Power Monitoring and Forecasting Using ANN with Utilisation of a Gateway Server and Digital Power Meter

Ahmed Visam Thaufeeq
School of Engineering
Asia Pacific University of Technology
and Innovation (APU)
Kuala Lumpur, Malaysia
tp038108@gmail.com

Chandrasekharan Nataraj
School of Engineering
Asia Pacific University of Technology
and Innovation (APU)
Kuala Lumpur, Malaysia
chandrasekharan@staffemail.apu.edu.my

Freddy Tan Kheng Suan
School of Engineering
Asia Pacific University of Technology
and Innovation (APU)
Kuala Lumpur, Malaysia
freddy.tan@staffemail.apu.edu.my

Abstract—**The power monitoring and forecasting system using the ANN technique is presented. It focuses on the implementation of ARIMA, LSTM, and ANN networks-based power monitoring and forecasting. Three energy forecasting models are simulated and tested to evaluate the accuracy of the power monitoring and forecasting. ANN was trained using a Bayesian Regularization backpropagation algorithm. The ANN approach has proved to produce better results than ARIMA and LSTM with an RMSE value of 4.631.**

Keywords— Power monitoring, Forecasting systems, ANN-based monitoring, ARIMA based forecasting, LSTM.

I. Introduction

Electricity has one of the greatest impacts on the change of technology worldwide. There are many devices for measuring power quality. Smart meters can be used to measure the power quality of any electrical supply connected to it. To monitor the power quality parameters and disturbances, smart meters can be used. These meters can share a huge amount of data such as consumption and power quality parameters. It helps in many ways like making an accurate billing system, helps to maintain how much energy is used by each consumer and makes appropriate steps as consumption can be known, and many more [1].

Compared to traditional meters, the smart meters provided homeowners and companies a system that is very flexible for tracking the usage of energy. These meters incorporated communication interfaces and digital sensors for monitoring and controlling to be done automatically [2]. It is very important to monitor the power quality using the latest technology, which helps to detect, record, and prevent any sort of power quality problems [3].

Smart meters are one of the sources to handle a very large amount of data for monitoring and forecasting electricity usage. Smart meter data is essential for billing, network management, and energy management purposes [4]. In many countries, power quality has been regulated in many standards. Few examples are EN 50160, IEEE 1159, etc. It is an important challenge for the operators in the distribution field as the distribution grids shortly will have to follow these standards to supply electricity [5].

The Internet-based Power Quality Monitoring System (IPQMS) has been presented for analyzing the power consumption. IPQMS transmits data to the cloud and it can show power quality data in real-time. The system helped to monitor and measure the parameters of Power Quality (PQ) such as voltage sag, voltage swells, interruptions, etc. [6]. Another study has presented for better use of observed data for data analysis by considered several determinants. The determinants used in the result were weather, location, and floor area [7]. The work has presented in monitoring a significant number of measurements of power quality parameters. It had voltage events like interruptions, transients, voltage sags and swells, and continuous parameters like frequencies, flicker, unbalance, harmonics and inter harmonics, signal voltage, and RVC. The smart meter system is proposed to measure the effect of climate, building characteristics, occupants' behaviour and appliance stock.

The online condition monitoring of a large synchronous generator under a short circuit fault is of utmost importance to diagnose the defects at an early stage. The process of fault diagnosis involves the detection of the type of fault and its severity [8]. The realization of a low-cost system using Arduino and ethernet-based smart energy meter for power quality monitoring has been designed [9].

Another work presented a low-cost smart solution for real-time energy management in a smart grid. It provided several power meters that continuously monitored the loads which were connected by communicating with a Datalogger through a bus of powerline. It is used as a low-cost solution method using CA 8252 to analyze the quality of electrical energy in an industrial customer [10].

The researchers proposed a hybrid artificial neural network-based which forecasted a day for smart grids. It is proposed the comparative analysis between Artificial Neural Network (ANN), Autoregressive Integrated Moving Average (ARIMA) and an Adaptive Neuro-Fuzzy System [11].

A time-based condition monitoring system using Neural Network and Fuzzy Logic is developed for the power monitoring system. The AI systems were applied to predict the uncertainty in the power fluctuations.

The accuracy of both the AI systems were discussed and concluded that Fuzzy Logic is less efficient compared to AI techniques [12].

From the review on power quality analysis, it is observed that many works were done on harmonics which were one of the biggest factors for a low power quality. It was also found that voltage swells, and voltage dips occur rarely but when it does it have effects on the system. Continuous monitoring has been considered rather than getting data in time intervals. Continuous monitoring helps to identify problems much faster and helps to make decisions faster as well. Therefore, it is important to look into these parameters in more detail for a better and efficient usage of power in the electrical system. Most of the researchers used LabVIEW as a good power monitoring software as it is very flexible for the implementation. This work emphasizes the integration of gate server using artificial intelligence for power monitoring and forecasting.

II. BLOCK DIAGRAM AND OPERATION OF THE PROPOSED SYSTEM

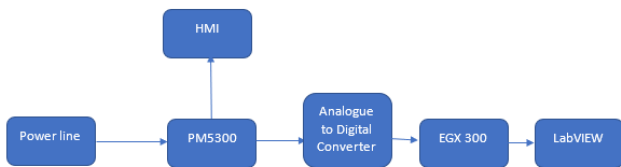### A. General Block Diagram

Fig. 1.   General Block diagram

The block diagram of the overall framework of the system is demonstrated as shown in Fig. 1. It uses a smart meter (PM5300) and a data logger (EGX300) system to analyze the electrical data. The smart meter was used to extract data from the three-phase power supply and the data logger is used to collect and store in the LabVIEW by using EGX300.

The source of power is taken from an electrical distribution box. Each phase (red, yellow, blue) was connected with a CT (current transformer) to decrease the current to be used with PM5300. To detect the voltage, 4 voltage probes are connected to the red, yellow, and blue phase as well as the neutral line.

A smart meter PM5300 is used to measure the power consumption. It is connected to the AC line from the electrical box by current transformers and voltage probes. As Malaysia follows the British standard of 230V in each phase to neutral and 415V is the voltage between two lines, the meter can be directly connected as it allows those range of voltages to be directly connected to the meter without any transformer to decrease the voltage. It has a normal operating range of 100V – 277V AC between line to neutral and a 415V AC between line to line taking +/-10% with those values. The meter has a response time of status input response time of 20ms which is an acceptable value. It is also able to take 64 samples per cycle. It can also take a few power quality measurements like minimum and maximum reading of THD This gives all the information for power analysis.

Human to Machine Interface (HMI) is located in the smart meter. The smart meter comes with a 128 * 128 resolution

monochrome graphics LCD. It has a four-button keypad which can be used to view all the features in the HMI. It has a pulse rate of 2.5 kHz for sensing and decoding purposes. EGX300 is the data logger used which is used to log data to the web application. It is connected to the smart meter with an RS485 connection. An ethernet cable is required to connect it to the desktop or laptop to gain the access to the web application. It has an input operating range of 24VDC (+/-10%).

The web application address was accessible when the data logger is connected to the laptop through an ethernet cable. The web application showed more detailed measurements that cannot be seen in the HMI of the meter. It was used to view graphs and record real-time trending values. However, it contained very less power quality measurements. LabVIEW was used to create graphs and show values that were not in the web application which was accessed by EGX300.
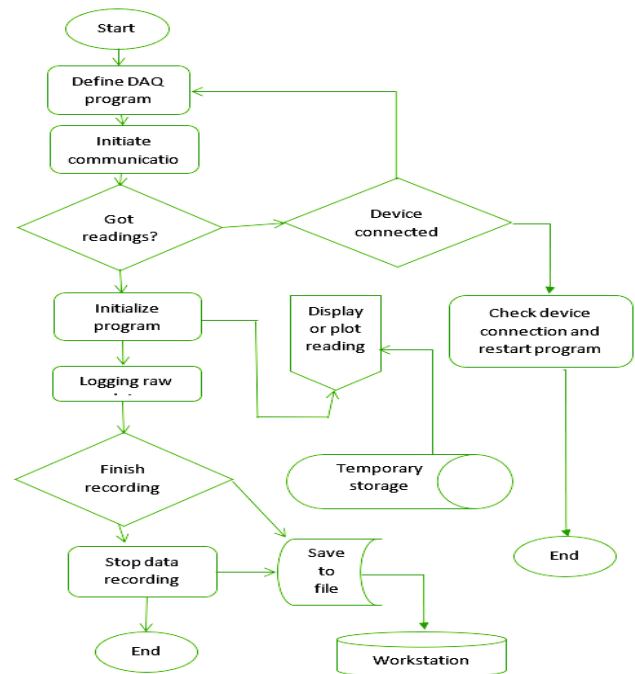
B. System Flow Chart:

Fig. 2.   System Flow chart

Fig. 2 shows the complete system flow chart. The program starts defining communication, port interfaces, functions, etc. After defining the program, serial connection between PM5300 and EGX300 and workstation get initiated. Raw data are fed. Once continuous data are attained, the raw data is converted to measurable units after the program is been initiated. The program performs all the raw data to measurable data conversion in the background and provides the data representation through a GUI. As the measurements are obtained, data logging is started and all the data are stored in the workstation. The data was also been displayed GUI. When the user stops the data collection, the data logging stops. After this, the logged data can be saved permanently as a file for later usage. The data can be exported to Microsoft Excel for a statistical analysis.

C. Constructional Diagram

Fig. 3 shows the constructional details of the power monitoring system. The power source, voltage, and current were received from a distribution box located at the APU FCET office. Each phase lines, (red, yellow, blue) were connected with a CT (current transformer) to decrease the current so that the current going to the smart meter PM5300 is safe. For voltage detection, 4 voltage probes were connected to each red, yellow, blue, and neutral line. The smart meter had voltage and current sensors inbuild. It detected the current and voltages and the smart meter analyzed these parameters and produces many other parameters such as power, harmonics, etc. All these values were shown in the Human to Machine Interface (HMI) on the smart meter. It showed the real-time values of the parameters which keeps getting updated in every second in a 128 * 128 resolution monochrome graphics LCD.
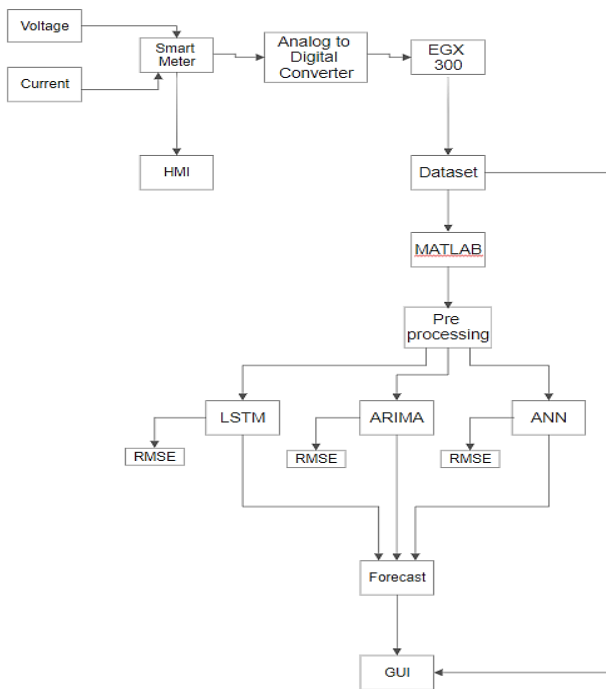

Fig. 3.   Constructional Diagram

An Analog to digital converter was used so that it can be connected to the data logger (EGX300). It took real-time reading which came from the smart meter and displayed it in a web application. An ethernet cable was required to connect the data logger to a laptop as it does not have a Wi-Fi feature.

The data logger collects and logs the data. It has its memory and the logged data is stored in its memory and displays in a web application. The data logger can be modified to collect the data in 5 minute, 15-minute, or 30-minute interval. For this project, it was set to collect the data at a 5-minute interval so that the data can be analyzed further. All the data are then collected and transferred to an excel sheet for more data analysis. These data sets are converted to hourly data and were exported to MATLAB since MATLAB was used to do more analysis of the data and produce the forecasting models. After creating each forecasting models (LSTM, ARIMA, ANN), the data were pre-processed before

putting the data into each model. For LSTM, only one variable was selected which is the energy delivered (kWh) data since this process is done univariately to do the forecasting. For ARIMA and ANN, multiple inputs and one target was selected to be able to forecast more accurately. After all the training and all the forecasting values were plotted, Root Mean Square Error (RMSE) between the predicted and the actual was calculated in each of the models. A GUI was made to show all the forecasted values and also to show all the data collected from the data logger.

Three months dataset is used in this work, which was obtained from the data logger connected to the smart meter. The smart meter gets the raw data values which are current and voltages. The data logger then gets all these readings and convert to different power and energy readings such as reactive power, apparent power, real power, energy delivered, energy received, etc. The forecasted variable chosen in this project is energy delivered (kWh). This was because it was the main power used by the electrical utility to calculate bills for the consumers. This range of data was used on all the models which were used in this project. The data used were from 2/6/2019 to 22/8/2019 date. This is around 3 months of data. The data collected was in a 5-minute interval and it was changed to hourly load data. The data was categorized into two parts which are inputs and targets. The output, which is used to forecast was energy delivered in kWh which is an accumulative power.

D. Neural Networks

The neural network learns from past data (training data), to predict the future values when relevant inputs are fed into the system. The advantage of the network is that it has adaptive learning compatibility, self-organizing, it also tolerance faults and it can do real-time operations. It was very useful as it can capture the non-linearity of the data and generalize it.

Feed-forward and feedback were the two-basic architecture of Artificial Neural Network (ANN). The feed-forward part of the system is where it carries the information from the input to the output in a forward direction via hidden layers. The layers were independent with each layer. Feedback carried the data in both directions such that each data in each layer is dependent on the output and input. It was more appropriate for time series and complex problems. Since the Artificial Neural Network forecast predicts the load, the input data was to be chosen by the variability of the data with each other.

The target value used to forecast was Energy Delivered (kWh) as that was the most important parameter needed to be forecasted since it is the parameter which all the electrical utilities and consumers refer to for billing the consumers, to make better decisions, etc. The neural network was done in MATLAB with the help of an inbuilt app. When the Neural Network Time Series app is opened from MATLAB, it showed which type of algorithm to choose. Nonlinear Autoregressive with External (Exogeneous) input or also known as NARX algorithm was selected. NARX is a recurrent dynamic network where it contains a feedback connection enclosing many layers of the network.

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

Y(t) is the future values of a time series from the past values which represents as "x(t)". Once the input and the target has been selected, the data was divided into three parts which are training, validation, and testing. 70% of the data was selected for training, 15% of the data was selected for validating and 15% of data were selected for testing as well.

    net.divideParam.trainRatio = 70/100;
    net.divideParam.valRatio = 15/100;
    net.divideParam.testRatio = 15/100.

In the NARX network, hidden neurons and delays need to be selected. The values of these are very crucial in the accuracy of the data. The NARX network had 2 inputs which are and external input and the other one is the feedback connection from the network output. But the feedback connection will be closed after the network is initially trained. Therefore, the data was trained a few times to get a higher accuracy. However, increasing the delays and hidden layer neurons could result in overfitting the data. Therefore, or input and output delay was set as 2 and the hidden layer neuron selected was 10.

    input delays = 1:2;
    feedback delays = 1:2;
    hidden layer size = 30.

After the selection of delay and hidden layer size, Bayesian Regularization back propagation algorithm was selected as the algorithm which gives a good generalization for datasets. Although it takes some time it was good to have higher accuracy and hence, the built-in Bayesian Regularization training function was used to perform the training of the feedforward ANN by backpropagation. This was a process that continuously updates the bias and weight values in the hidden layers of ANN. This was done to reduce the error output while using training inputs and target output. The target dataset consists of the actual load Energy Delivered (kWh) for a given predictor dataset.

    trainFcn = 'trainbr'

After the delays, hidden layers, training, validation, and testing values are selected, the data was set to train. The training was done in an open-loop. The training stopped until the error failed to decrease in the validation process at six iterations.

E. ARIMA

ARIMA was used to create the nonstationary linear time series model. ARIMA consisted of 3 parts. "AR" stood for Autoregressive; "I" stood for Integrate and "MA" stood for moving average. In the code, These 3 parts are shown as p, D,q where p defines AR, "D" defines "I" and q defines "MA" in ARIMA. ARIMA was built in MATLAB using the econometric modeler.

Non-stationary behavior and weekly seasonal pattern were observed in the data. These needed to be removed to get a stationary result or else ARIMA forecasting would not be as accurate. Therefore, non-stationary behavior was removed by subtracting the same data of the day before from each data of the original series using a differentiating operator. Hence a new series of data was created which was stationary because its mean was near zero.

The weekly seasonal pattern was removed by putting a seasonal differencing operator of period 168 which is the hours of a week. This helped in distinguishing the different days of the week in the ARIMA model. It was assumed that there were no holidays in the week. After all the tests were done, Autoregressive order and Moving Average order was defined as 1,0,1 respectively. After doing a prediction of the past values, forecast values were generated.

Long Short-Term Memory (LSTM) is capable of learning features from sequence data automatically. It also supports multivariant data as well as univariant data. In this project, only Energy Delivered (kWh) was used to forecast Energy Delivered (kWh) of future data. Univariate means having a single series of observations (1 variable) for the model. LSTM observes the past series of data and learns the trend which allows it to predict the next value sequence of data.

Before the univariate data is fed to the model, the data needs to be prepared. The data was divided into a sequence of multiple input/output pattern samples. In this case, 3 consecutive data were used as an input the next data which is one step after the 3 consecutive data was used as the output. This is called one sample. Likewise, 3 consecutive inputs and the output as the next data was selected and all the possible samples were made for the model to learn the trend. This trend was used to learn 1 step prediction. LSTM can learn the mapping of inputs to outputs. When the input and output values are increased, more step ahead predictions were obtained.

Finally, A GUI was made in MATLAB to show the results and plot graphs of the forecasts as well as all the data received from the data logger. The data was made to display in a time series. The forecasts with the past values with each model were made to display hourly. The data received from the data logger was made to show in a 5-minute interval.

III. PERFORMANCE TESTING AND SIMULATION RESULTS

The overall performance of the developed system has been evaluated by conducting various simulations using ANN and ARIMA platforms.

A. Simulation of different delays using ANN

ANN was tested with different delays data to forecast 24 hours data. For a measurement of accuracy, RMSE values were found to compare with the different delays. The following shows the inputs used in testing the network. These were taken as hourly data. It included 1933 sets of data.

    1-   Humidity   2-   Temperature
    3-   Reactive power   4-   Average current
    5-   Total Harmonic Distortion Current A
    6-   Total Harmonic Distortion Current B
    7-   Total Harmonic Distortion Current C

Each delay was then tested 5 times and taken the average. This is done because each the network is trained, it resulted in different values but the values were very close with each other. This happens due to different initial weight and bias values and different sets of data into the tests, validate, and training sets. As a result, the neural network can give close but different results on the same problem or the same amount of data sets. Therefore, to ensure a good accuracy of the system, it needs to be tested several times. In this case, it was tested 5

times with each delay and an average RMSE value was taken. The number of hidden neurons was set at 10. Fig. 4 shows the result in RMSE for each delay. By referring to Fig. 4, it was seen that using one delay decreases the RMSE value significantly. Moreover, it looks less time as well. When the delay was increased, it increases the time for the system to finish running and also it increased the RMSE value. It also observed that when it reached 4 delay value, the RMSE value skyrocketed. Therefore, for this dataset, a delay of 1 was found to be the best value for the network.
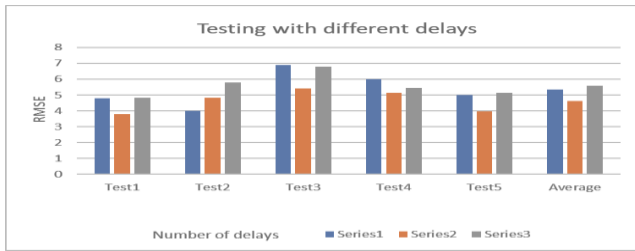

Fig. 4.   RMSE values for the different delays

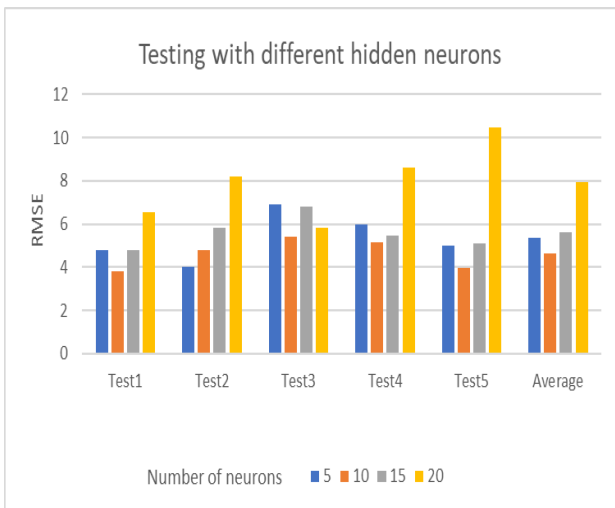### B.   Testing with different hidden neurons

Fig. 5.   Fuzzy Temperature Alert-Low working

ANN was tested with different hidden neuron layers data to forecast 24 hours data. Different levels of neurons are used by keeping the delay of 1 and the number of data set used as 1933. Fig. 5 shows the RMSE values for different number of hidden neurons. Each hidden neuron was tested 5 times by keeping the delay at 1. It was found that using the hidden neuron value at 10 gives an RMSE value the lowest on average. Hence the hidden neuron layer value of 10 was found to be the best value for the network.

C.   Simulation of different input data used in ANN

ANN was tested with different input data to forecast 24 hours of data. Temperature and humidity data were taken out from the input data. The following input data are used.
1-   Reactive power
2-   Average current
3-   Total Harmonic Distortion Current A

4-   Total Harmonic Distortion Current B
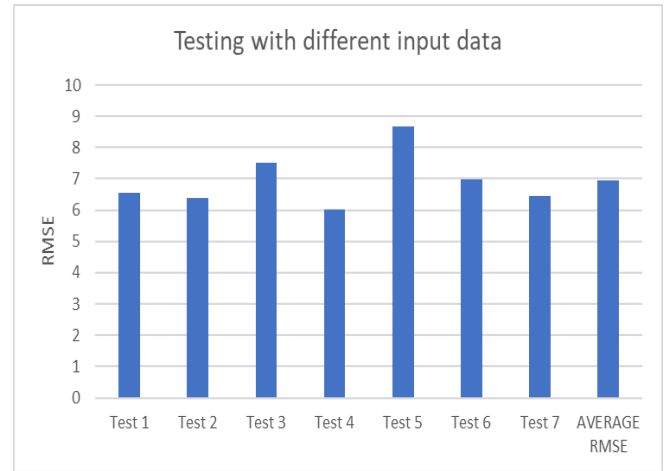5-   Total Harmonic Distortion Current C

Fig. 6.   Fuzzy Bar graph of RMSE values for the different input data

To test this data, the delay and hidden neurons are kept constant. Delay was set as 1 and hidden neuron value was set as 10 with the total number of data set as 1933. Fig. 6 shows the RMSE values for 7 tries without the temperature and humidity. It was used in 10 neurons RMSE so that it can be compared as both data used a delay with a value of 1 and the hidden neuron layer at a delay of 10. It was seen that when the temperature and humidity were taken, the RMSE value increased to 4.631 from 6.942. This showed the temperature and humid do affect the forecasting accuracy and the load depends on temperature and humidity as well.

D.   Accuracy test with LSTM and ARIMA

For LSTM, the input values are Energy delivered. It only uses one variable and learns the trend to forecast. 1933 data sets were used. The setting used for the network were as follows;

MaxEpochs = 100
GradientThreshold = 1
InitialLearnRate = 0.002
LearnRateDropPeriod = 125
LearnRateDropFactor = 0.2
Verbose = 0

When the model was run, it showed a huge RMSE value for a 24 ahead forecast. This can be due to the lack of good preparation of data. RMSE value showed 435 which is not an acceptable range. For this dataset, it can be concluded that the LSTM model code does not give a proper value hence the error is very high.

In ARIMA model, three different letters represent the parts of the ARIMA model. "p" represents the AR part which is the autoregressive part of the model. "D" is for the differentiating part and "q" is for MA part which is the Moving Average part of the model. The value of p,d,q values are set in the model as 6,5,5 respectively. When the model is run, it showed a huge RMSE value.

When the ARIMA model is run, it showed an RMSE value of 25898 which is also not a good value. Hence, it can be

concluded that the used ARIMA model is not useful for the forecasting data.

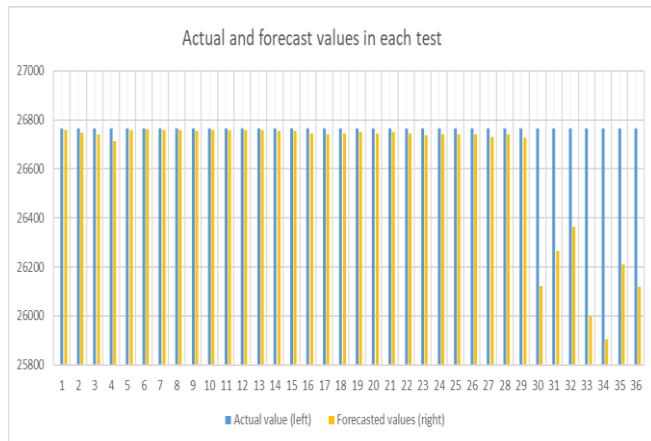*E. Performance Evaluation*

Fig. 7.   RMSE values between forecasted and actual data

The forecasting values were compared with the actual values for the performance evaluation. The results were taken at the last point where the data were forecasted as in at the 24th hour which is the last hour of the forecasted and the actual value. Fig. 7 shows the difference between the actual and the forecasted values at the 24th hour of the forecasted. Fig. 9 shows the difference and variance between actual and the forecasted data. It can be observed that there is not much difference between the actual and forecasted values. However, these differences matter a lot and contribute significantly to the power monitoring and forecasting system. The lower the difference the better. It can be concluded that the values less than 7 are good which falls under an RMSE of 6.

IV. CONCLUSION

The development of a power monitoring and forecasting system using ANN with the utilization of a gateway server and a digital power meter is demonstrated. Three energy forecasting models were made to be compared with each other. For analyzing the data, the collected data was viewed in the GUI and then the data was analysed further to produce new parameters for the evaluation. It also presented the implementation of ARIMA, LSTM, and ANN networks. For LSTM only one variable was used which is Energy Delivered (kWh) to produce a 24 hour ahead forecast of the same variable. For ANN and LSTM, several inputs were selected to produce a 24 hour ahead Energy Delivered (kWh) result. The approach showed that ANN produced better results than ARIMA and LSTM. It produced a value of 4.631 RMSE. It also observed the parameter variations and compared the results of theoretical and simulation results.

REFERENCES

[1] A. Kavousian, R. Rajagopal, and M. Fischer, "Determinants of residential electricity consumption: Using smart meter data to examine the effect of climate, building characteristics, appliance stock, and occupants' behavior," Energy, 55(0):184 – 194, 2013.

[2] A. Ahmad, N. Javaid, A. Mateen, M. A, Z. A. Khan, "Short-Term Load Forecasting in Smart Grids: An Intelligent Modular Approach", Energies, 2019.

[3] A. Cristian, M. Octavian Popescu, "System used for power quality monitoring in a distribution grid", In 2014 16th International Conference on Harmonics and Quality of Power (ICHQP), pp. 357-359. IEEE, 2014.

[4] G. Aurilio, D. Gallo, C. Landi, M. Luiso, "A Low-Cost Smart Meter Network for a Smart Utility", EEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, pp. 380–385, 2014.

[5] H. Das, L.C. Saika, "Ethernet-based smart energy meter for power quality monitoring and enhancement, Recent Developments in Control, Automation & Power Engineering (RDCAPE), 2017.

[6] H. Nguyen and C. K. Hansen, "Short-term electricity load forecasting with Time Series Analysis," 2017 IEEE International Conference on Prognostics and Health Management (ICPHM), Dallas, TX, pp. 214-221, 2017.

[7] J. Buitrago, S. Asfour, "Short-Term Forecasting of Electric Loads Using Nonlinear Autoregressive Artificial Neural Networks with Exogenous Vector Inputs", IEEE Transactions on Sustainable Energy, 3(1), pp. 1-24, 2017.

[8] L.C.M de Andrade, and I.N. da Silva, "Very short-term load forecasting based on ARIMA model and intelligent systems", 15th International Conference on Intelligent System Applications to Power Systems (ISAP '09), pp. 1-6, 8-12 Nov 2009.

[9] M. Music, N. Hasanspahic, A. Bosovic, D. Aganovic and S. Avdakovic, "Upgrading smart meters as key components of integated power quality monitoring system," IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), pp. 1-6, 2016, Florence, Italy.

[10] M. Samhouri, A. Al-Ghandoor, S. Alhaj Ali, I. Hinti, W. Massad, "An Intelligent Machine Condition Monitoring System Using Time-Based Analysis: Neuro-Fuzzy Versus Neural Network" Jordan Journal of Mechanical and Industrial Engineering, Vol. 3(2009), No.4, 294-305.

[11] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, ''Short-term residential load forecasting based on LSTM recurrent neural network,'' IEEE Trans. Smart Grid, vol. 10, no. 1, pp. 841–851, Jan. 2019.

[12] F. L. Quilumba, W.-J. Lee, H. Huang, D. Y. Wang, and R. L. Szabados, "Using smart meter data to improve the accuracy of intraday load forecasting considering customer behavior similarities," IEEE Trans. Smart Grid, vol. 6, no. 2, pp. 911–918, 2015.