

Anomalies detection for smart-home energy forecasting using moving average

Jesmeen Mohd Zebaral Hoque¹, Gajula Ramana Murthy², Jakir Hossen¹, Jaya Ganesan³,
Azlan Abd Aziz¹, Chy. Mohammed Tawsif Khan¹

¹Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia

²Department of Electronics and Communication Engineering, Alliance College of Engineering and Design (ACED), Alliance University, Bangalore, India

³Alliance School of Business, Alliance University, Bangalore, India

Article Info

Article history:

Received Oct 6, 2020

Revised Aug 2, 2022

Accepted Aug 14, 2022

Keywords:

Anomaly

Data quality

Forecasting time series

Moving average

Smart home

ABSTRACT

In the past few years, the increase in the relation between the physical and digital world over the internet was witnessed. Even though the applications can enhance smart home systems, it is still early stages and challenges in the field of internet of things (IoT). An extreme level of data quality (DQ) system management is essential to produce a meaningful vision. However, in most home energy management system has no straightforward process of removing abnormal data. Hence, the research aims to propose and validate the model of anomaly detection for power consumption in real-time. The moving average (MA) approach identifies and removes abnormal energy consumption data. The results obtained from the forecasting time series autoregressive integrated moving average (ARIMA) model demonstrated that the proposed heuristics effectively enhanced energy usage forecasting. The selection of optimum parameter values for the MA was comprehended for time-series forecasting error minimization by comparing mean squared error (MSE). These outcomes proved the effectiveness of the existing technique and precision of choice of the appropriate. Therefore, the method can effectively route the cleaned sequence data streams in a real-time environment, which is valuable for spotting the anomalies and eliminating for enhancing energy usage time series.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Ramana Murthy Gajula

Department of Electronics and Communication Engineering, Alliance College of Engineering and Design (ACED), Alliance University

Bangalore, Karnataka, India-562106

Email: ramana.murthy@alliance.edu.in

1. INTRODUCTION

SMART Homes is one of the areas which first involves internet of things (IoT). A large number of data is collected from the home energy management system (HEMS), which also involves different challenges in different data analysis stages. Not only in the smart home field, accurate forecasting is required in different fields, such as Weather forecasting [1], patients number forecasting [2], marketing researches forecasting [3], mortality rates forecasting [4], rainfall forecasting [5], and more. The data can be collected, pre-processed, analyzed, and monitored using predictive analysis (PA), and advance intelligent technologies can help convert these data into reports, charts, and graphs. In 2019, it was reported by [6], that Malaysia Tenaga Nasional Berhad (TNB) raided many properties that were assumed of snooping on electricity supply in a bitcoin mining operation, which resulted in \$25 million loss for the utility company. These problems are

limited to Malaysia and conquered in different other countries, such as Iran, Argentina, Brazil, Venezuela, Turkey and few other European countries, and eastern powerhouse Russia.

As stated by [7], “data quality (DQ) is generally described as the capability of data to satisfy stated and implied needs when used under specified conditions”. To maintain DQ, it is essential to initiate the most common dimensions of DQ (i.e. Accurate date, complete data, consistent data) [8], [9], for further there are other dimensions such as consistent representation, accessibility, timeliness, and relevancy [10]. It was proved by different resources about the loss of billions of dollars due to poor DQ [11], [12]. Low-level DQ can cause due to wrong or missing data and is very essential to handle this type of dataset [13]. It may lead to incorrect or misleading decisions, predictions, or instructions. As stated by [14], dirty data can slow down any processing depending on data analysis (DA) and even affect the total cost for the organization; the cost can be over billions of dollars per year. It was also mentioned that around 60% of data in an organization contain data issues; hence, organizations are now worried about those dirty data.

Leonardi *et al.* [15] states that ‘PeerEnergyCloud’ covered the use cases related to the application of smart home technology in different fields. It indicates various applications are responding differently to different DQ phases besides to variable grades. It addressed the most common DQ aspects faced in smart home energy monitoring systems. They are data accuracy, data completeness, and data delay. However, they did not propose any real solutions to the issues. It is also essential to overcome the issues. Moreover, the study [16] has stated the importance of anomaly detection system requirement in a residential community for electrical load dataset. Detecting real-time abnormal behaviors are essential for smart home system users and TNB. Anomalies detection can help to point out outlier data and examine details and can avoid keeping attention on anomalous meters. Accurate load forecasting has become a significant part of planning and operation for all active participants in the electricity market; it will enable effective outcomes in HEMS and provide precise prediction and healthy real-time power control [17], [18]. With the new market structure in place, the penalty costs for under or over contracting electricity have significantly increased, minimizing costs and revenue losses more critical than ever [19]. To achieve the goal of an efficient system anomaly detection models by using technical analysis tool is proposed. These detection models will be implemented and adapt to the ever-growing usage of IoT devices involved in time series forecasting cases. The output is detected based on past energy usage, it is known as a trend-following or lagging indicator.

The first contribution is creating an application programming interface (API) for handling duplicate features and data deployed in a smart home environment. The main contribution is modelling the unusual behaviors of this smart home environment using moving average (MA) models; it can be utilized for detecting abnormal behaviors. The previous studies detected anomalies in energy usage in a household or a building apartment. However, they did not validate the system for time series forecasting energy usage. In response, a system was developed for time series data anomalies detection that will perform other pre-processing steps, eliminate anomalies without being liable on a specific monitoring system/tool, and modularize detecting anomalies method in a certain amount of evidently best definite modules. Here, an anomaly in energy consumption is detected and removed for the predictive model. Finally, the system was evaluated using the time series forecasting model auto regressive integrated moving average (ARIMA) to the specified scenario of HEMS; the model-assisted the reduction of forecasting error mean squared error (MSE) from 0.179 to 0.066.

2. LITERATURE REVIEW

Most researchers focused on anomaly detection to find the abnormal data which can threaten the management system. Vikhorev *et al.* [20] had developed an anomaly detection model for building management that caused extreme energy usage and probable growth in carbon taxes. Sial *et al.* [21] had presented four heuristics to identify abnormal energy consumption using a contextual grouping of smart meters. They grouped the meters containing the similar context of energy usage within the neighborhood of the meters by using distance from K-nearest neighbor days. However, using K-nearest neighbor model labelled datasets is required, which is challenging to obtain from different households for the study presented in this paper. Another system had proposed an anomaly detector for smart home security systems using hidden Markov model (HMM). They had achieved good accuracy while classifying potential anomalies that indicate attacks [22]. HMM can be implemented as both supervised and unsupervised learning. However, this author also implemented it for supervised learning.

Anomaly detection process can be beneficial for different aspects; for example, [23] had identified the anomalies in consumers’ behaviors over streaming data study, since anomaly can occur due to customer profile changing (known as concept drift) in the circumstances relatively theft, scam, or damages. Andrysiak *et al.* [24] had focused on detecting anomalies in the last-mile radio frequency (RF) smart grid communication network to find out the energy theft, and customers using energy deliberately or unconsciously, which cause disturbance in the system. A similar work [25] proposes an anomaly detection

system for thefts detection in communication in a simple process rather than using central point or electricity meters. Cui and Wang [26] had studied electricity consumption data of school and tested five models to detect anomalies. They had projected a hybrid model that connects polynomial regression and Gaussian distribution to identify anomalies in facilities management company datasets. They had concluded that the proposed prototype could also be implemented for different types of time series datasets. However, a good training dataset will be required to train the model manually before the actual detection process. Moreover, a MA is a statistical model which does not require any training. Most of the researchers indicated data anomaly caused due to energy theft in low-voltage networks. The old technique to detect these types of anomalies is by going through irregularities in the customer billing information. This centralized method requires time to detect as it requires scanning over manually historical long-term usage data. Zidi *et al.* [27] indicates the use of different sophisticated methods for theft detection. Another research work was held for daily real-time usage prediction using a hybrid neural network integration with ARIMA model for daily energy usage. They also detected anomalies by finding the differences between actual and predicted usage using the two-sigma rule [28]. Yu *et al.* [29] had applied k -nearest-neighbor with a sliding window and found that it is a valuable tool for identifying anomalies in hydrological time series cases and it improves DQ and helps to make better [29]. It also concludes that it is crucial to select the best combination of sliding window size and confidence coefficient for the specific use case and time series input. Hence, it is essential to select the best parameters to overcome the fitting issue.

Since time-series data may contain abnormal data, it is essential to remove these data for better forecasting. MA with a sliding window is the most straightforward technique used for anomaly detection. However, it is essential to get the tune parameters for the optimal detection system. As concluded by the researcher cited above, many had detected anomalies, but they had not evaluated after eliminating the anomalies improving the time-series forecasting. In this study, ARIMA was selected for testing the enhancement of time-series forecasting, as ARIMA is a powerful technique in which an own series history is used as an explanatory variable. Due to uni-variate modeling capability, ARIMA cannot exploit the leading indicators or descriptive variables.

3. METHOD: DATA ANALYTICS PROCESS WITH DATA QUALITY

Depending on HEMS, a DA model is designed and developed to analyze a smart home energy monitor system. The DA process was implemented in steps presented in Figure 1. Data issues are usually involved in different stages of DA. For instance, in the DA stages of collection, pre-processing and cleaning, and analysis, the issues that can handle are integrity, incomplete and duplicate data, and inaccurate data, respectively. Hence, it is required to maintain the primary cases for data issues in every stage of DA.

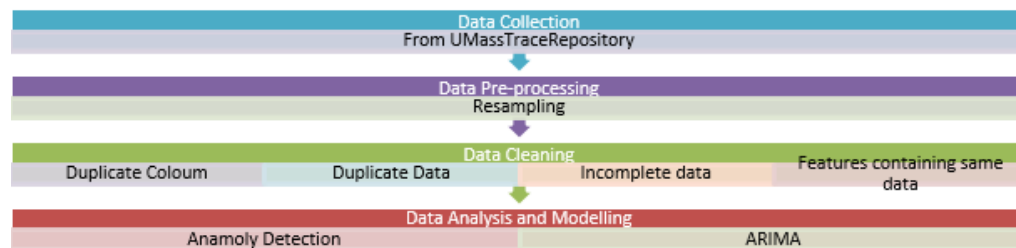


Figure 1. DA process overviews

3.1. Data collection

One of the significant roles to obtain result accuracy is the data collection stage. Collecting data from a reliable source is very important to make sure data is in real-world data. Typically, collected data contains data issues. It is essential to examine the data set to help to obtain accurate results clearly. The data is collected from UMass Trace Repository, which contains network, storage, and other traces. The Electrical dataset (aggregated and individual circuits) was used as input from 'HomeC' in this research. This data set will be helpful to validate the proposed approach for detecting and eliminating anomalies in time series.

3.2. Data pre-processing and cleaning

Data combined from heterogeneous sources can contain data inconsistencies and missing values. It is very important to clean the data before data analyzing [30]. Before data cleaning, a few steps of data

processing are held as follows presented in algorithm 1. Here, the first API, ‘*getDuplicateColumns*’ will be able to receive the dataset and find the columns name containing duplicate data by iteration process. The following API, ‘*dropDuplicateColumns*’ will receive the dataset too, and by using ‘*getDuplicateColumns*’ will find the duplicate columns and drop those columns.

Algorithm 1. Handle duplicate data (rows and columns)

```

This program handles duplicate data and drops unwanted data
param df: Dataframe object
Retrieve duplicate columns name and list
function getDuplicateColumns (Argument df){
    duplicateColumnNames = set()
    col= all the columns
    # Dataframe 'df' columns iteration
    for x in col:
        # Select column at xth index.
        # Iteration for col in df starting from (x+1) th position till end
        for y in range(col [x + 1], col):
            # Select column at yth index.
            secondCol = col [y]
            # match if two columns at xth and yth position are equal
            if col.equals(otherCol):
                Add secondCol in duplicateColumnNames
        forend
    forend
    return Columns name list which contains duplicate data.
end }
param dt: Dataframe object
function dropDuplicateColumns(Argument dt) {
    # Delete duplicate columns, call getDuplicateColumns function to get the columns
    dt = dt.drop(columns=getDuplicateColumns(dt))
    return dataset after cleaning duplicates, dt
end }

```

3.3. Data analysis and modelling

Predictive modelling will be executed by using a fully scripted Python Language. Here, in the time series in Figure 2, the output of the total cost data for a particular sure on a daily basis (Month on the horizontal axis and cost on the vertical axis) is presented. In this time series, the anomalies were detected for one month after 2nd January 2016 till 1st February 2016.

However, for forecasting 70/30 training to testing ratio will be selected, as it is one of the ideal ratios for training and testing dataset [31]. According to [32], 50% to 70% of the training set will more likely help get a good model, where a total of 70% of data will be used for the training purpose of the ARIMA method and the rest 30% for testing purpose. Present-day data is calculated with the help of past data using a moving average technique. MA is categorized into simple moving average (SMA) and exponential moving average (EMA). EMA outcome is contributed from the latest data, unlike SMA. In this experiment, MA is executed on a weekday basis for SMA and EMA. The records from the last weekdays are taken into the system for current value estimation. For example, to estimate the present Monday’s value, records from last Monday are used. Here, MA can identify and filter out the abnormal short-term fluctuations and smooth out the outcome. By calculating the values, it will be easy to detect and monitor the anomalies. These anomalies are one of the DQ issues [30], which occurred due to unusual behaviors such as incorrect data from the meter or energy theft. These will enable minimizing the outcomes of a variety of attacks from inside and outside structures. The processing of handling anomalies is presented in algorithm 2.

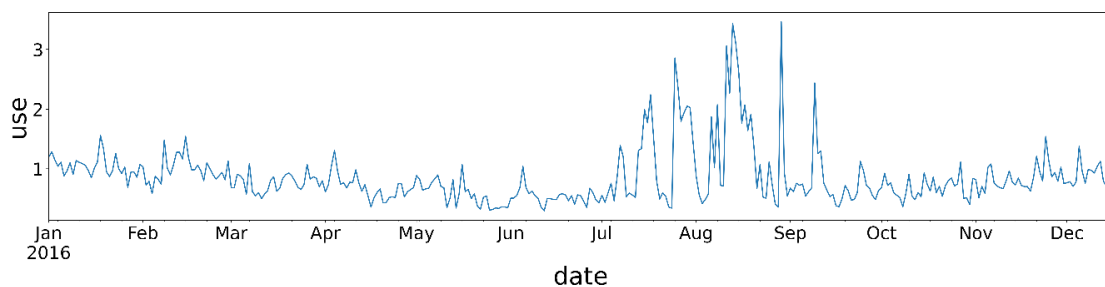


Figure 2. Time series plot of energy usage data

Algorithm 2. Handle data Anomaly

```

function meanAbsoluteError(Argument y_true, Argument y_pred){
    return np.mean(np.abs((y_true - y_pred) / y_true))
end }
Argument Input:
    df - dataframe with timeseries
    window - rolling window size
    plotIntervals - show confidence intervals (by default False)
    plotAnomalies - show anomalies by default False
function plotMovingAverage(Argument df, Argument window, Argument plotIntervals=False,
Argument plotscale=1.90, Argument plotAnomalies=False) {
    rolling_mean = df.rolling(window = window).mean()
    plot: timeseries with Moving average for window size 'window' with Rolling mean trend
    # Plot confidence intervals for smoothed values
    if plotIntervals:
        mae = meanAbsoluteError(df[window:], rollingMean[window:])
        sdeviation = np.std(df[window:] - rollingMean[window:])
        lowerBond = rollingMean - (mae + plotscale * sdeviation)
        upperBond = rollingMean + (mae + plotscale * sdeviation)
        plot: upperBond and lowerBond in timeseries
        # detect abnormal data and plot in timeseries
        if plotAnomalies
            anomalies = all data from df with timeseries
            anomalies[df<lowerBond] = df[df<lowerBond]
            anomalies[df>upperBond] = df[df>upperBond]
        Output anomalies without null value
        Remove anomalies from the list
        Plot: Mark anomalies in the figure.
    endif
    plot: Actual values of df with timeseries
end }

```

Here, mean absolute error (MAE) function can calculate the mean absolute error (output presented in section 4.2), it is used to calculate the upper bound and lower bound from the rolling mean.

Next, ARIMA model is trained in such a way that it will be able to predict based on historical observations. In this case of the energy monitoring system, the prediction will have the capacity to better forecast electricity use. This can result in decreased costs by reducing the usage of electricity. This is the domain of machine learning (ML), with a detailed group of approaches and procedures mainly suitable for value prediction of a dependent class to time perspective. This prediction model is a combination of autoregression (AR) and MA. AR is calculated using lagged values for y and MA is calculated using lagged errors, as presented in (1).

$$\hat{y}_t = \overbrace{\Phi_1 y_{t-1} + \dots + \Phi_n y_{t-n}}^{\text{AR terms (lagged values for } y\text{)}} + \overbrace{\theta_1 e_{t-1} + \dots + \theta_m e_{t-m}}^{\text{MA terms (lagged errors)}} + \epsilon_t \quad (1)$$

The \hat{y}_t contains predicted values using AR and MA terms. Where, $AR:n$ is order of the autoregression of the model and $MA:m$ = order of the MA aspect of the model. According to the calculation, the ARIMA function is specified by three order parameters: (n, i, m) . Here, i is Integration; it uses observation differences to produce the time series stationary, i.e. degree of difference.

4. RESULTS AND DISCUSSION

4.1. Detected anomaly data

From the outcome of anomaly detection, it was found that there are several data anomalies in one month. These anomalies were detected by using the mean rolling trend (green line) and calculate upper bond and lower bond of usage (red line), as indicated in graphs in Figure 3, for various window sizes such as 6 in Figure 3(a), 12 in Figure 3(b), 24 in Figure 3(c), 48 in Figure 3(d) and 96 in Figure 3(e) (using SMA) and Figure 4, for various values of alpha such as 0 in Figure 4(a), 0.05 in Figure 4(b), 0.10 in Figure 4(c), 0.20 in Figure 4(d), 0.40 in Figure 4(e) and 0.80 in Figure 4(f) (using EMA). The red dot indicates anomaly detected. Data plotted outside the red line are considered anomalies. Here the detection proceeds with 24 hours window. Here, the advantage of smaller window size had increased sensitivity to the usage changes in the primary process of generating the data as required, i.e. predicted data. However, a larger window size helped to reduce data noise due to the size of the data. Therefore, the task is to select the best window size, which will provide maximum predictive accuracy and minimum predictive error. It will maximize predictive accuracy by reducing error, which is the predictive value minus the predictive error as shown in Table 1.

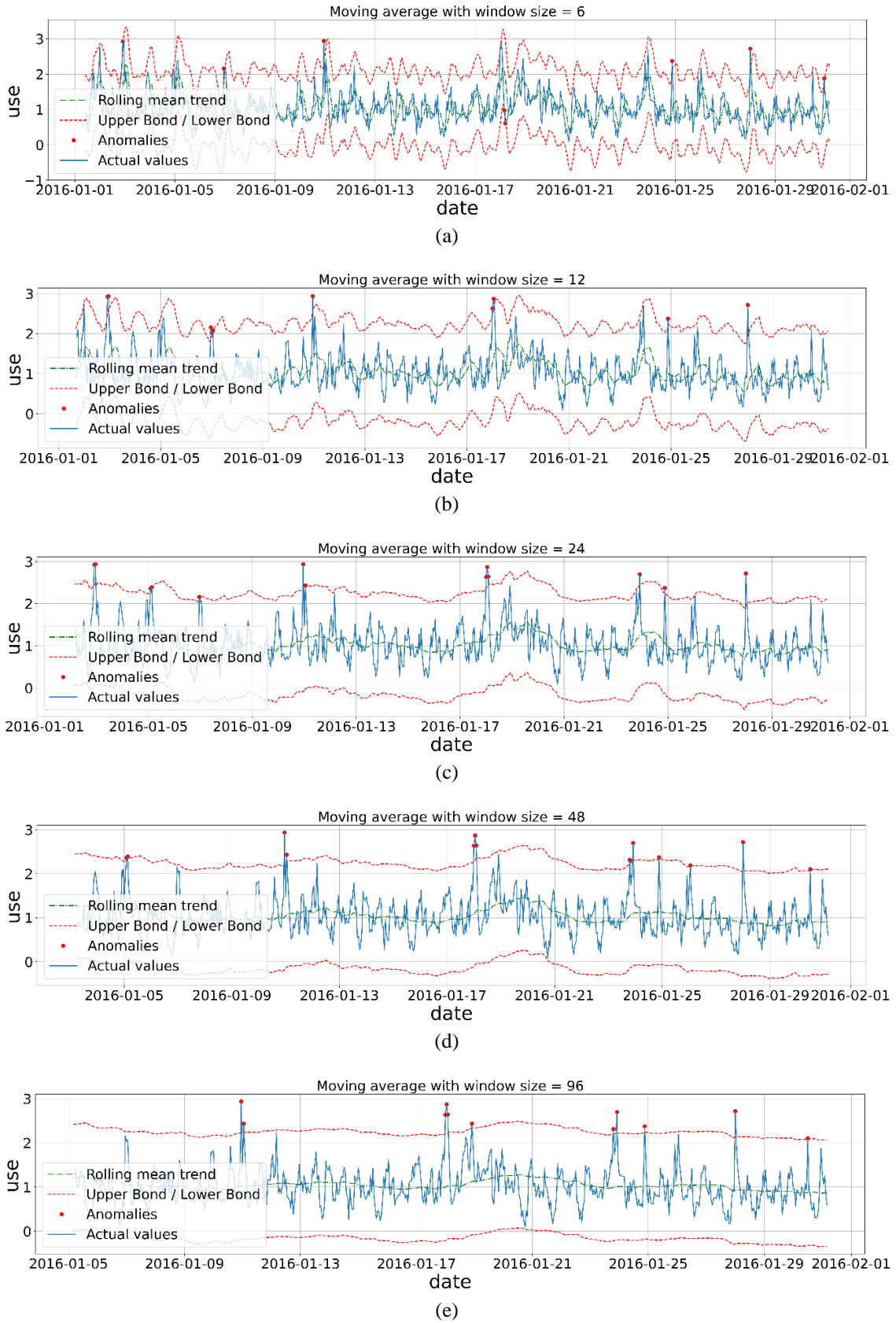


Figure 3. SMA for different windows with upper bond/lower bond line: (a) output for window size=6, (b) output for window size=12, (c) output for window size=24, (d) output for window size=48, and (e) output for window size=96

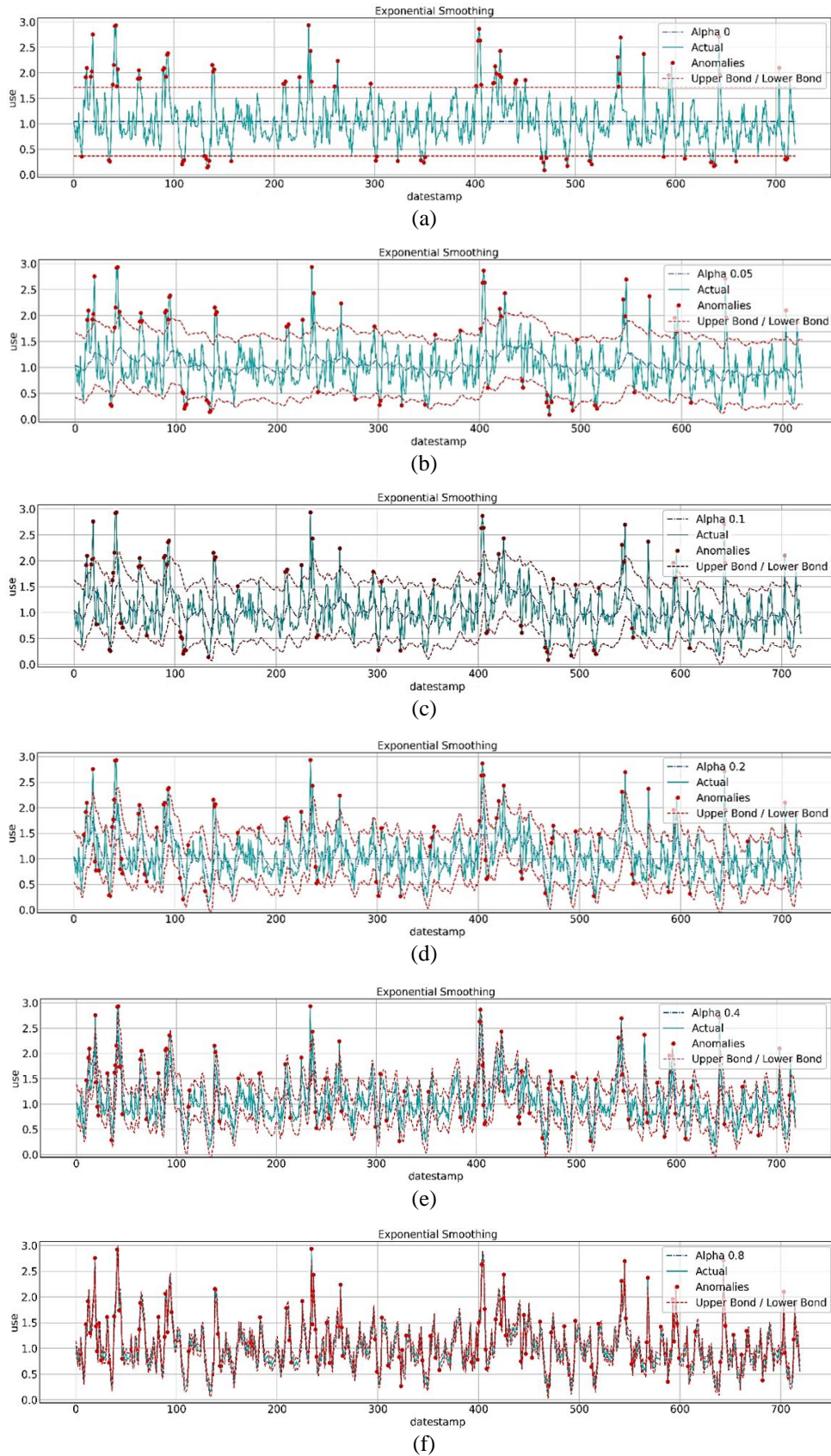


Figure 4. EMA for alpha value with upper bond/lower bond line; (a) output for alpha:0, (b) output for alpha: 0.05, (c) output for alpha: 0.10, (d) output for alpha: 0.20, (e) output for alpha: 0.40, and (f) output for alpha: 0.80

Table 1. Mean absolute error respected to hours of window

Mean absolute Error	
6	0.29746
12	0.34851
24	0.33639
48	0.33412
72	0.33770
96	0.3348

Here, all MA output is a substantial downside due to the lagging indicators. As MA is dependent on past facts of usage before there are changes in trend, it undergoes a time lag. Electricity usage may change quickly before a MA can exhibit a new trend transformation. In this case, a shorted MA faces issue from this lagging than in a longer MA. The output of SMA was found to be the most straightforward calculation, as the average usage obtains it over a chosen time period. With the use of SMA as a rolling mean, the identification of anomaly in the time was made easily using upper bond and lower bond line. However, EMA provided higher weighting to recent usage changes, its responses more rapidly to the values changes than the SMA. EMA was not proven effective for anomaly detection in forecasting time series (comparison in Tables 2 and 3).

Table 2. MSE with respect to Hours of Window for SMA

Window	Anomaly	Not Anomaly	MSE	Predicted	Expected
-	-	8,399	0.179	0.943062	0.935902
6	233	8,166	0.069	0.929682	0.920798
12	261	8,138	0.070	0.927745	0.935902
24	217	8,182	0.066	0.921924	0.935902
48	196	8,203	0.072	0.928044	0.935902
96	204	8,195	0.068	0.950717	0.935902

Table 3. MSE with respect to Alpha for EMA

Alpha	Anomaly	Not Anomaly	MSE	Predicted	Expected
-	-	8,399	0.179	0.943062	0.935902
0	367	8,032	0.014	0.875242	0.87774
0.05	710	7,689	0.124	0.904041	0.87774
0.1	719	7,680	0.09	0.883606	0.883105
0.2	710	7,689	0.049	0.878613	0.876929
0.4	725	7,674	0.034	0.874588	0.890962
0.8	729	7,670	0.043	0.861561	0.89183

4.2. Error measurement

Error measurement is a statistical calculation used to obtain the model error. Here, the mean absolute error, MAE is calculated using (2). It is used to analyze the amount of error in the model sure to different MA window sizes.

$$MAE = \frac{1}{n} \sum_{t=1}^n |\hat{y}_t - y_t| \quad (2)$$

The concept of MAE is used here to find the degree of closeness. It is obtained from the new estimated outcome (i.e. predicted values, \hat{y}_t) and the exact value (i.e. observed values, y). Here, n is the number of observations. The calculated output of different window sizes is presented in Table 1 and plotted in Figure 5.

The MAE shown in Figure 5 is the absolute difference between the estimated value and exact value. Here the rolling mean line is the estimated line. The window 12 MAE was considered to plot the upper and lower bound because after 12; the value seems stable. It is better to take a small window for faster processing.

4.3. System evaluation

The proposed system was evaluated by using the time series forecasting model ARIMA. The output obtained from ARIMA is evaluated by different parameters. By using *forecast()* function from 'statsmodels.tsa.arima_model.ARIMA'(ARIMA) class for making forecasting. Initially, the data set is splatted into a training and testing set. The training set is used to fit the model by using *fit()* function. It will

help to produce predicted outcomes for every element of the test set. Rolling forecasting is performed by re-creating the ARIMA model once each new observation is provided. This rolling forecasting is important for dependence on elements in earlier time steps for differencing and AR models. Using variable 'history', all elements were kept on track. This variable was seeded with the training dataset, and next, in each iteration, new elements were appended.

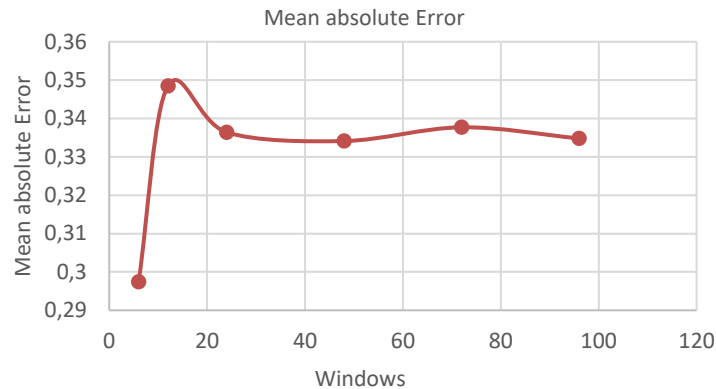


Figure 5. Mean absolute error

The time series forecasting graph is presented for ARIMA in Figure 6, for various window sizes such as original in Figure 6(a), 6 in Figure 6(b), 12 in Figure 6(c), 24 in Figure 6(d), 48 in Figure 6(e), and 96 in Figure 6(f) (see in appendix). Here, a blue straight line is plotted to present the expected outcome and then the dotted green line after the red horizontal line is plotted to compare with the rolling forecast predictions. It can be clearly understood, the forecasted elements illustrate trends close to the expected one, and they are on the correct scale too. The reason the proposed model outperformed the common statistical method, because the detected anomaly is based on current trend rather than fixing a threshold value.

4.4. Comparison using mean squared error

mean squared error (MSE) is the average of the squared errors used to calculate the forecasting error. Errors of opposite signs will not cancel each other out in either measure. MSE the values are all positive due to the squaring; this makes it easier to use in an optimization technique. The output for ARIMA for SMA and EMA is presented in Tables 2 and 3, respectively. Here, in the tables term "Window" indicates window size in an hour, term "Anomaly" and "Not Anomaly" is the number of usage anomalies detected and the number of usages not anomalies, term "MSE" output of error in the ARIMA model after removing the anomalies, term "Predicted" is predicted forecasting outcome removing the anomalies, and term "Expected" actual forecasting outcome in the time series forecasting after removing the anomalies. The first row of the table is the output of the training of the forecasting model without removing anomalies. Hence, the expected outcome should be close to 0.935902 to get an effective forecasting outcome.

5. CONCLUSION

This paper shows an effective proposal and algorithms, which will help detect anomalies in electricity use for the especially for smart home energy monitoring system. This will solve the issue of inaccurate data in data analysis stage. To achieve better anomaly detection outcome for time series smart HEMS, a practical model is presented a combination of MA and rolling mean. Besides, it also contains the implementation of cleaning data, such as the removal of duplicate or unwanted data. Handling duplicate and unwanted data will overcome the issue in the data pre-proposing stage. Next, ARIMA is implemented for forecasting time series data of smart home energy usage with detection and removal of anomalies. ARIMA is executed in the data analysis stage. Hence it was used to evaluate the proposed system. Before the evaluation process after detection and removal, the ARIMA model was tested and found that the forecasted elements illustrated closely to the expected outcome. Finally, it was proved that on detecting and removing anomalies by using SMA provides better forecasting than with anomalies. SMA helped to reduce the forecasting error MSE from 0.179 to 0.066. Moreover, EMA is not adequate for detecting anomalies compared to SMA as it reduces the expected outcome than the original one.

APPENDIX

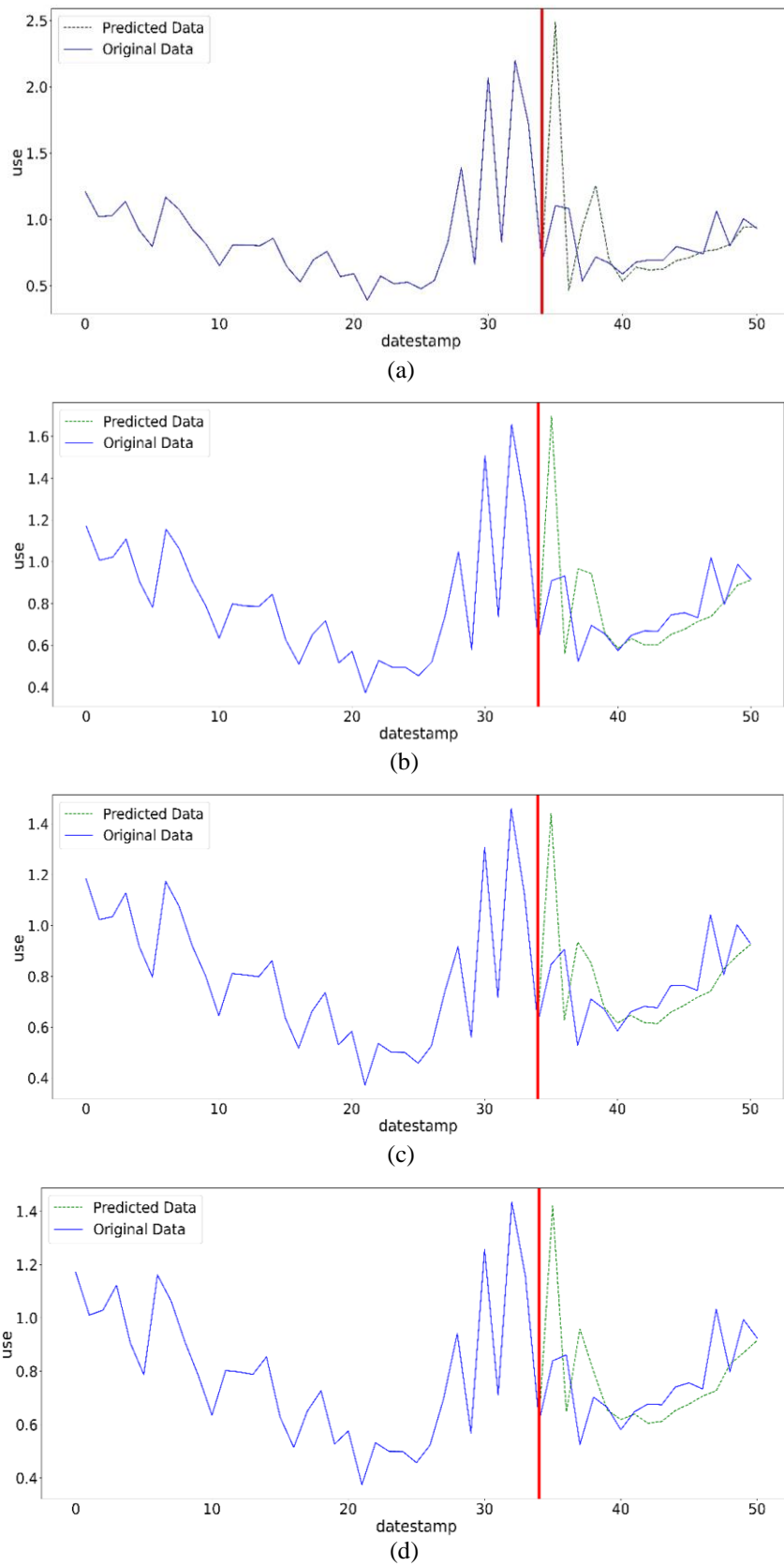


Figure 6. Time series forecasting using ARIMA: (a) original, (b) window:6, (c) window:12, (d) window:24, (e) window:48, and (f) window:96 (continue)

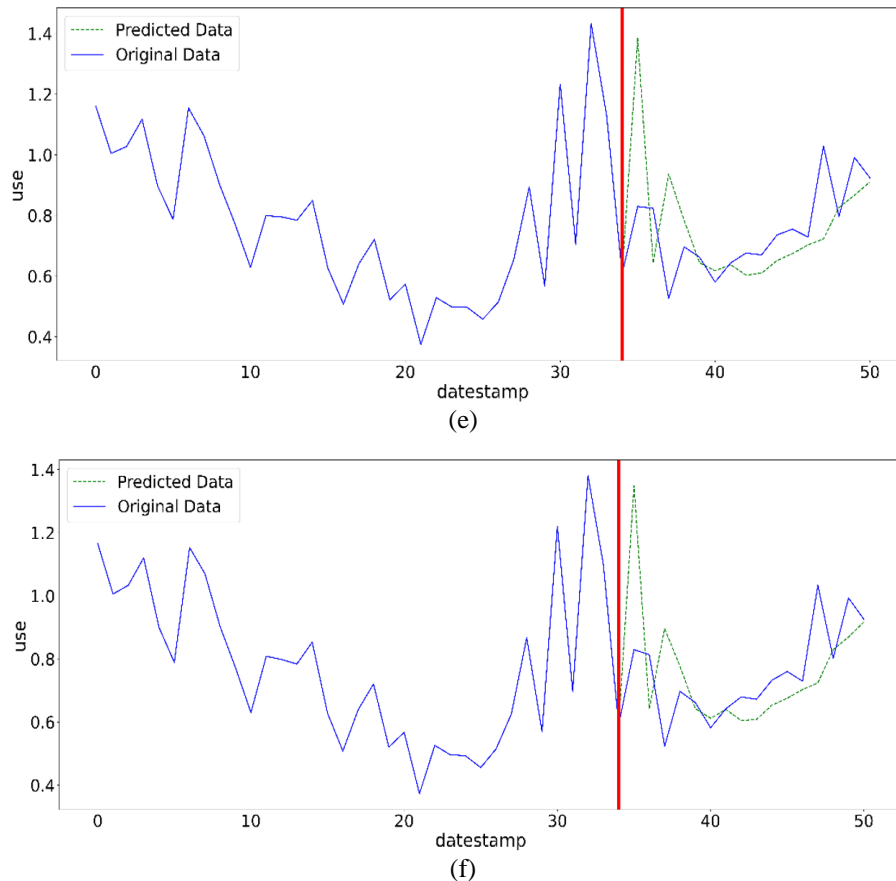


Figure 6. Time series forecasting using ARIMA: (a) original, (b) window:6, (c) window:12, (d) window:24, (e) window:48, and (f) window:96

ACKNOWLEDGEMENTS

This work was supported in part by Telekom Malaysia under Grant TMRND [MMUE/190007].




REFERENCES

- [1] A. G. Salman, Y. Heryadi, E. Abdurahman, and W. Suparta, "Weather forecasting using merged long short-term memory model (LSTM) and autoregressive integrated moving average (ARIMA) model," *Journal of Computer Science*, vol. 14, no. 7, pp. 930–938, Jul. 2018, doi: 10.3844/jcssp.2018.930.938.
- [2] D. M. Khairina, R. Khairunnisa, H. R. Hatta, and S. Maharani, "Comparison of the trend moment and double moving average methods for forecasting the number of dengue hemorrhagic fever patients," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 10, no. 2, pp. 978–987, Apr. 2021, doi: 10.11591/eei.v10i2.2711.
- [3] C. A. Rayed, "Using business intelligence solutions for forecasting in marketing researches," *International Journal of Informatics and Communication Technology (IJ-ICT)*, vol. 8, no. 2, pp. 102–110, Aug. 2019, doi: 10.11591/ijict.v8i2.pp102-110.
- [4] W. Z. Wan Husin and N. S. Z. Abidin, "Point forecasts of mortality rates in Malaysia: a comparison of principal component methods," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 19, no. 3, pp. 1661–1669, Sep. 2020, doi: 10.11591/ijeecs.v19.i3.pp1661-1669.
- [5] I. Cholissodin and S. Sutrisno, "Prediction of rainfall using improved deep learning with particle swarm optimization," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 5, pp. 2498–2504, Oct. 2020, doi: 10.12928/telkomnika.v18i5.14665.
- [6] T. Holman, "Electricity theft for bitcoin mining imposes loss of \$25 Million in Malaysia," *CryptoNewsZ*, 2019. <https://www.cryptonews.com/electricity-theft-for-bitcoin-mining-imposes-loss-of-25-million-in-malaysia/37197> (accessed Feb. 14, 2020).
- [7] F. Sidi, P. H. S. Panahy, L. S. Affendey, M. A. Jabar, H. Ibrahim, and A. Mustapha, "Data quality: A survey of data quality dimensions," in *2012 International Conference on Information Retrieval & Knowledge Management*, Mar. 2012, pp. 300–304, doi: 10.1109/infrkm.2012.6204995.
- [8] J. M. Z. H *et al.*, "A survey on cleaning dirty data using machine learning paradigm for big data analytics," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 10, no. 3, pp. 1234–1243, Jun. 2018, doi: 10.11591/ijeecs.v10.i3.pp1234-1243.
- [9] I. Taleb, H. T. El Kassabi, M. A. Serhani, R. Dssouli, and C. Bouhaddioui, "Big data quality: A quality dimensions evaluation," in *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing*

- and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCCom/IoP/SmartWorld), Jul. 2016, pp. 759–765, doi: 10.1109/UIC-ATC-ScalCom-CBDCCom-IoP-SmartWorld.2016.0122.
- [10] S. Juddoo, “Overview of data quality challenges in the context of Big Data,” in *2015 International Conference on Computing, Communication and Security (ICCCS)*, Dec. 2015, pp. 1–9, doi: 10.1109/ICCCS.2015.7374131.
- [11] G. E. Liepins and V. R. R. Uppuluri, *Data quality control: theory and pragmatics (statistics: A series of textbooks and monographs)*. CRC Press, 1990.
- [12] N. Laranjeiro, S. N. Soydemir, and J. Bernardino, “A survey on data quality: classifying poor data,” in *2015 IEEE 21st Pacific Rim International Symposium on Dependable Computing (PRDC)*, Nov. 2015, pp. 179–188, doi: 10.1109/PRDC.2015.41.
- [13] J. M. Z. H. et al., “AUTO-CDD: automatic cleaning dirty data using machine learning techniques,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 4, pp. 2076–2086, Aug. 2019, doi: 10.12928/telkomnika.v17i4.12780.
- [14] D. Dey and S. Kumar, “Reassessing data quality for information products,” *Management science*, vol. 56, no. 12, pp. 2316–2322, Dec. 2010, doi: 10.1287/mnsc.1100.1261.
- [15] A. Leonardi, H. Ziekow, M. Strohbach, and P. Kikiras, “Dealing with data quality in smart home environments—lessons learned from a smart grid pilot,” *Journal of Sensor and Actuator Networks*, vol. 5, no. 1, Mar. 2016, doi: 10.3390/jsan5010005.
- [16] X. Wang and S.-H. Ahn, “Real-time prediction and anomaly detection of electrical load in a residential community,” *Applied Energy*, vol. 259, Feb. 2020, doi: 10.1016/j.apenergy.2019.114145.
- [17] Y.-A. Daraghmi, E. Yaser Daraghmi, M. Daadoo, and S. Alsaadi, “Forecasting for smart energy: an accurate and efficient negative binomial additive model,” *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 20, no. 2, pp. 1000–1006, Nov. 2020, doi: 10.11591/ijeecs.v20.i2.pp1000-1006.
- [18] Z. M. Yasin, N. F. A. Aziz, N. A. Salim, N. A. Wahab, and N. A. Rahmat, “An accurate medium-term Load forecasting based on hybrid technique,” *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 12, no. 1, pp. 161–167, Oct. 2018, doi: 10.11591/ijeecs.v12.i1.pp161-167.
- [19] S.-C. Yip, W.-N. Tan, C. Tan, M.-T. Gan, and K. Wong, “An anomaly detection framework for identifying energy theft and defective meters in smart grids,” *International Journal of Electrical Power & Energy Systems*, vol. 101, pp. 189–203, Oct. 2018, doi: 10.1016/j.ijepes.2018.03.025.
- [20] K. Vikhorev, R. Greenough, and N. Brown, “An advanced energy management framework to promote energy awareness,” *Journal of Cleaner Production*, vol. 43, pp. 103–112, Mar. 2013, doi: 10.1016/j.jclepro.2012.12.012.
- [21] A. Sial, A. Singh, and A. Mahanti, “Detecting anomalous energy consumption using contextual analysis of smart meter data,” *Wireless Networks*, vol. 27, no. 6, pp. 4275–4292, Aug. 2021, doi: 10.1007/s11276-019-02074-8.
- [22] S. Ramapatruni, S. N. Narayanan, S. Mittal, A. Joshi, and K. Joshi, “Anomaly detection models for smart home security,” in *2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, May 2019, pp. 19–24, doi: 10.1109/BigDataSecurity-HPSC-IDS.2019.00015.
- [23] G. Fenza, M. Gallo, and V. Loia, “Drift-aware methodology for anomaly detection in smart grid,” *IEEE Access*, vol. 7, pp. 9645–9657, 2019, doi: 10.1109/ACCESS.2019.2891315.
- [24] T. Andrysiak, Ł. Saganowski, and P. Kiedrowski, “Anomaly detection in smart metering infrastructure with the use of time series analysis,” *Journal of Sensors*, vol. 2017, pp. 1–15, 2017, doi: 10.1155/2017/8782131.
- [25] E. Yu Shchetinin, “Cluster-based energy consumption forecasting in smart grids,” *Journal of Physics: Conference Series*, vol. 1205, Apr. 2019, doi: 10.1088/1742-6596/1205/1/012051.
- [26] W. Cui and H. Wang, “A new anomaly detection system for school electricity consumption data,” *Information*, vol. 8, no. 4, pp. 151, Nov. 2017, doi: 10.3390/info8040151.
- [27] S. Zidi, A. Mihoub, S. Mian Qaisar, M. Krichen, and Q. Abu Al-Haija, “Theft detection dataset for benchmarking and machine learning based classification in a smart grid environment,” *Journal of King Saud University - Computer and Information Sciences*, pp. 66–81, May 2022, doi: 10.1016/j.jksuci.2022.05.007.
- [28] J.-S. Chou and A. S. Telaga, “Real-time detection of anomalous power consumption,” *Renewable and Sustainable Energy Reviews*, vol. 33, pp. 400–411, May 2014, doi: 10.1016/j.rser.2014.01.088.
- [29] Y. Yu, Y. Zhu, S. Li, and D. Wan, “Time series outlier detection based on sliding window prediction,” *Mathematical Problems in Engineering*, vol. 2014, pp. 1–14, 2014, doi: 10.1155/2014/879736.
- [30] A. F. Jabbar and I. J. Mohammed, “BotDetectorFW: an optimized botnet detection framework based on five features-distance measures supported by comparisons of four machine learning classifiers using CICIDS2017 dataset,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 377–390, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp377-390.
- [31] H. Liu and M. Cocea, “Semi-random partitioning of data into training and test sets in granular computing context,” *Granular Computing*, vol. 2, no. 4, pp. 357–386, Dec. 2017, doi: 10.1007/s41066-017-0049-2.
- [32] Y. Xu and R. Goodacre, “On splitting training and validation set: A comparative study of cross-validation, bootstrap and systematic sampling for estimating the generalization performance of supervised learning,” *Journal of Analysis and Testing*, vol. 2, no. 3, pp. 249–262, Jul. 2018, doi: 10.1007/s41664-018-0068-2.

BIOGRAPHIES OF AUTHORS






Jesmeen Mohd Zebaral Hoque     currently a PhD student in Engineering and specialization in artificial intelligence from Multimedia University, Malaysia. She completed master’s in engineering (Multimedia University, Malaysia), and bachelor’s degree in computer science and engineering (International Islamic University Chittagong, Bangladesh). Her research interest is artificial intelligent, big data, smart home and machine learning. She can be contacted at email: jesmeen.online@gmail.com.






Gajula Ramana Murthy    Eng, MIET, SMIEEE received B. Tech degree from Acharya Nagarjuna University, Andhra Pradesh, India in 1990, M. Tech degree from G.B. Pant University of Agriculture & Technology, Uttar Pradesh, India in 1993, and PhD from Multimedia University, Malaysia and secured the grant from Telekom Malaysia in 2019. Currently he is working as Professor in E.C.E Department from Alliance College of Engineering and Design at Alliance University, Bangalore, India. His main research interests include VLSI, embedded systems, nanotechnology, memory optimization, low-power design, FPGA, evolutionary algorithms. This grant was secured by him in 2019 May from TMR&D, Malaysia. He can be contacted at email: gajularm@gmail.com.






Jakir Hossen    is graduated in Mechanical Engineering from the Dhaka University of Engineering and Technology (1997), Master's in communication and Network Engineering from Universiti Putra Malaysia (2003) and PhD in Smart Technology and Robotic Engineering from Universiti Putra Malaysia (2012). He is currently a Senior Lecturer at the Faculty of Engineering and Technology, Multimedia University, Malaysia. His research interests are in the area of artificial intelligence (fuzzy logic, neural network), inference systems, pattern classification, mobile robot navigation and intelligent control. He can be contacted at email: jakir.hossen@mmu.edu.my.






Jaya Ganesan    is currently working as Professor in Alliance School of Business at Alliance University, Bangalore, India. Dr. Jaya Ganesan is the current project leader for this grant funded by TM. As an active researcher she is a member of the Faculty's Research and Innovation Committee and served as Coordinator for Post Graduate Programs by Research for five years. Dr. Jaya Ganesan has 25 years of teaching, research and administrative experience in the higher education sector. Her areas of expertise include Human Resource Analytics, Green Human Resource Management, OB, IHRM. She is a reviewer and editorial member for internationally recognized journals. He can be contacted at email: jaya.ganesan@mmu.edu.my.



Azlan Abd Aziz    was conferred the B.S. degree in electrical and computer engineering by the Ohio State University, USA in 1998. Then, he obtained the M.S. degree in communication engineering from the University of Manchester, UK in 2004 and a Doctoral degree in engineering and computer science from Nagoya Institute of Technology, Japan in 2012 under the Chevening and Mombukakusho scholarships. He has more than a decade industrial experience in telecommunication sectors and is a certified professional engineer in communication engineering. He is currently attached to the Faculty of Engineering and Technology, Multimedia University, Malaysia. His research interests include coding theory in communication systems, signal processing and deep learning for vehicular communication applications. His current research is primarily in physical layer security for wireless networks, vehicular communications, and smart antenna applications. He can be contacted at email: azlan.abdaziz@mmu.edu.my.



Chy. Mohammed Tawsif Khan    is currently a PhD student in Engineering Program and is researching Artificial Intelligence, Event Processing, and big data from Multimedia University (MMU). He pursued a Master's degree in Engineering from MMU; and Bachelor's degree in Computer Science and Engineering from International Islamic University Chittagong, Bangladesh. He can be contacted at email: tawsif.online@gmail.com.