Computing, Performance and Communication Systems (2022) DOI: 10.23977/cpcs.2022.060105
Clausius Scientific Press, Canada ISSN 2371-8889 Vol. 6 Num. 1

Power Consumption in Wireless Sensor Network: A Machine Learning Approach

Hamid Ali Abed AL-Asadi[1,2], Reham Hasan[3], Mohammad Nassr[3], and Mohammad Anbar[3]

[1]Communications Engineering Department, Iraq University College, Basrah, 61004, Iraq.
[2]Computer Science Department, University of Basrah, Basrah, 61004, Iraq.
[3]Communication Technology Engineering Department, Tartous University, Syria.

Abstract

Power consumption in wireless sensor network is a serious issue as the location of the deployed sensors may prohibit a feasible power charging. This research work applies a machine learning technique in conjunction with cloud platform for enhancing the network life time of a wireless sensor network and making end-user experience more plausible. Raspberry Pi 3 model B has been used to create a private cloud in the proposed experiment and Arduino UNO to program the used wireless sensor network. Three machine learning techniques such as Time Series Prediction, Linear Regression and Artificial Neural Networks have been applied in the proposed work. Python with its different libraries and packages have been used in order to analyze the data on cloud resources. Dht22 sensors, Bluetooth & Wi-Fi shields have been used in the wireless sensor network. Results are very encouraging and suggests for its possible implementation in future wireless sensor network.

Keywords: Wireless Sensor Network (WSN), Cloud Computing, Virtual Machine (VM), Machine Learning (ML), Power Consumption.

1. Introduction

WSN suffers from limited resource capacity especially in terms of computing, power consumption, and storage [1]. Several algorithms have been suggested to improve the network performance of WSN at different levels (physical layer, routing layer, security…..etc.) [2][3]. However, most of these algorithms are resource hungry and find it difficult for its execution on sensor nodes. Cloud computing environment provides a platform of huge resources and services, offered over the internet. Cloud resources are usually in form of virtual machines and is suitable for implementation of those algorithms that are difficult in the resource bound physical domain of the WSN [4][5][6] .

Cloud computing can support WSN in terms of storage, processing and many other aspects. The integration between WSN and the cloud has emerged as an important scientific area of research.

1.2 The Problem

Power consumption is an important challenge in wireless sensor network which comprise of four components: Transmission power (P_{tr}), Receiving power (P_r), Processing power (P_{comp}), and Sensing power (P_s). Equation (1) represents the total power consumed in WSN as sum of all these power components.

$$P_{consu=}\ P_{tr}\ +\ P_r\ +\ P_s + P_{comp} \qquad (1)$$

It has been observed that 80% of power consumption in WSN is mainly during the transmission and receiving. Several routing algorithms have been proposed to reduce the power consumption in these networks [7][8].

In the integrated sensor-cloud environment, power consumption is mainly due to multi-hop connections while delivering data to the cloud. This implies that sensor nodes rarely play a role in communication. This in turn may lead to preservation of nodes energy, since energy is mainly consumed by the repeated transmission and receiving operations. Therefore in WSN in integration with cloud, the focus of power consumption must be mainly on a forwarding mechanism between WSN and the cloud.

The rest of this paper is organized as follows. Section 2 discusses the related work in this area of research. Sections 3 focuses on the proposed model and introduce the applied three ML techniques by proposing three different scenario of study and analysis of WSN power. Section 4 performs a comparative study with few related recent work in this area. Section 5 conclude this work and points some future work possible in this domain.

2. Related Work

The integration of WSN with cloud is a new area of research and many work have been presented to supplement this research. This topic was addressed with its challenges and advantages by S.K. Dash et al. [9], wherein the researchers have discussed the most important issue to be consider from the sensors' as well from the cloud side. In another work by L. P. Dinesh Kumar, a cloud was used to store sensing data which were captured from the physical domain and applied a filtering algorithm on this data using a neural network which is located within the cloud gateway. In addition, they implemented a compression algorithm at the sensor network gateway before sending data to the cloud [10]. Another work, produced by Shah and Khan, develops a framework in the previous study by offering strong security levels and guarantees [11]. A work published in the same year by Savas, O., Jin, G., and Deng, J, considered trust as the most important factor within this integration and introduces an efficient trust management system within four stages through the sensor-cloud environment [12]. Misra et al. published the first paper which dealt with the theoretical and mathematical modeling of cloud-sensor structure which wasn't covered by the previous studies in this domain. However, this study was limited to only homogenous networks [13]. Another work was published by Chatterjee et al. that expands the mathematical modeling to heterogeneous WSN that contain different types of sensor nodes through different regions [14].

A work, published in 2016, focused on diagnosing the problem of uploading and transferring the sensed data to the cloud within a specific time. Here, multiple mobile sinks have been suggested to collect data and reduce delay time in the delivery process by processing an algorithm Time Adaptive Schedule Algorithm (TASA) [15].

Most of the work, in this area, are limited to either homogeneous network or some other constraints for power saving in WSN. The proposed work applies machine learning approach in conjunction with cloud for better power management and at the same time giving end-users a complete flexibility.

3. The Proposed Model

In the proposed model, three scenario have been considered based on the three different ML tools used. These are as follows.

3.1 First Scenario

The first scenario is called the "traditional state", where two groups of **dht 22 sensors** are programmed using **Arduino UNO**. The network sent the sensed data to a base station for storing and processing purpose. It has been modified for other scenarios as follows.

In this, we studied WSN power during two hours of time. Samples have been taken every two seconds. "Visual Studio" is used to create a data set which gets its data through a connection between the Arduino and the base station. End-user, in this scenario, requests data with a specific sample rate and power parameters through a GUI.

3.2 Second scenario

In the second scenario, we tried to make own cloud10 as a virtual sink where the "end user GUI" is connected via URL address to access the data. In this, cloud handles the processing and the storing operations.

Data Visualization

Visual analysis is an important step in order to explore data and extract some ideas/indications out of these. Temperature has been visualized, during an entire climate cycle, and is shown in figure 1.

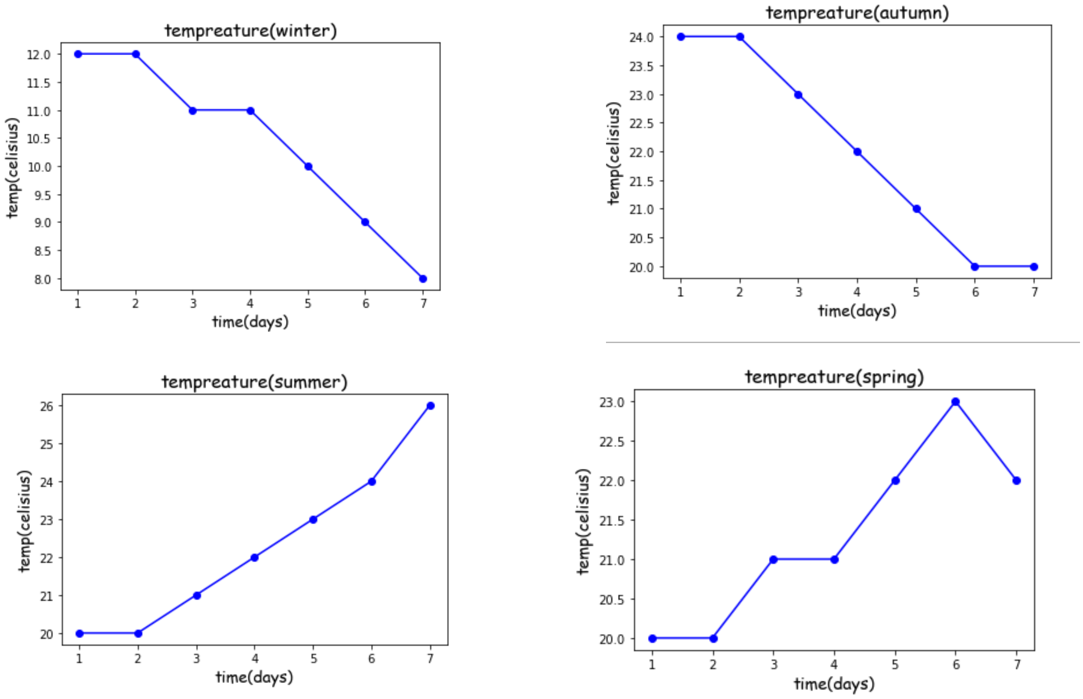

Fig .1 Sample of temperature during an entire climate cycle

Data Cleaning

Data cleaning ensures that there are no outliers or null values as shown in figure 2 or variables that have no predictive values. Machine learning model wouldn't perform its job properly in the presence of these values. Many reasons lead to such values in a dataset. It might be due to experimental fault, data variation, malicious behavior, unexpected change etc.

```
21       False
22       False
23       False
24       False
25       False
26       False
27       False
28       False
29       False
...        ...
12702    False
12703    False
12704    False
12705    False
12706    False
12707    False
12708    False
12709    False
12710    False
12711    False
12712    False
12713    False
```

predict	temp	time	
25.0	24.0	1.0	0
25.0	24.0	1.0	1
25.0	24.0	1.0	2
25.0	24.0	1.0	3
NaN	NaN	NaN	4

Fig.2 Data Preprocessing

We have studied WSN power during this scenario with the same previous conditions (within a minute and transmission also was every 2 seconds) As a result the "forward mechanism" to the cloud platform must be considered as a critical point for the WSN power.

3.2.1 Time Series Forecasting

A time series is a collection of observations taken during a given period of time, and the sensor readings are an example of these chains. Temperature and humidity can often be predicted especially in the places where these values remain stable for a long period of time. Cloud computing is a suitable execution environment for the implementation of these algorithms.

Therefore, in order to do the predictions through time series in order to reduce the number of receive and transmit operations to maximum possible extent by handling with only meaningful readings, the proposed model acts as per the following steps.

3.2.2 Data Manipulation and Smoothing

In order to manipulate the data, we dealt with three main time series components as follows.

- ✓ *Seasonal component at a specific time (S_t):* It is a very important factor for making future charts. It includes seasonal fluctuations during a changing year or season.
- ✓ *Irregularity component at a specific time (I_t):* Random deviations at a given moment of time. These deviations, in the time series, are caused by some unexpected influences that are unusual and do not occur during specific periods. Also, there are no precisely defined statistical techniques for identifying these random effects.
- ✓ *Slite variation component (s_t):* Soft variations in data that are constant for a long period of time.

The "data smoothing" means average extraction in order to take only the meaningful data. Therefore, Mean Average and Central Mean Average have been extracted besides S_t, I_t. This step is followed by a step called "Deseasonalize" which included dividing real data by seasonal components for calculating "trend coefficients" which only requests S_t extraction.

3.2.3 Linear Regression

Regression is a ML technique for classification. Regression is about the conditional distributions. The goal of regression is to understand "as far as possible with the available data how the conditional distribution of some response **y** varies across the sub-population determined by the possible values of the predictor or predictors (x)". There are various regression models as some special case of generalized linear model. These are normal (linear) regression, binomial regression, Poisson regression, or some other less common forms.

Linear Regression, as given in equation (2), has been applied in this work where ε is a random error component, x is a dependent variable, y is an independent variable. The goal is how far x can give us a prediction about y. In the normal case, its mean of distribution is assumed to be zero. It has been noticed that the trend of visualized data is linear over the time, as shown in figure 2.

$$y = a + bx + \varepsilon \qquad (2)$$

3.2.4 To Ensure Performance

A technique, called Analysis of Variance technique (ANOVA), is used to have an idea on how much a model is appropriate for the prediction and whether there is a linear relation between variables or not. Many other information is exhibited from this summarization. For example, coefficients p-value must not be zero in order to have an effect in the proposed model. Therefore a specific condition must be satisfied (*p-value < 0.05*) as shown in figure 3. The predictor would be meaningful i.e. smaller *p-value* indicates the decision is more close to reject the null-hypothesis (H_O) and accept alternative hypothesis (H_A) which is a linear relationship. On the other hand, higher p-value indicates that the decision is closer to accept the hypothesis (H_O) and there is no linear relationship between the variables.

3.2.5 The Proposed Model in this scenario

The following flowchart describes the proposed model. Data is visualized and then fed into machine learning model. Before this, data should be cleansed and should be made suitable for the training. "Data cleansing" or "Data manipulation" includes dealing with the null values and fixing the outliers. The outliers may destroy the regression outcome due to the sensitivity of the model to the outliers.

Next step is "Data smoothing" which includes finding the mean and the central mean (MA(6), CMA(6)) of data during six rounds in one day in order to take the most important data slices.

Three basic components of the time series (S_t, I_t, s_t), as explained previously, are helpful to make the prediction over the time series where the original data is calculated as given in equation 3. The previous steps are followed by the "Deseasonalize". Finally, the linear regression is fed with data and the model is trained by dividing the dataset into 80% as a training data and 20% as testing data.

$$Y_t = S_t \times I_t \times T_t \qquad (3)$$

In the considered case, H_O was rejected and a linear relationship between variables is found. Therefore, linear equation could be used to calculate the "trend coefficient" which is the most important component in this prediction. It is used to reach the prediction values after it was multiplied with seasonal components $(S_t \times T_t)$. Finally, temperature values of next two days can be predicted depending on the previous two days temperatures. Figure 3 shows the flowchart of the proposed model.

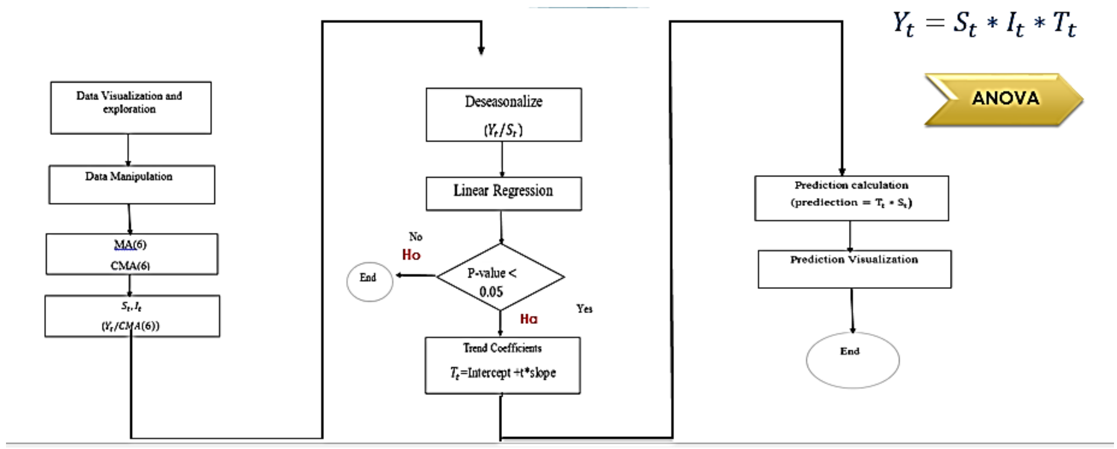

Fig.3 The Flowchart of the proposed model

3.2.6 Prediction

With the previous proposed model, we are able to predict the temperature for the next two days based on the data of the previous two days as shown in figure 4. For doing so, a forwarding mechanism has been modified to be every 4 hours at a command from the cloud platform to the Arduino considering Arduino as the slave and Cloud as the master.

Accuracy of the model is 84% with about 9186 samples divided into 7333 as training Data and 1835 as test Data. By increasing the number of samples to include a complete climate cycle, the accuracy has been improved up to 89%.

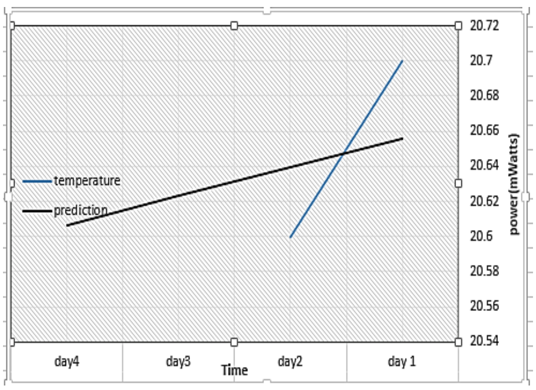

Fig.4 The temperature prediction

3.3 Third Scenario

In this scenario, we tried to make a prediction on the dataset using Neural Network as a deep learning model. We performed a comparison between neural network and linear regression for making prediction over the dataset.

3.3.1 Artificial Neural Networks (ANN)

In recent, ANN has evolved as a promising AI tool for many real word applications. ANN consists of a simple system with highly connected neurons which processes the information. ANN is used to derive a relationship between a set of input parameters and their output responses [16].

The main characteristics of the ANN includes its capacity for learning by example. Once the ANN learns the desired function, it will be able to calculate the output on the given input. ANN is composed of number of interconnected neurons as the processing elements that are joined by the weighted connections. The training algorithm adjusts the connections weights through an iterative procedure minimizing the error. The use of ANN has envisaged new perspectives since they do not hypothesize on data distribution [17].

3.3.2 The Proposed Model in this Scenario

Various steps in this model are as follows.

Data Preprocessing

In data preprocessing, following is done.

- Read the file in CSV (comma separated values) format and convert it to arrays.
- Split the dataset into the input features (call x) and the label (call y).
- Scale the data (call *normalization*) so that the input features have similar orders of magnitude.
- Split the dataset into the training set (7000 samples), the validation set (1500 samples) and the test set (1500 samples).

The Architecture setup

We set up the architecture by the following three layers.

- Hidden layer 1:32 neurons, ReLU activation
- Hidden layer 2:32 neurons, ReLU activation
- Output Layer: 1 neuron, Sigmoid activation

Filling in Template

As we need to find the best numbers for the template, we configure the model before the training as follows.

- The algorithm for optimization (stochastic gradient descent)
- The Loss function (binary cross entropy)
- The metrics to observe apart from the Loss function (accuracy)

The function is called ‘fit’ as we intend to fit the parameters to the data. Training data is specified as *X_train and Y_train*. Then, the size of the mini-batch is specified besides how long one has to train it for epochs. Finally, the validation data is specified so that the model would exhibit on how the model is doing on the validation data at each point. The output of this function would be a history to be saved under the variable hist. This variable is used a little later for visualization purposes.

4. A Comparative Study

This section performs a comparative study with recent researches in this area on the experimental results of the previous scenarios.

The work by Misra et al. [13], which ignored the forwarding mechanism to the cloud, was dedicated for homogeneous networks assuming that monitoring the consumption of power is within months. The cumulative power consumption of the Misra model was about $8\times 10^{8}\ mj$, which became about $4\times 10^{8}\ mj$ when integrated with a CSP (Cloud Service Provider) as shown in figure 5. Another study introduced by Chatterjee et al. focused on optimal formation of Virtual Sensor (VS_s). In this work, an algorithm was proposed for efficient virtualization of the physical sensor nodes and the optimal Composition Of Virtual (COV) sensors within the same geographic region (COV-I) and spanning across multiple regions (COV-II). Experimental results demonstrate that compared to existing strategy of optimal composition of (VS_s), (COV-I) improved the cumulative energy consumption and the network lifetime by 34.9% and 61.04% respectively. Moreover, (COV-II) enhanced the parameters by 68.4% and 29.59% respectively.

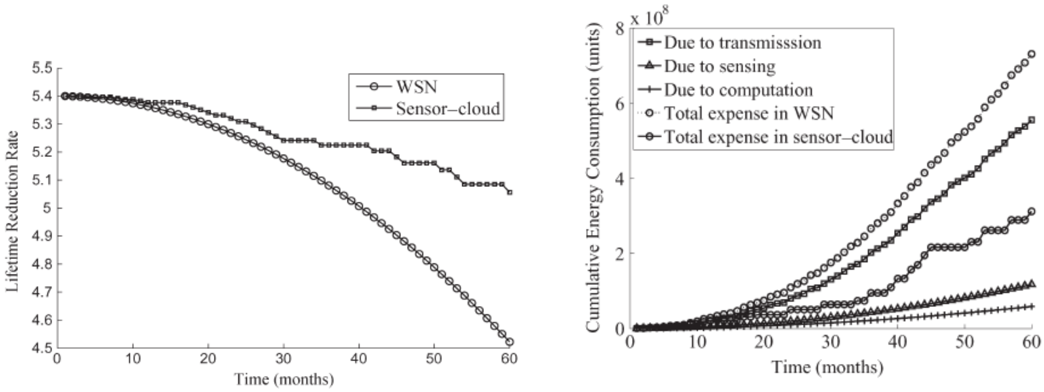

Fig.5 WSN vs. Sensor-cloud life time (energy consumption) [13]

4.1 Experimental Results

Unfortunately, it is noticed that the power state in the second scenario got worse than the first scenario.

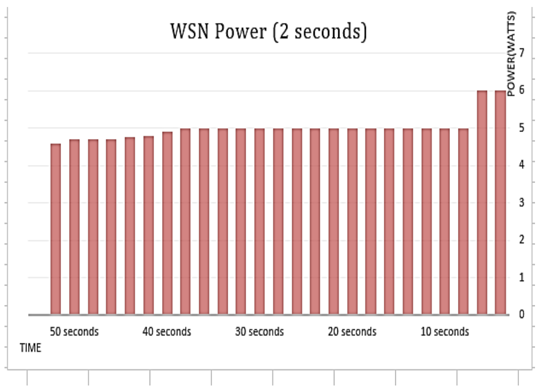

Fig.7 WSN-Cloud Power during minute

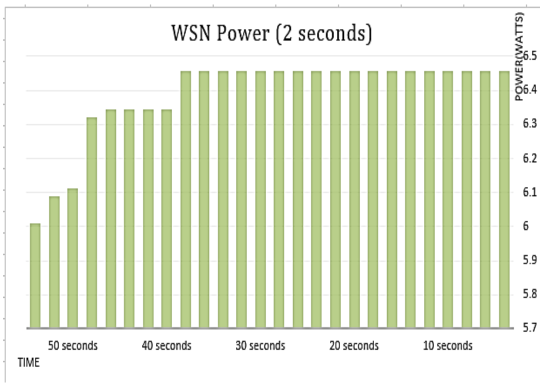

Fig. 6 WSN-Power during minute

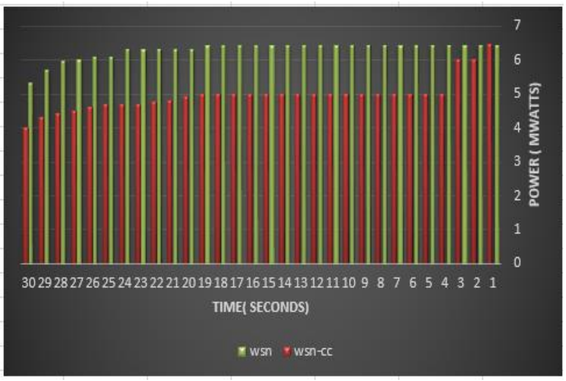

Fig.8 WSN vs. WSN-Cloud Power

The WSN power has been studied after applying a time series forecasting in the second scenario (deploy a machine learning model) during two days, while we expressed one day by (p1, p2, p3,p4, p5, p6) as shown in figure 9.

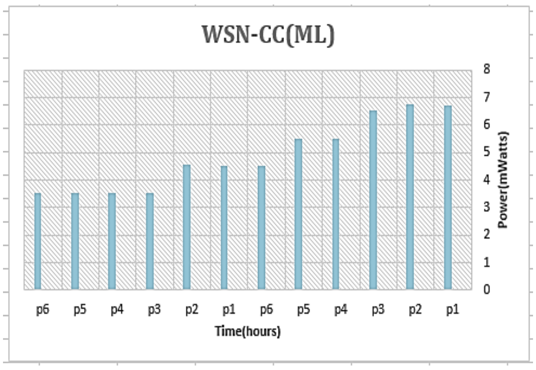

Fig.9 : WSN-Cloud (Machine learning model) Power

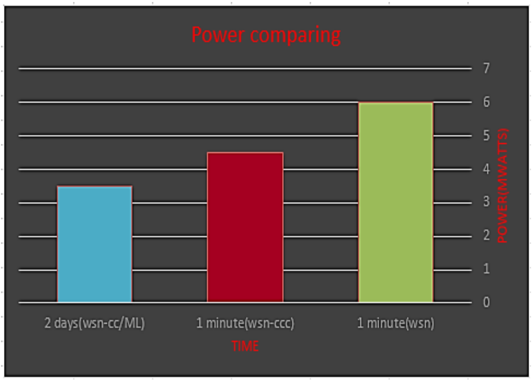

Fig.10 Energy Comparison in three scenarios

In the third scenario, it is noted that the ANN presented high accuracy as shown in Figure 11. It explains that the accuracy has increased during the training process (about 90%) with about 9000 samples of which 70% are training data, 30% are test and validation data. However, it is noticed that the training process took more time than it took with linear regression training.

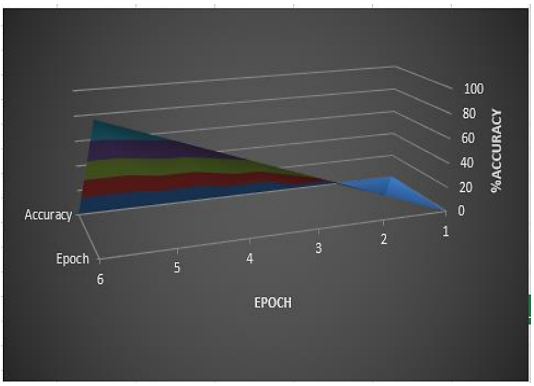

Fig.11 Neural Network Accuracy

5. Discussion and Conclusion

Because of the narrow band connection of the sensors compared to the wide band cloud connection, we may get the reverse of the expected result from integration of cloud to WSN as shown in figure 7. Contrary to expected, the first scenario was better as shown in figure 8. Therefore, data analysis processes are applied using python as a basis to reduce the number of transmissions as much as possible for preserving the WSN power. First, Linear Regression as a machine learning tool is used and the performance of the model is analyzed. The Coefficient of Determination or R-square was equal to 0.90, meant that 90% of the variation of y-values around the mean are evaluated linearly by the x-values. In other words, 90% of the values fit the model. Another important indicator was Multiple R or correlation Coefficient, which was equal to 0.95 in the proposed model (95%).

This model was adaptive. In other words, the forwarding mechanism changes based on the prediction process. It is found that WSN power in traditional state was 12 mWatts at the beginning of simulation which decreased to 6 mWatts after 1 minute of transmission (figure (6)). It further decreased to 4.5 mWatts when it was integrated with cloud platform considering the same previous conditions of the transmission rate and period as shown in figure (7). Then, a machine learning model is deployed through the cloud platform and the “forwarding mechanism" is modified based on the prediction accuracy of this model**.** With this, the WSN power decreased to just 3.6 mWatts after two days of transmission as shown in figure (9) which is much better as compared to the previous scenarios shown in figure (10).

We moved to deep learning applying ANN for the prediction over the dataset. It gave higher accuracy comparing with linear regression.~~,~~ Though linear regression produced good prediction in less time. However, linear regression works better only when the linear regression equation is the best fit to the available dataset.

Thus, it is observed that in a cloud based data transmission of WSN, by paying attention to forwarding mechanism and deploying a machine learning model in the cloud platform, it is possible to efficiently manage the energy consumption of wireless sensor network.

Future work will include the study on the effect of using ANN and other machine learning tools on WSN performance by the integration of a public cloud.

References

[1] Sheng, Z., Mahapatra, C., Zhu, C., & Leung, V. C. (2015). Recent advances in industrial wireless sensor networks toward efficient management in IoT. *IEEE Access*, *3*, 622-637.

[2] Navarro, M., Li, Y., & Liang, Y. (2014, June). Energy profile for environmental monitoring wireless sensor networks. In *2014 IEEE Colombian Conference on Communications and Computing (COLCOM)* (pp. 1-6).

[3] Patel, M., Aggarwal, A., & Chaubey, N. (2019). Detection of Wormhole Attack in Static Wireless Sensor Networks. *Advances in Computer Communication and Computational Sciences* (pp. 463-471). Springer, Singapore.

[4] Agrawal, A., & Kaushal, S. (2015, September). A study on integration of wireless sensor network and cloud computing: requirements, challenges and solutions. In *Proceedings of the Sixth International Conference on Computer and Communication Technology 2015* (pp. 152-157).

[5] Zhang, Y., Su, H., Yang, M., Zheng, D., Ren, F., & Zhao, Q. (2018). Secure DE duplication Based on Rabin Fingerprinting over Wireless Sensing Data in Cloud Computing. *Security and Communication Networks*.

[6] Luo, M., Luo, Y., Wan, Y., & Wang, Z. (2018). Secure and efficient access control scheme for wireless sensor networks in the cross-domain context of the IoT. *Security and Communication Networks*.

[7] Albath, J.(2008). Energy efficient clustering and secure data aggregation in wireless sensor networks. *MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY* 2008.

[8] Kulkarni, S. S., & Bhoi, A. D. (2017). Security and timing analysis of hybrid algorithm for ZigBee in wireless sensor network. *International Journal of Engineering Science*, Volume *13506*.

[9] Dash, S. K., Sahoo, J. P., Mohapatra, S., & Pati, S. P. (2012, January). Sensor-cloud: assimilation of wireless sensor network and the cloud. In *International Conference on Computer Science and Information Technology* (pp. 455-464). Berlin: Springer.

[10] Kumar, L. D., Grace, S. S., Krishnan, A., Manikandan, V. M., Chinraj, R., & Sumalatha, M. R. (2012, April). Data filtering in wireless sensor networks using neural networks for storage in cloud. In *2012 International Conference on Recent Trends in Information Technology* (pp. 202-205). IEEE.

[11] Shah, S. H., Khan, F. K., Ali, W., & Khan, J. (2013, March). A new framework to integrate wireless sensor networks with cloud computing. In *IEEE Aerospace Conference* (pp. 1-6). IEEE.

[12] Savas, O., Jin, G., & Deng, J. (2013, May). Trust management in cloud-integrated wireless sensor networks. *International Conference on Collaboration Technologies and Systems (CTS)* (pp. 334-341). IEEE.

[13] Misra, S., Chatterjee, S., & Obaidat, M. S.,et al. (2014). On theoretical modeling of sensor cloud: A paradigm shift from wireless sensor network. *IEEE Systems journal*, *11*(2), 1084-1093.

[14] Chatterjee, S., & Misra, S. (2015, June). Optimal composition of a virtual sensor for efficient virtualization within sensor-cloud. In *2015 IEEE International Conference on Communications (ICC)* (pp. 448-453). IEEE.

[15] Wang, T., Li, Y., Wang, G., Cao, J., Bhuiyan, M. Z. A., & Jia, W. (2017). Sustainable and efficient data collection from WSNs to cloud. *IEEE Transactions on Sustainable Computing*.

[16] Barbosa, L. C. M., Gomes, G., & Junior, A. C. A. (2019). Prediction of temperature-frequency-dependent mechanical properties of composites based on thermoplastic liquid resin reinforced with carbon fibers using artificial neural networks. *The International Journal of Advanced Manufacturing Technology*, *105*(5-6), 2543-2556.

[17] Bocco, M., Willington, E., & Arias, M.,et al. (2010). Comparison of regression and neural networks models to estimate solar radiation. *Chilean Journal of Agricultural Research*, *70*(3), 428-435.