P-ISSN 2586-9000
E-ISSN 2586-9027
Homepage : https://tci-thaijo.org/index.php/SciTechAsia
Science & Technology Asia
Vol.25 No.4 October - December 2020
Page: [81-92]

Original research article

Data Prediction and Forecasting Techniques Based Energy Efficient Cloud-integrated Sensor Network

Kalyan Das*, Satyabrata Das

Department of Computer Science & Engineering,
Veer Surendra Sai University of Technology, Burla 768018, India

Received 14 November 2019; Received in revised form 2 August 2020
Accepted 4 August 2020; Available online 24 December 2020

ABSTRACT

An Energy-efficient Sensor cloud is a challenging task due to the limited life span battery of the sensor and massive consumption of power at the data center. An energy-efficient cloud-integrated sensor network model is proposed using the combination of the artificial neural network (ANN) based prediction model with two activation functions and forecasting methods using Autoregressive Integrated Moving Average (ARIMA). The ANN model analyses the nonlinear components. After that, ARIMA examines the linear part utilizing the output of ANN and input data. The combination of ANN and ARIMA provides better results for modeling of nonlinear and linear patterns one by one using the separate models, and then merging the forecast to enhance the performance. In our model, we have first used the ANN model, having two activation functions, which predicts the temperature for a given hour with an accuracy of 94%. Then we used the results of ANN along with the previous real temperature as an input to the ARIMA forecasting model. In traditional approaches, all the user's requests must redirect to the sensor network, which consumes more energy as the requests of the users are very frequent. In our approach, the sensor communicates with the cloud every 5 hours. Most of the user's requests replied at the cloud system itself by using the combination of prediction using a neural network model having two activation functions and the forecasting method with an accuracy of 98%, which results in less communication and more battery life of the sensor. The proposed method consumes less energy as compared to the traditional techniques for the sensor cloud environment as per our simulation.

Keywords: Artificial neural networks; Sensors; Clouds; Data prediction; Wireless sensor network

***Corresponding author**: kalyandas1983@gmail.com **doi:** 10.14456/scitechasia.2020.50

1. Introduction

A Wireless Sensor Network (WSN) consists of many sensors that communicate with each other wirelessly. Cloud computing is an emerging technology where the end-users get resources on rent from cloud service providers. A Sensor Cloud is the integration of the sensor network and cloud system where end-users get resources from sensor networks using the cloud. The cloud service provider provides sensing as a service where the sensor owner adds sensors in the cloud so that the user can access them using the cloud platform. The cloud can give virtualization services where multiple users can share a sensor. An energy-efficient Sensor cloud is a challenging task due to the limited life span battery of the sensor and massive consumption of power at the data center for providing the resources. The primary motivation of the paper is to make sensor-cloud energy efficient.

In the traditional models, every user request is redirected to the WSN through the cloud. Every user request needs communication between the cloud and the sensor network. The energy consumption of a node is of the form $E = K_1 + K' r E_y d^{\alpha}$, where K_1 is energy consumption when the node is idle, K' = constant, r = data rate of the node, E_y = the energy required for successful decoding the sink [1]. Let the constant $K = K' E_y$. The energy consumption of a node by substituting values of K is of the form $E = K_1 + Krd^{\alpha}$ where α within the range from 2 to 4 depending on the environmental factor. Energy consumption in the sensor network depends on the rate and distance of data transmission. More data transmission between the sources to sink can cause the sensor node to dry. Sensors generally consume more power while transmitting data. Usually, user requests are frequent, and all the user's requests are redirected to the sensor network through the cloud system. As every user request needs communication between the cloud and the WSN, the energy consumption is more than in the traditional models. There is a need for prediction methods that predict future sensor data in advance within the cloud. So, there will be less communication between the cloud system and the senor network so that less energy will be consumed within the sensor network. Generally, ANN and ARIMA analyze the nonlinear and linear forecasts. Combining both, the model improves the performance as analysis of nonlinear and linear components is carried out one by one using the separate models.

In our model, the sensor communicates less frequently, and most of the user requests are replied to by the cloud system using ANN-based prediction and an ARIMA based forecasting model. This approach results in less commutation between the sensor network and the cloud system, and more battery life for the sensors. The related works are described in Section 2. Section 3 discusses the ANN-based prediction model, forecasting techniques using ARIMA, and the advantage of the combination of the ANN and the ARIMA model. The results of the combination of ANN and ARIMA models and the comparison of energy consumption in the proposed and traditional methods are explained in Section 4. The conclusion of the proposed model is described in Section 5.

2. Related Works

Various temperature prediction methods have been proposed [2-12]. In [2], a study was conducted to predict the land surface temperature of Jaipur, an Indian city. Moderate-Resolution Imaging Spectroradiometer & Advanced Spaceborne Thermal Emission and Reflection Radiometer sensors were used for it. The results of this study are beneficial to conduct studies on the urban heat island effect of any location. Papantoniou et al. [3] used the neural network method on the data obtained from some European cities to predict the outdoor temperature. By comparing predicted and measured outdoor temperature, the efficiency of the prediction was measured. The comparison made on the measured and predicted outdoor air temperature produced proper training of the neural network. In [4], the study was carried out with a method of using thermal imaging recognition for choosing the values and processing the images. The neural network technology was successfully applied to predict the minimum resolvable temperature difference. In [5], the study carried out was to learn the multilayer perceptron networks for predicting the maximum and minimum temperature by the past recorded temperature observations. The use of temperature gradient feature improves the temperature prediction with greater accuracy. In [6], the authors proposed a method to predict the daily mean water temperature. They compared it with the traditional modeling approach and utilized it for water resource managers as predictive tools. In [7], a simulation model of the cement kiln is used to find the best method to check the fuel consumption by the rotary kiln. The BP & Elman network-based cement kiln model produced faster convergence speed and high precision. In [8], a Back Propagation neural network model was introduced by establishing a temperature prediction model of the pavement in the winter season. The algorithm was improved with new prediction methods to get more accuracy in results. The methods used in this paper referred to predicting the temperature of pavements. In the article [9], using the temperature monitoring data of some gravity dams, the authors have proposed a mathematical modeling principle and partial least-squares regression method. The obtained results from the model have more accuracy, advanced computing, and more realistic explanations than the general least-squares regression method. 1n [10], the dependency of temperature is explained using the combination of backpropagation and genetic algorithm. In [11], for the uncertain temperature field prediction, the paper has proposed two methods, such as the first-order fuzzy perturbation finite element method and the modified fuzzy perturbation finite element method. These methods are useful for the prediction of the temperature. In [12], the authors developed a nonlinear predictive model for weather analysis using ANN. Also, the effectiveness of the models was examined to forecast maximum temperature throughout the year.

In the references [13-25], the authors proposed various energy-efficient methods in the sensor network and cloud computing. The authors in [13] [14] have proposed a data energy-efficient sensor cloud model in which the prediction method is used in the cloud system to save energy consumption in the sensor network. The authors used the logistic activation function in the Rprop-algorithm. The sensors are grouped into independent sets and the author in [15] maximized the number of separate groups with less rearrangement of sensors to minimize energy consumption. In [16], the authors analyzed various transport protocols for energy efficacy using a real-time system. In [17], the authors used tools for energy measurement and proposed optimizing input and output operations for energy saving. In [18], the authors used prediction techniques to predict computational requirements for future virtual machines to save energy in the cloud system. In [19], the authors minimized the network lifetime using a minimum spanning tree. Authors in [20] reduced power consumption in WSN by choosing backbone nodes using a heuristic approach. In [21], the authors analyzed the loss of power in the WSN in terms of the supply voltage. The authors in [22] proposed an energy-efficient protocol based on the maximization of coverage using sleep schedule techniques. In [23], the authors suggested a game algorithm-based packet forwarding technique that improves network lifetime. The author in [24] proposed an aggregation method that aggregates sensor data and forwards it to the cloud system with optimum bandwidth to minimize delay and cost. In [25], the authors proposed a method which selects the minimum number of server and performed load balancing among the server. As a result, energy consumption is less for the cloud.

Energy prediction methods were proposed in the papers [26-31]. Chaianong et al. [26] designed a forecasting model of photovoltaics, which expected nearly 14 percentage use of photovoltaic energy. The accuracy of the forecasting could be improved further. The research carried out by Krishna et al. in [27] used a time series analysis to predict future sensor data from the old data. This method reduced the communication and, as a result, saved energy. This model does not fit large scale datasets. Jemal et al. [28] predicted the monitoring values of Quality of Service (QoS) parameters, which reduced the transmission energy in the WSN. This approach was not implementable in a large area with a real-time environment. Xia et al. proposed [29] an energy-efficient body area network based on the prediction of transmitted data. This method has yet to analyze packet loss and latency of transmission. In reference [30], the authors proposed a prediction-based technique that predicted the remaining energy of nodes to balance energy consumption. This method was not implementable in real-time scenarios. The authors in [31] proposed a prediction-based approach in a sensor cloud environment which minimizes the energy usage in the sensor network. This technique did not optimize the QoS parameters.

From the literature survey carried out in this section, we are motivated to make Sensor Cloud energy efficient. There is a massive consumption of power in cloud data centers to provide storage, and the battery life of the sensor is finite. There is a need for implementing the combination of prediction and forecasting systems in the cloud system, which predicts future sensor data in advance so that it can reduce the communication of data as most of the user's requests receive replies at the cloud level.

3. Proposed Method

In the conventional models, all users request are transferred to the sensor network through the cloud system. Generally, users' requests are frequent. All the users' demands must be redirected to the sensor through the cloud system in the traditional approaches, which requires more data communication and consumes more energy. In our energy-efficient sensor cloud model, most of the user's requests are replied to at the cloud level using the combination of the artificial neural network-based prediction model having two activation functions and forecasting methods using ARIMA. The combination ANN based prediction model having two activation functions and forecasting methods using ARIMA [32] is used to predict the future sensor data in advance within the cloud. The proposed energy-efficient cloud-integrated sensor network model is shown in Fig. 1.

.

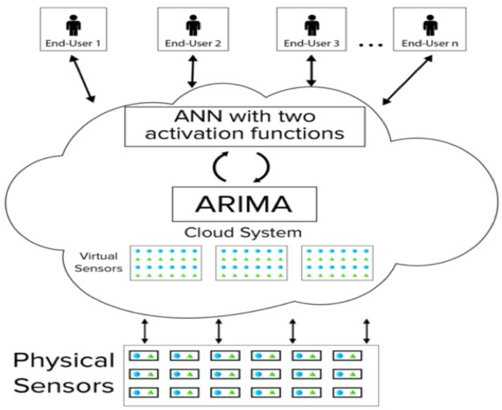

Fig. 1. Energy-efficient cloud-integrated sensor network.

We have used $\frac{e^x}{1-e^x}$ as the transfer function in the hidden layer and $\frac{e^x}{1+e^{-x}}$ as the activation function in the output layer for the simulation. Fig. 2 shows the combination of two activation functions based on the prediction method and forecasting using ARIMA.

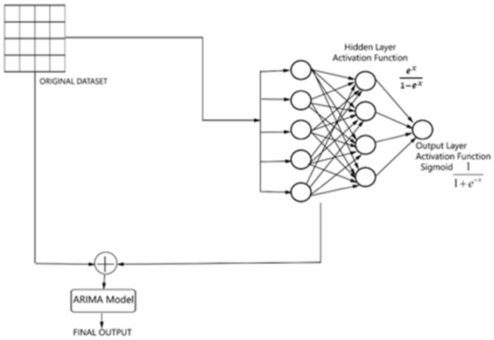

Fig. 2. The combination of two activation function based prediction and ARIMA.

3.1 ANN-based prediction

ANN models are useful for the modeling of various nonlinear type problems. In our ANN model, the output of the feed-forward network is as follows:

$$Output = \sigma\ (Dot\ Product\ (V, I) + C), \quad (3.1)$$

where I denotes the input for the neuron with bias C, V is the weight, and σ is known as the activation function. In our model, we are using $\frac{e^x}{1-e^x}$ as the activation function in the hidden layer with four nodes and $\frac{e^x}{1+e^{-x}}$ as the activation function in the output layer with one node.

The network's weight updates using the following equation:

$$\Delta\ output = \Sigma_j \frac{\partial output}{\partial V_j} \Delta V_j + \frac{\partial output}{\partial C_j} \Delta C_j, \quad (3.2)$$

The smoothness of σ means that a small change in the weight (ΔV), and a little change in the bias (ΔC) will produce a slight change in output. The cost function is as:

$$F\ (V, C) = \frac{1}{2n} \Sigma_x \| g(x) - \omega \|^2, \quad (3.3)$$

where V denotes the collection of all weights in the network, C denotes all the biases, n is the total number of training inputs, ω is the output from the network when x is the input and the sum overall training input x.

In the gradient descent technique, we are trying to minimize the value of the error function as:

$$\Delta F \approx \nabla F \cdot \Delta y,$$

where

$$\nabla F \equiv \left(\frac{\partial F}{\partial y_1}, \frac{\partial F}{\partial y_2}, \dots, \frac{\partial F}{\partial y_n} \right)^T, \quad (3.4)$$

such that $y_1, y_2, \dots y_n$ are the variables and

$$\Delta y = -\rho \nabla F, \quad (3.5)$$

with the learning rate ρ.
By applying the following functions repeatedly, we can easily find the minimum of the cost function:

$$V \rightarrow V' = V - \rho \frac{\partial F}{\partial V}, \quad (3.6)$$

$$C \rightarrow C' = C - \rho \frac{\partial F}{\partial C}. \quad (3.7)$$

The changes in weight and biases are as follows:

$$V \rightarrow V' = V - \frac{\rho}{m} \frac{\partial F}{\partial V}, \quad (3.8)$$

$$C \rightarrow C' = C - \frac{\rho}{m} \frac{\partial F}{\partial C}, \quad (3.9)$$

The symbol m is the size of the mini-batch. We calculate the error in a layer using the following function:

$$\delta_j^L = \frac{\partial F}{\partial a_j} \sigma(U^L j, \quad (3.10)$$

where a is the output of the layer, σ' is the derivative of the activation function we have used, $\frac{\partial F}{\partial U}$ is a measure of the error in the neuron and $U^L = V^L \cdot I^L - 1 + C^L$.

3.2 Forecasting using ARIMA model

ARIMA models are useful for the modeling of various linear type problems. The output of ANN with two activation functions is supplied to the ARIMA model. This ARIMA model consists of the 1st order of the autoregression, the 1st order of differencing, and 2nd order moving average. First of all, we applied non-linear log transformation to make the time series stationary. The p-value is more significant than the 5% level of significance, so it was concluded that the null hypothesis of data is not stationary cannot be rejected. That means the data is not stationary. Then, we apply the differencing method to make the time series stationary. Differencing is one of the most common ways of dealing with both trend and seasonality. In first-order differencing, you compute the differences between consecutive observations in the time series. This technique usually improves the stationarity of the time series. To apply an ARIMA model to our time series, we need to find optimal values for the following three model parameters (p, d, q): The number of autoregressive (AR) terms (p): AR terms are the lags of the dependent variable. So, if p=1, it means that predictors of x(t) will be x(t-1). The number of moving average (MA) terms (q): MA terms lagged forecast errors in the prediction equation. For instance, if q=2, the predictors for x(t) will be m(t-1) and m(t-2) where m(i) is the difference between the moving average at the ith instant and the actual value. The number of differences (d): These are the number of non-seasonal differences. In this case, $d=1$, as we are modeling using the first order differenced time series. Fig. 3 shows the Autocorrelation function's (ACF) values and partial autocorrelation function (PACF), respectively. The x-axis of the graph shows the randomly selected date from the dataset, and the y-axis shows the normalized temperature within a scale of 1.

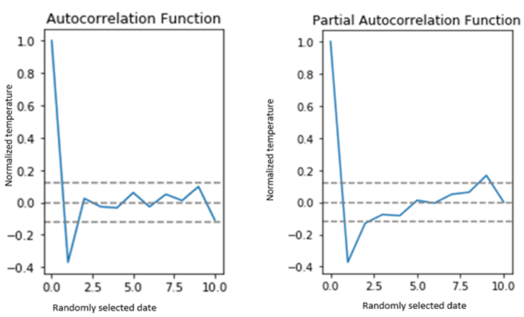

Fig. 3. The values of autocorrelation and partial autocorrelation function.

3.3 Combination of ANN and ARIMA

This section provides the steps for the ANN-based prediction model, having two activation functions and forecasting techniques using ARIMA.

Fig. 3 shows the flow chart for ANN-based prediction with two activation functions and ARIMA based forecasting.

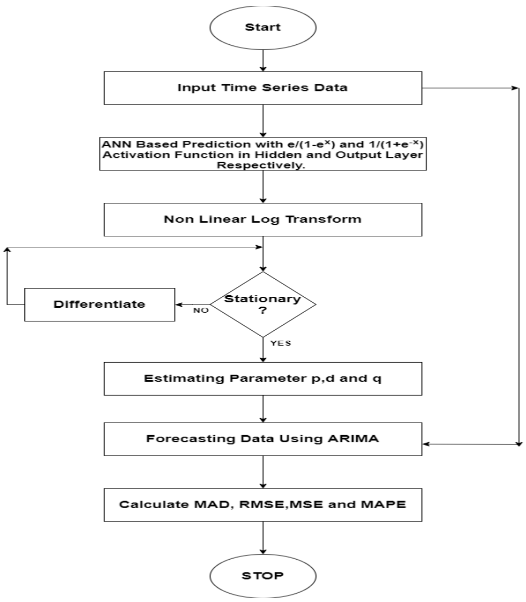

Fig. 4. The flow chart for ANN-based prediction with two activation functions and ARIMA based forecasting.

The time series Ts can represent [33-34]:

$$Ts = Ls + Ns,$$

where Ls and Ns are linear and nonlinear components present in the model, respectively. In the first step, ANN gives nonlinear forecasts, and the residuals we get from nonlinear components believe in having the direct link. Let Es denotes the residuals we got from ANN model so:

$$Es = Ts - Xs,$$

where Xs is the value, we have predicted for the ANN model. In the second step, the linear component is forecast by the ARIMA model, and the results we get are combined with the result of a nonlinear model to improve the performance of models. The new time series is represented as:

$$Ts' = Xs + Zs,$$

where *Zs* represents the forecasted value, we get from the residual data of the ARIMA model.

The combination of ANN and ARIMA provides better forecasting accuracy compared to ANN. This combined method predicts future sensor data in advance within the cloud system. As a result, there is less data communication between the sensor network and the cloud system. Less data communication saves energy within the sensor network.

4. Results

This section provides the results for the ANN-based prediction system, and this output result is fed into the ARIMA Model to get more accuracy. This section explains the energy consumption in the proposed and traditional methods. The parameters used for the calculation of the results are displayed in Table 1.

Table 1. Parameters used for result calculation.

Parameters	Values/Unit
Layer Used	Input, Hidden, Output
Activation Function in Hidden layer	$\frac{e^x}{1-e^x}$
Transmission Range	15 meter
lag order (p)	1
Degree of differencing (d)	1
Order of moving average (q)	2
No of Sensor Nodes	100
Area of Simulation	100 *100 square meter
Data Rate	1 Mbps
Actual Transferred Data	5Mb per Hour
Sink Position	Middle of the Square
Types of Communication	Multi-hop

4.1 Results of the integration of ANN and ARIMA

This section provides a brief description of a model that uses ANN using back-propagation [35] with two activation function-based predictions combined with the forecasting using ARIMA within the cloud system. The meteorological data collected from the "Kaggle" website from the year 2012 to 2017 of San Diego, California, USA, is used as the dataset for our simulation [36]. We choose data from the dataset randomly with a ratio of 0.83:0.17 for the model's training and testing. The dataset contains 45524 instances of measurements, which are divided into 37524 training and 8000 test samples. Our simulation uses temperature as the target parameter and four other parameters (i) Humidity, (ii) Pressure Pa, (iii) Wind Speed Km/h, and (iv) Wind Dir Degrees as the predictors. Here the temperature is in the units of Kelvin. With two activation functions, the ANN model predicts the temperature for the next hour with an accuracy of 94%. The prediction of this temperature for the next five hours can save five times the energy compared to the prediction of temperature data one hour in advance. Using only the ANN model for prediction for the next five hours may provide more power saving, but the accuracy may be less. Therefore, for more power-saving and more accurate forecasting, ANN and ARIMA methods merged. Applying the combination of two activation function-based ANN based prediction and ARIMA based forecasting model, the actual and predicted temperatures are displayed in Table 2.

Table 2. Predicted and actual temperatures.

Predicted Value	Actual Value	Predicted Value	Actual Value
21.089098	20.801429	21.889354	21.187083
20.899749	20.783929	19.112408	18.944597
23.596117	23.864940	17.289513	16.942838
23.311050	22.754702	17.935443	18.520119
22.299035	21.609444	17.949105	17.589479
20.161868	19.922661	20.735709	20.191898
19.73019	20.938869	20.192408	19.897738

It can be observed that the real and predicted values are nearly equal.

The mean absolute deviation (MAD), root mean square error (RMSE), mean squared error (MSE) and mean absolute percentage error (MAPE) for the actual and forecasted values using the ANN-ARIMA integration method and ANN method are displayed in Table 3.

Table 3. Comparison of performance for ANN-ARIMA integration method and ANN method.

Algorithm	Forecasted Temperature Data set (Hourly)	MAD	RMSE	MSE	MAPE
ANN-ARIMA	1-37524	0.4546	0.5323	0.2833	2.2355
ANN	1-37524	1.5568	2.0224	4.0902	8.6572

The performance measures, as reported in Table 3, reflect that this ANN with two activation functions integrated with ARIMA performs better than the ANN model since there is not much difference between the predicted values and the actual values. The observed values and the rolling one-step out-of-sample forecast are shown in Fig. 5.

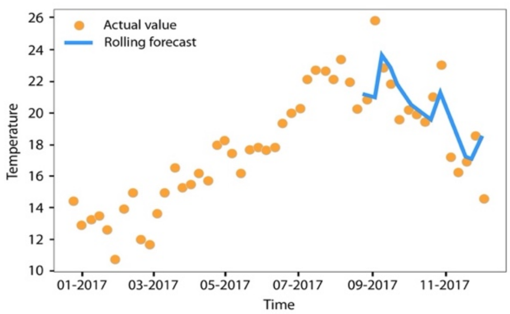

Fig. 5. The observed values and the rolling one-step out-of-sample forecast.

The x-axis of the graph shows the time from January 2017 to November 2017, and the y-axis of the figure shows the actual temperature within that period. Here the temperature per hour is plotted in dotted blue, and the green line shows the next 5 hours rolling forecast.

After every hour, the forecasted temperature gets added to the previous data, and the whole dataset, along with the previously predicted temperature, becomes an input to the ARIMA model. The ANN model with two activation functions is used first in this paper, predicting the temperature one hour in advance with an accuracy of 94%. After that, the ANN results and the previous real temperature are used as input to the ARIMA model. The prediction using the combination of ANN with two activation functions and ARIMA forecasting model provides the temperature of the next five hours with an accuracy of 98%.

4.2 Calculation of energy consumption in proposed and traditional approaches

Here the set of the sensor is $S = \{S1, S2, S3, \ldots, Sn\}$. Each sensor is represented as three tuples, i.e., <x-coordinate, y-coordinate, M> where y-coordinate and x-coordinate represent the sensor location, and M represents the measurement of the sensor. The sensor sends data to the gateway, and the end-user can access the data from the sensor through the cloud using this gateway. One sensor's data may be obtained by any end-user using the virtualization concept. Each sensor has a range of communication. If the gateway is within the scope of the sensor, the sensor can directly send data to this gateway using single hop communication, otherwise the sensor can send data using the multi-hop connection. During the multi-hop transmission, the bandwidth of outgoing traffic of a node is the sum of inflow traffic and the traffic generated by that node. The total power consumption of the wireless sensor network is the sum of power consumption of all the individual active sensors which take part in the communication process.

Our simulation environment is similar to the environment [1] consider by Guha et al. In our simulation, 100 sensor nodes are uniformly distributed in a square of side 100 meters. We assumed that each node transmits 1Mbps of traffic to the sink, which is in the middle of the square. The power consumption of all the 100 nodes was calculated by assuming that every node transmits data for five seconds per hour. In the traditional environment, all 100 nodes are active in communicating in 5 seconds per hour, so in 50 hours, all 100 nodes are active for 250 seconds. In our proposed approach, all the nodes are active for 5 seconds per five hours, so the total transmission time of all the 100 nodes is 50 seconds. Most of the user requests are replied to by the cloud system using ANN-based prediction with two activation functions and ARIMA based forecasting models. As fewer data transmissions take place, a smaller amount of data is transmitted. As a result, less energy is consumed in the proposed approach. Fig. 6 compares the energy consumption in the traditional methods and the proposed method.

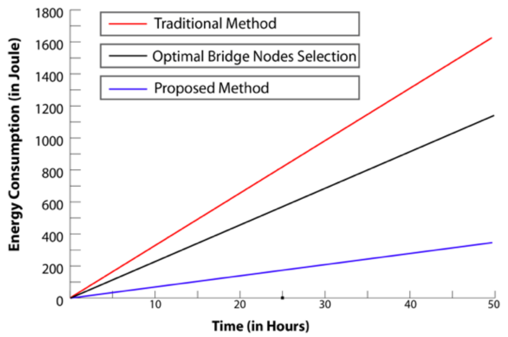

Fig. 6. Energy consumption in the Proposed and Traditional methods.

Our methods save energy compared to the optimal bridge nodes selection approach [37] as there is less transmission of data between sensor and cloud in the proposed method. Our proposed approach uses forecasting methods that forecast future sensor data within the cloud. The optimal bridge nodes selection method always sends real data from the sensor to the cloud. In the proposed approach, the RMSE is equal to 0.5323 between the actual and forecasted data within the acceptable range. The energy consumption will further minimize if we forecast temperature for more than 5 hours in advance. But the RMSE of actual and forecasted temperature will increase. So, in the proposed approach, we choose to forecast temperature for the next five hours so that the RMSE is within the acceptable range. Also, the energy consumption is less compared to traditional approaches.

The predictions are in the correct scale and are picking up the trend in the original time series. The combination of ANN with two activation functions and the ARIMA forecasting model is implemented within the cloud system so that most of the user's requests are replied to at the cloud level. As a result, there is less communication of data between the cloud and the sensor network, saving energy consumption. The RMSE of the forecasting was also less, which means the ANN with two activation function integrated ARIMA provided the forecasted temperatures, which are nearly equal to the actual values. Our simulation observed that the proposed method consumes less energy than the traditional approaches for the sensor cloud environment. Our proposed technique may not be saving much power when there is a sudden change of temperature that takes place frequently.

5. Conclusions

An energy-efficient sensor cloud model was proposed using the ANN-based prediction model with two activation functions and forecasting methods using ARIMA. Initially, two activation functions based on the ANN model were used to predict temperatures one hour in advance with 94% accuracy. After that, the result of ANN with previous real temperature was used as an input to the forecasting model based on ARIMA. After an hour, the forecasted temperature data was added to the last temperature data, and the whole dataset fed as an input to the ARIMA model. Our model can predict the temperature for the next five hours with 98% accuracy. Generally, all the user's requests are redirected to the sensor network, which results in more communication of data and shorter battery life of the sensor. In our model, the sensor communicates with the cloud every 5 hours. Most of the user requests are replied to by the cloud system using ANN-based prediction with two activation functions and ARIMA-based forecasting models. This approach results in less communication between the sensor network and the cloud system and increases battery life for the sensors. It was observed using the simulation that the proposed method consumes much less power as compared to the traditional approaches for the sensor cloud environment.

Acknowledgements

We are thankful to Mr. Aurobindo Mohapatra of Aptus Data Labs for providing useful suggestions.

References

[1] Guha RK, Gunter CA and Sarkar S. Fair Coalitions for Power-Aware Routing in Wireless Networks. IEEE Trans. Mobile Comput. 2007; 6(2):206-20.

[2] Mathew A, Sreekumar S, Khandelwal S, Kaul N, Kumar R. Prediction of Land-Surface Temperatures of Jaipur City Using Linear Time Series Model. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2016; 9: 3546-52.

[3] Papantoniou S, Kolokotsa D. Prediction of outdoor air temperature using neural networks: Application in 4 European Cities. Energy Build. 2016; 114: 72–9.

[4] Fang Y.C, Wu B.W. Prediction of the Thermal Imaging Minimum Resolvable (Circle) Temperature Difference with Neural Network Application. IEEE Trans. Pattern Anal. Mach. Intell. 2008; 30: 2218-28.

[5] Pal N.R, Pal S, Das J, Majumdar K. SOFM-MLP: A hybrid neural network for atmospheric temperature prediction. IEEE Trans. Geosci. Remote. Sens. 2003; 41: 2783–2791.

[6] Grbić R, Kurtagić D, Slišković D. Stream water temperature prediction based on Gaussian process regression. Expert Syst. Appl. 2013; 40: 7407–7414.

[7] Baosheng Y, Xiushui MA. Temperature prediction and analysis based on BP and Elman neural network for cement rotary kiln. In Proceedings of SPIE, 2010; 7997: 1-6.

[8] Xu B, Dan H, Li L. Temperature prediction model of asphalt pavement in cold regions based on an improved BP neural Network. Appl. Therm. Eng. 2017; 120: 568–80.

[9] Yu-Qing Z, Yan-Liang. Temperature Prediction of RCC Based on Partial Least-Squares Regression. ELSEVIER Energy Procedia. 2012; 17: 326–32.

[10] Singh S, Bhambri P, Gill J. Time series based temperature prediction using back propagation with genetic algorithm Technique. Int. J. Comput. Sci. 2011; 8: 28–32.

[11] Wang C, Qiu Z. Uncertain temperature field prediction of heat conduction problem with fuzzy parameters. Int. J. Heat Mass Transf. 2015; 91: 725–733.

[12] Abhishek K, Singh M.P, Ghosh S, Anand A. Weather Forecasting Model using Artificial Neural Network. ELSEVIER Procedia Technol. 2012; 4: 311–8.

[13] Das K, Das S, Darji R.K, Mishra A. Energy Efficient Model for the Sensor Cloud Systems. In Proceeding of IEEE RTEICT 2017: 373-5

[14] Das K, Das S, Mishra A, Mohapatra A. Energy Efficient Data Prediction Model for the Sensor Cloud Environment. In Proceeding of IEEE ICIOT 2017: 1-3.

[15] Hwang J, Du D.H.C, Kusmierek E. Energy efficient organization of mobile sensor networks. Int J Parallel Emergent Distrib Syst, 2015; 20(3-4): 221–33.

[16] Mamatas L, Tsaoussidis V. Exploiting energy-saving potential in heterogeneous networks. Int J Parallel Emergent Distrib Syst, 2008; 20(4): 309-24.

[17] Liua W, Zhoub J, Gongc B, Daic H, Guod M. Improving the energy efficiency of data-intensive applications running on Clusters, Int J Parallel Emergent Distrib Syst, 2018; DOI https://doi.org/10.1080/ 17445760. 2018.1455835

[18] Altomare A, Cesario E, Vinci A. Data analytics for energy-efficient clouds: Design, implementation and evaluation. Int J Parallel Emergent Distrib Syst, 2018; 34(6): 690-705.

[19] Tseng YC, Chang YN, Tzeng BH. Energy-Efficient Topology Control for Wireless Ad Hoc Sensor Networks. J Inf Sci Eng 2014; 20: 27-37.

[20] Chuang S.Y, Chen C. SmartBone: An Energy-Efficient Smart Backbone Construction in Wireless Sensor Networks. J Inf Sci Eng 2007; 23: 1023-39.

[21] Tapaonoi T, Phukpattaranont P, Suntiamorntut W, Chetpattananondh K. Analysis of power loss in sensor node (Unode). KKU Eng. J. 2012; 39(2): 147-53

[22] Danratchadakorn C, Pornavalai C. Coverage Maximization with Sleep Scheduling for Wireless Sensor Networks, ECTI-CIT 2016; 10: 55-63.

[23] Singsanga S, Usaha W. Fair Route Selection in Multi-Domain WSNs using Non-Cooperative Game Theory in Separate Sink Scenario ECTI Trans. EEC 2017; 15: 36-46.

[24] Sandar NM. Capacity Planning of Aggregators and Multi-objective Optimization Approach to Optimal Data Transmission in Cloud Providers for Meteorological Sensor Network. ISJET 2019; 3: 15-24.

[25] Xu H, Yang B. Energy-Aware Resource Management in Cloud Computing Considering Load Balance. J Inf Sci Eng 2017; 33: 1-16.

[26] Chaianong A, Bangviwat A & Menke C. Forecast of Rooftop PV Adoption in Thailand. Science & Technology Asia 2019; 24(4): 13-41.

[27] Krishna G, Singh, SK, Singh JP and Kumar P. Energy Conservation through Data Prediction in Wireless Sensor Networks. In Proceedings of 3rd International Conference on ICIoTCT 2018; 986-92.

[28] Jemal A, Hachicha M, Halima RB, Kacem AH, Drira K, Jmaiel M. Energy saving in WSN using monitoring values prediction. *Procedia Comput. Sci.* 2014; 32: 1154-9.

[29] Xia F, Xu Z, Yao L, Sun W and Li M. Prediction-based data transmission for energy conservation in wireless body sensors. In Proceedings of the 5th Annual ICST-WICON 2010; 1-9.

[30] Lin K and Chen M. Reliable routing based on energy prediction for wireless multimedia sensor networks. In Proceedings of the 53rd IEEE GLOBECOM 2010; 1–5.

[31] Samarah S. A data predication model for integrating wireless sensor networks and cloud computing. Procedia Comput. Sci. 2015; 52: 1141–6.

[32] Introduction to ARIMA models. [Internet]. [cited 2019 November 12]. Available From https://people.duke.edu/~rnau/411 arim.htm

[33] Ayub S, Jafri ZY Comparative Study of an ANN-ARIMA Hybrid Model for Predicting Karachi Stock Price. American Journ of Math and Stat. 2020; 10(1): 1-9.

[34] Zhang GP. Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing. 2003; 50: 159-75

[35] D. E. Rumelhart, G. E. Hinton, R.J. Williams. Learning representations by back-propagating errors. Nature, 1986;323: 533-6.

[36] Meteorological data of San Diego, California, USA. [Internet]. [cited 2019 November 12]. Available From https://www.kaggle.com/selfishgene/historical-hourly-weather-data

[37] Chatterjee S, Sarkar S, Misra S. Energy-Efficient Data Transmission in Sensor-Cloud. In Proceeding of IEEE AIMoC 2015: 1-6.