasets Used

The daily rainfall data, measured in millimeter (*mm*), was obtained from Subang Meteorological Station in Malaysia from January 1980 to May 2009 for a total period of 29 years. In this paper, four different forecasting models have been built, each using a different dataset. Summing up the daily rainfall values in a particular month gives monthly data for that month. The same process was repeated for quarterly (by adding 3-months data at a time), biannually (6 months) and yearly (12 months). For each experiment, 80% of the dataset was used for training and the remaining 20% was used for testing.

B. Pre-processing of Data

In the raw data, the value of "-33.3" was denoted to mean "trace" waterfall levels which are less than 0.1 mm. Therefore the first step is to convert such values to a very small arbitrary number which was taken to be "0.0000025". Before applying the data to the neural network, firstly, all the data are normalized to lie within the range from -1 to 1, which makes it easier for the network to handle. The formula used for normalization is given by:

$$y = \frac{(y_{\max} - y_{\min}) \times (x - x_{\min})}{(x_{\max} - x_{\min})} + y_{\min} \qquad (1)$$

x = Data to be normalized.
x_{max} = The maximum value of all the input data.
x_{min} = The minimum value of the all the input data.
y = Normalized data.
y_{max} = The desired maximum normalized value.
y_{min} = The desired minimum normalized value.

Secondly, all the constant rows which do not give any new information to the system are removed.

C. Neural Network Model

All the four different forecasting models were implemented using Focused Time-Delay Neural Networks (FTDNNs). The basic architecture is the same for all the models. A FTDNN is basically a feed-forward neural network (shown in Fig. 1) with a tapped delay line at the input. It is a member of family of dynamic networks, called focused networks, in which the dynamics appear only at the input layer of a static multilayer feed-forward network, which makes it suitable for time series prediction.

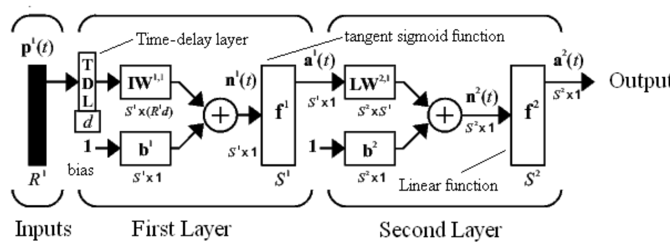

Figure 1. The Feed-Forward Neural Network architecture.

D. Neural Network Architecture

Each of the FTDNNs in this paper has an input layer whose size (decided by the delay) varies for each model, a hidden layer whose size is determined by the number of hidden neurons which again is different from model to model and lastly an output layer which contains a single neuron. Although there have been several attempts to approximate the number of hidden neurons and the delay [12], it has been found and established that it is impossible to accurately determine those parameters by calculation since they depend on many factors. That is why in this paper, the optimal parameters for the neural networks have been obtained from systematic trial and error.

The transfer functions for the hidden layer and the output layer are hyperbolic tangent sigmoid function and pure linear function respectively.

E. Training the Neural Network Models

The network uses the Levenberg-Marquardt back-propagation network training function which updates weight and bias values according to Levenberg-Marquardt optimization. The stopping criteria for the training process are:

1. The maximum number of epochs (repetitions) is reached.
2. The maximum amount of time is exceeded.
3. The performance gradient falls below the minimum gradient.
4. Validation performance has increased more than the maximum amount of times permitted to fail.

IV. RESULTS & ANALYSIS

All implementations and experiments have been carried out in MATLAB R2008a.

A. Evaluation Critierion

In statistical forecasting models, there have been numerous proposed methods to measure the accuracy [13] which judges how 'good' the model is, i.e. the reliability of the model in real life. The most common measure of accuracy is the Mean Absolute Percent Error (MAPE) [14]. Although several authors [14] have asserted that the Symmetrical Mean Absolute Percent Error (SMAPE) gives a better measurement of the accuracy, [15] has found out that if we consider positive and negative errors, SMAPE is far from symmetric, especially where these errors have large absolute values. Therefore, in this paper, MAPE has been adopted and it is given by:

$$MAPE\ (\%) = \frac{1}{n}\sum_{t=1}^{n}\left\{\left|\frac{At-Ft}{At}\right| \times 100\right\} \quad (2)$$

A_t = Observed value or true value or actual value.
F_t = Forecast (predicted) value.

Thus, the accuracy is given by:

$$Accuracy\ (\%) = 100 - MAPE\ (\%) \quad (3)$$

From the above equation, it follows that if the MAPE (%) is greater than 100%, then the accuracy is zero.

B. Results for Yearly Dataset

Table I shows the ANN architecture parameters for the yearly dataset. Both the optimal number of hidden neurons and delay were found to be '5'. From the percentage accuracy of the training data, it can be observed that the model can fit the training data almost perfectly. With the testing data, the accuracy was found to be 94.25%.

TABLE I. ANN ARCHITECTURE FOR YEARLY DATASET

Parameter	Optimal Value
Total number of data	29
Number of training data	23
Number of testing data	6
Number of hidden neurons	5
Delay	5
Accuracy (%) of training data	99.99
Accuracy (%) of testing data	94.25

Fig. 2 and Fig. 3 illustrate the results for the training and testing data respectively. The solid line represents the observed (true) values and the dashed line is for the forecast (predicted) values. As given by the percentage accuracy for the training data, the solid and dashed lines almost completely overlap each other in Fig 2. From Fig. 3 (testing dataset), it can be seen that the observations and the forecasts are very close, in addition to the very high similarity in the trends.

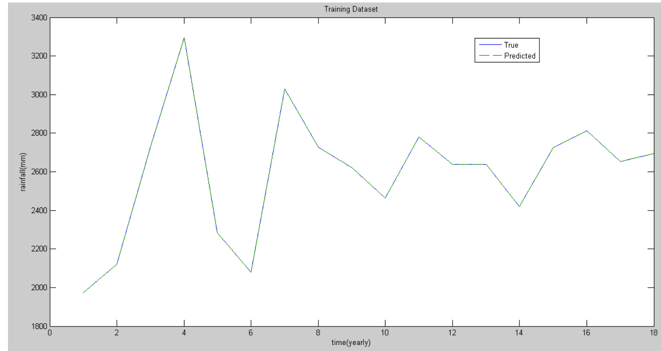

Figure 2. Rainfall forecasts for yearly training dataset.

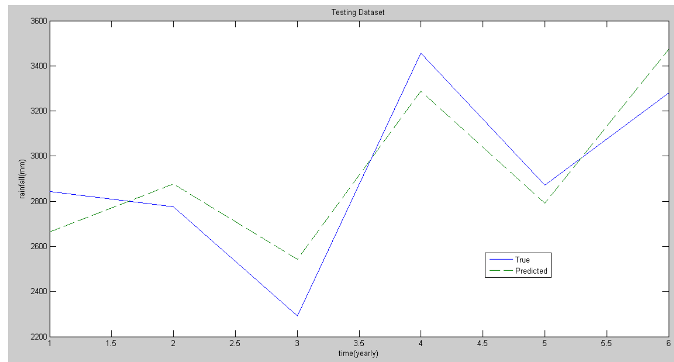

Figure 3. Rainfall forecasts for yearly testing dataset.

In Table II, a comparison of true rainfall values and corresponding forecast values for the yearly dataset is given.

TABLE II. DETAILS OF RESULTS OBTAINED WITH YEARLY TESTING DATASET

True Value (mm)	Forecast Value (mm)
2843.875	2664.368
2776.350	2876.584
2293.150	2543.849
3455.525	3287.140
2872.550	2792.339
3279.450	3475.308

C. Results for Biannual Dataset

For the biannual dataset, as shown in the table below, the optimal parameters for the neural network were experimentally found to be the same as for the yearly dataset. Both the accuracy of the training data and testing data were less than those obtained with the yearly dataset as observed in the table, Fig. 4 and Fig. 5.

TABLE III. ANN ARCHITECTURE FOR BIANNUAL DATASET

Parameter	Optimal Value
Total number of data	58
Number of training data	46
Number of testing data	12
Number of hidden neurons	5
Delay	5
Accuracy (%) of training data	98.13
Accuracy (%) of testing data	81.11

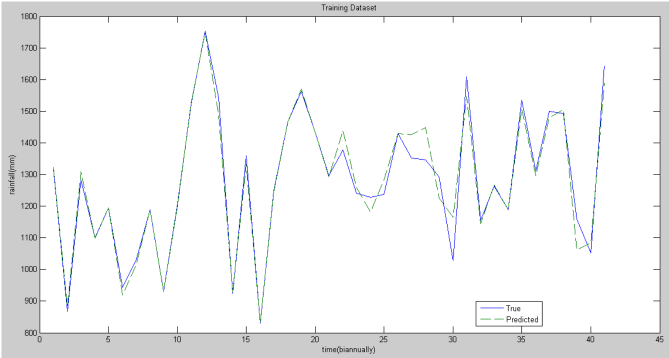

Figure 4. Rainfall forecasts for biannual training dataset.

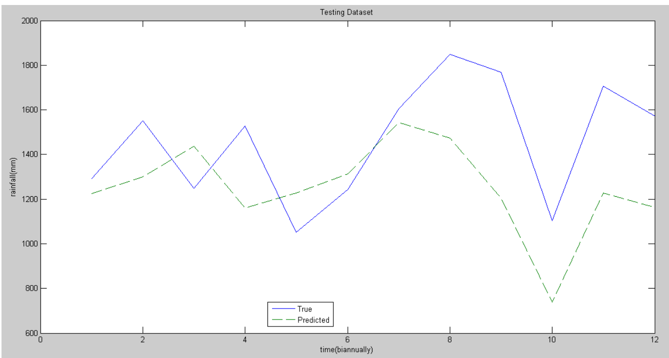

Figure 5. Rainfall forecasts for biannual testing dataset.

True rainfall values and corresponding forecast values for the biannual testing dataset are compared in Table IV.

TABLE IV. DETAILS OF RESULTS OBTAINED WITH BIANNUAL TESTING DATASET

True Value (mm)	Forecast Value (mm)	True Value (mm)	Forecast Value (mm)
1293.225	1225.826	1606.425	1542.473
1550.65	1301.19	1849.1	1472.551
1247.7	1437.023	1769.125	1204.826
1528.65	1159.413	1103.425	738.7929
1051.275	1228.759	1707	1227.731
1241.875	1312.839	1572.45	1162.578

D. *Results for Quarterly Dataset*

The number of optimal hidden neurons was '5' and the optimal delay was found to be '10'. The accuracies are approximately 82% and 76% for training and testing data respectively. Fig. 6 and Fig. 7 show that the overall trend of forecast values is still very similar to the true values.

TABLE V. ANN ARCHITECTURE FOR QUARTERLY DATASET

Parameter	Optimal Value
Total number of data	117
Number of training data	93
Number of testing data	24
Number of hidden neurons	5
Delay	10
Accuracy (%) of training data	81.96
Accuracy (%) of testing data	76.03

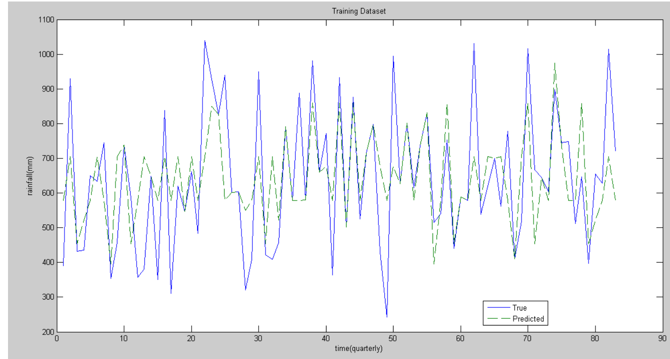

Figure 6. Rainfall forecasts for quarterly training dataset.

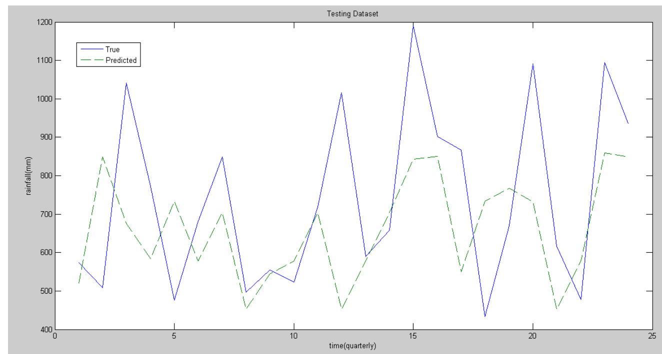

Figure 7. Rainfall forecasts for quarterly testing dataset.

True rainfall values and corresponding forecast values for the quarterly testing dataset are given in Table VI.

TABLE VI. DETAILS OF RESULTS OBTAINED WITH QUATERLY TESTING DATASET

True Value (mm)	Forecast Value (mm)	True Value (mm)	Forecast Value (mm)	True Value (mm)	Forecast Value (mm)
573.975	521.1305	554.25	544.4624	866.475	550.6822
509.1	849.4989	523.5	578.7178	432.975	733.6639
1041.55	676.0765	718.375	704.1122	670.45	766.7566
771.3	584.2418	1015.875	452.9292	1090.75	732.2491
476.4	733.6639	590.55	578.7177	616.25	452.9292
680.125	578.7178	658.675	704.1122	477.675	578.7178
848.525	704.1664	1190.425	843.0638	1094.775	859.0583
497.025	452.9297	902.65	849.5779	934.3	849.1824

E. *Results for Monthly Dataset*

Using the monthly dataset, the optimal delay for the network was established to be '55' which is an enormous increase compared to previous datasets. The accuracy of the model also plunges. However, as illustrated in Fig. 8 and Fig. 9, there is still an acceptable level of correlation in the trends between the true values and forecast values.

TABLE VII. ANN ARCHITECTURE FOR MONTHLY DATASET

Parameter	Optimal Value
Total number of data	353
Number of training data	282
Number of testing data	71
Number of hidden neurons	5
Delay	55
Accuracy (%) of training data	55.21
Accuracy (%) of testing data	56.02

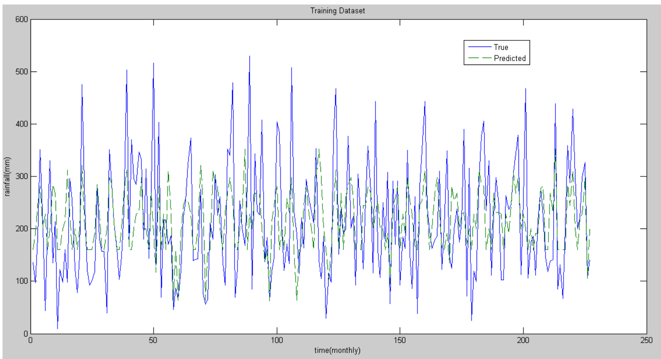

Figure 8. Rainfall forecast for monthly training dataset.

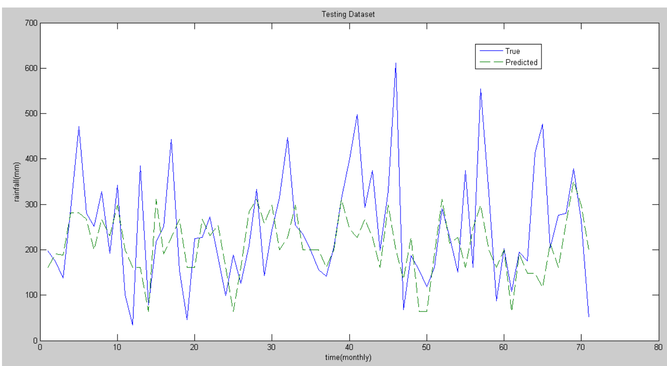

Figure 9. Rainfall forecast for monthly testing dataset.

For the monthly dataset, the table of detailed true and forecast values has been omitted due to space constraints.

F. Summarized Results

The table below shows the summary of MAPE and accuracy for all the four datasets.

TABLE VIII. SUMMARY OF RESULTS FOR ALL THE DATASETS

Type of Dataset	MAPE (%)		Accuracy (%)	
	Training Data	Testing Data	Training Data	Testing Data
Yearly	< 0.0001	5.74873	99.99	94.25
Bi-annually	1.872	18.8879	98.13	81.11
Quarterly	18.0421	23.971	81.96	76.03
Monthly	44.7867	43.9812	55.21	56.02

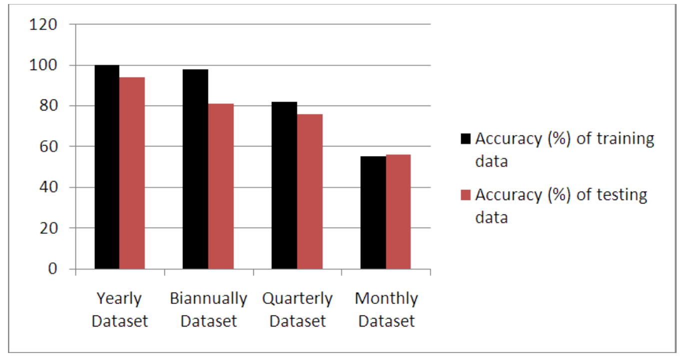

Figure 10. Comparison (bar graph) of accuracies of all the datasets.

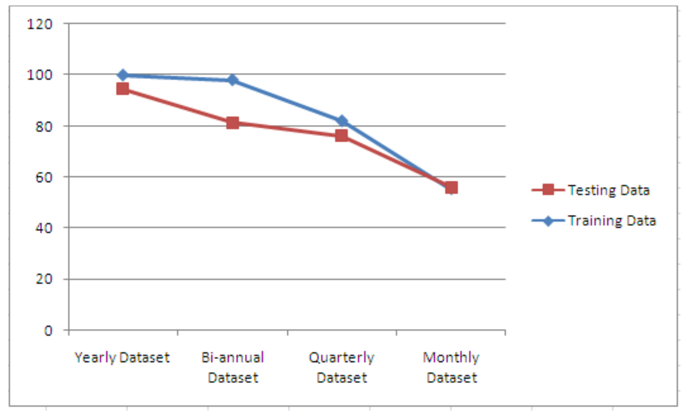

Figure 11. Comparison (line graph) of accuracies of all the datasets.

As depicted in Fig. 10 and Fig. 11, the model trained with the yearly dataset gives the highest accuracy for both training and testing data, followed by the biannual, quarterly and monthly datasets in that order. There is a clear relationship between the sampling interval of the input data and the resultant accuracy, i.e. the larger the sampling interval for the input data, the greater the accuracy of the model becomes. This could be due to the periodicity and dynamics of rainfall in nature and possible noises and distortions associated with the random fluctuations in the daily rainfall data. Considering the yearly rainfall values gives a more general (and often more useful) information than looking from a smaller scale, such as monthly or daily rainfall values.

V. CONCLUSION & FUTURE WORK

In this paper, focused time-delay neural networks were designed and implemented to build rainfall forecasting models using different datasets of yearly, biannually, quarterly and monthly-sampled values. For each of the dataset, 80% of the data was used for training and validation, and the remaining 20% was used for testing. Each of the datasets contains the rainfall level (measured in mm) during the period of January 1980 to May 2009 for a total of 29 years. Yearly rainfall dataset gives the most accurate results (94.25%) with the testing data. The forecast accuracies decrease for the biannual, quarterly and monthly datasets in that order (81.11%, 76.03% and 56.02% respectively.) As future work, additional weather parameters such as temperature, relative humidity, sunshine and cloud cover which affect the rainfall could be incorporated to the neural networks for more accurate rainfall forecasts.

Furthermore, the forecast accuracies could be greatly improved by using additional rainfall data during training.

ACKNOWLEDGMENT

We extend our thanks to Malaysia Meteorological Department (MMD) for providing us with the daily rainfall dataset that has been used in this paper. The research was carried out at MIMOS Berhad, Malaysia.

REFERENCES

[1] N.Q. Hung, M.S. Babel, S. Weesakul, and N.K. Tripathi, "An artificial neural network model for rainfall forecasting in Bangkok, Thailand," *Hydrology and Earth System Sciences Discussions* , vol. 5 , 2008, pp. 183-218 .

[2] N. Chantasut, C. Charoenjit, and C. Tanprasert, "Predictive Mining of Rainfall Predictions Using Artificial Neural Networks for Chao Phraya River," Proceedings of Joint Conference The 4th International Conference of The Asian Federation of Information Technology in Agriculture and The 2nd World Congress on Computers in Agriculture and Natural Resources August 9-12, 2004.

[3] R. Kuligowski and A. Barros, "Experiments in short-term precipitation forecasting using artificial neural networks," *Monthly weather review*, vol. 126, 1998, pp. 470-482.

[4] D. Silverman and J. Dracup, "Artificial neural networks and long-range precipitation prediction in California," *Journal of applied meteorology*, vol. 39, 2000, pp. 57-66.

[5] K. Mar and T. Naing, "Optimum Neural Network Architecture for Precipitation Prediction of Myanmar," World Academy of Science, Engineering and Technology; Volume 48, December 2008.

[6] N. Srikalra and C. Tanprasert, "Rainfall Prediction for Chao Phraya River using Neural Networks with Online Data Collection," Proc. the 2nd Indonesia-Malaysia-Thailand Growth Triangle (IMT-GT) Regional Conference on Mathematics, Statistics, and Applications, Vol. 4 (Computer Science Applications), pp. 31-38.

[7] A. Query, A. Abraham, and D. Steinberg, "Is Neural Network a Reliable Forecaster on Earth?," Proceedings of the 6th International Work-Conference on Artificial and Natural Neural Networks: Bio-inspired Applications of Connectionism-Part II, p.679-686, June 13-15, 2001

[8] J. Connor, R. Martin, and L. Atlas, "Recurrent neural networks and robust time series prediction," *IEEE Transactions on Neural Networks*, vol. 5, 1994, pp. 240-254.

[9] T. Hall, H. Brooks, and C. Doswell III, "Precipitation forecasting using a neural network," *Weather and Forecasting*, vol. 14, 1999, pp. 338-345.

[10] K.C. Luk, J.E. Ball, and A. Sharma, *An Application of artificial neural networks for rainfall forecasting*, Sydney: University of New South Wales - Water Research Laboratory, 1998.

[11] M. Adya and F. Collopy, "How effective are neural networks at forecasting and prediction? A review and evaluation," *J. Forecasting*, vol. 17, 1998, pp. 481-495.

[12] F. Lin, X. Yu, S. Gregor, and R. Irons, "Time Series Forecasting with Neural Networks," *Complexity International*, vol. 2, 1995, pp. 1320-0682.

[13] J. Tayman and D. Swanson, "On the validity of MAPE as a measure of population forecast accuracy," *Population Research and Policy Review*, vol. 18, 1999, pp. 299-322.

[14] S. Rayer, "Population forecast accuracy: does the choice of summary measure of error matter?," *Population Research and Policy Review*, vol. 26, 2007, pp. 163-184.

[15] P. Goodwin and R. Lawton, "On the asymmetry of the symmetric MAPE," *International Journal of Forecasting*, vol. 15, 1999, pp. 405-408.