J Inf Process Syst, Vol.18, No.1, pp.115~129, February 2022
https://doi.org/10.3745/JIPS.04.0235

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

A Novel Framework Based on CNN-LSTM Neural Network for Prediction of Missing Values in Electricity Consumption Time-Series Datasets

Syed Nazir Hussain[1,*], Azlan Abd Aziz[1], Md. Jakir Hossen[1], Nor Azlina Ab Aziz[1],
G. Ramana Murthy[2], and Fajaruddin Bin Mustakim[3]

Abstract

Adopting Internet of Things (IoT)-based technologies in smart homes helps users analyze home appliances electricity consumption for better overall cost monitoring. The IoT application like smart home system (SHS) could suffer from large missing values gaps due to several factors such as security attacks, sensor faults, or connection errors. In this paper, a novel framework has been proposed to predict large gaps of missing values from the SHS home appliances electricity consumption time-series datasets. The framework follows a series of steps to detect, predict and reconstruct the input time-series datasets of missing values. A hybrid convolutional neural network-long short term memory (CNN-LSTM) neural network used to forecast large missing values gaps. A comparative experiment has been conducted to evaluate the performance of hybrid CNN-LSTM with its single variant CNN and LSTM in forecasting missing values. The experimental results indicate a performance superiority of the CNN-LSTM model over the single CNN and LSTM neural networks.

Keywords

CNN-LSTM Neural Network, Electricity Consumption Prediction, Large Gaps of Missing Values, Prediction of Missing Values in Time-Series Data, Smart Home System

1. Introduction

The Internet of Things (IoT) technology has been widely deployed in various sectors due to wireless communication and interactivity to environmental factors [1]. The smart home system (SHS) is among the most significant IoT invention through which users can manage and control physical objects in their surroundings [2].

The emergence of smart sensors and smart grid technologies has given rise to the concept of developing SHS that assist users in taking control of electricity consumption. In SHS, the electricity consumption data is obtained from IoT sensors and stored on remote servers used for data representation by several applications [3]. While examining IoT data, the critical issue that needs to address carefully is the data quality [4]. Data quality is crucial while developing IoT based SHS. If the data are of low quality;

Manuscript received December 17, 2020; first revision February 15, 2021; accepted March 5, 2021.

* **Corresponding Author:** Syed Nazir Hussain (syednazir13@gmail.com)

[1] Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia (syednazir13@gmail.com, azlan.abdaziz@mmu.edu.my, jakir.hossen@mmu.edu.my, azlina.aziz@mmu.edu.my)

[2] Dept. of Electronic and Computer Engineering, Vignan's Foundation, Pradesh, India (gajularm@gmail.com)

[3] Universiti Tun Hussein Onn Malaysia (UTHM), Johor, Malaysia (fajardin5695@gmail.com)

Current affiliation for author, G. Ramana Murthy, is Dept. of Electronic and Computer Engineering, Alliance College of Engineering and Design (ACED), Alliance University, Bangalore, India

judgments are likely to be incorrect. Generally, IoT data has two types of values, which may significantly minimize the consistency of data collected from appliances, such as the abnormal values and the missing values. It usually occurs due to an incorrect response or no response.

Imputation of missing values by estimated values, several methods have been proposed in previous research. These methods mainly classified into two categories, such as statistical and machine learning. Statistical approaches for predicting missing values based on mean, mode, and least square regression (LSR), commonly used over a long time. In [5], the authors briefly described the cases of missing data and several statistical-based approaches used to handle missing data. Among these approaches, multiple imputation and maximum likelihood are the two modern imputation approaches for handling missing data [6]. Machine learning methods are also gain great concern in the study of the imputation of missing data. The robust supervised machine learning classification and regression methods such as a k-nearest neighbor (KNN), support vector regression (SVR), and random forecast have also been used in missing data imputation [7–10]. These popular machine learning imputation methods support only categorical or multivariate data in which several variables are available to estimate a particular missing variable [11].

In previous research, various imputation approaches proposed to estimate missing values in univariate time-series datasets. Sridevi et al. [12] developed the autoregressive (ARLSimpute) method to identify missing values and prediction techniques, namely linear prediction and quadratic prediction. Mohamed et al. [13] proposed a multiple imputation type framework that operates on the principle of iterative and successive forward and backward predictions to estimate and ensemble the missing value. Wu et al. [14] proposed a novel method to predict missing values in time series data without imputation. The method uses the least square support vector machine (LSSVM) in which the input patterns with the temporal information defined as local time index (LTI) feds into the model. Dhevi [15] performed the imputation of missing values using the inverse distance weighted interpolation (IDWI) method for the Istanbul Stock Exchange dataset. This approach assigns values to unknown points from a weighted total of identified point values and works well with results calculated at uneven time intervals.

Caillault et al. [16] proposed a framework for estimating large gaps of missing values in univariate time series data based on dynamic time wrapping (DTW). The DTW-based imputation algorithm finds and replaces the most similar subsequence with missing values. The framework follows certain constraints, such as the assumptions of recurring time-series data (high cross-correlation indicator), which needs high computation time for considerable missing intervals. Bokde et al. [17] presented a new method that modifies the pattern sequence forecasting (PSF) algorithm, which concurrently forecasts and backcasts missing values for imputation. The PSF impute fits perfectly for datasets of periodic elements such as univariate time series, even when observations in continuous blocks are missing. Ding et al. [18] experiment to perform imputation on the IoT time-series missing values dataset. In this experiment, three algorithms, such as radial base function (RBF), moving less square (MLS), and adaptive inverse distance weighted (AIDW), are used and compare that imputed accuracy and efficiency with KNN. Among these three, the estimation results of MLS are the finest. Chaudhry et al. [19] overcome the challenges of imputing missing data in univariate and multivariate time-series datasets by proposing a new method that transforms a single variable into a multivariate type. The method exploits the high seasonality and random missingness of the average cell variable throughput in the long term evaluation LTE spectrum dataset.

Kim et al. [20] carried out an imputation on weather data that is a very significant environmental factor for the accurate forecasting of photovoltaic (PV) system. In this process, four imputation methods of statistical and machine learning are compared, such as linear interpolation (LI), mode imputation (MI), KNN, and multivariate imputation by chained equations (MICE). However, the comparison results show that the KNN method yields the lowest error for large missing data ranges. Al-Milli and Almobaideen [21] proposed a hybrid algorithm based on Jordan neural network and genetic algorithm to predict medical IoT applications categorical missing data. The algorithm split the data into complete and incomplete sets while training a neural network to predict missing values, and the genetic algorithm used to optimize the neural network weight. The existing approaches to deal with large gaps missing values in univariate time series data are not fully cover the entire series [16]. The practical and optimal solution needed for real-time systems ensures continuity of time-series datasets without any manual assistance.

This paper proposes a novel framework to automatically detect and predict large gaps of missing values from real-time SHS named beLyfe. The beLyfe system represents several home appliances electricity consumption time-series dataset to its users, suffering from large missing values gaps. The framework uses a hybrid convolutional neural network-long short term memory (CNN-LSTM) model to predict missing values of large gaps by taking time-series data of individual appliances as input. Fig. 1 represents an air-condition home appliance time-series dataset in which we can observe several large gaps of missing values.

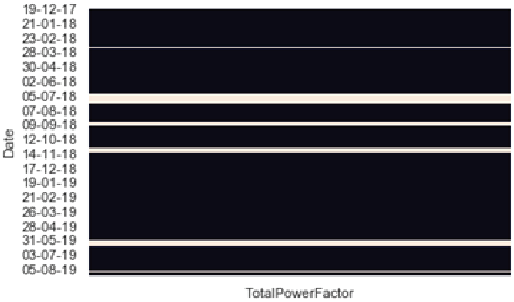


Fig. 1. Time-series data of air-condition appliance measured in kilowatt (kW).

When we analyze several appliances time-series datasets, we determine the missing values are within the range of 7, 15, and 30 time-steps. The proposed framework architecture initially design to predict the missing values within these detected ranges, such as 7, 15, and 30 time-steps. Our proposed framework follows a series of steps for automatic detection, prediction, and reconstruction of missing values from the input time-series datasets.

The proposed framework adopts a forecasting approach to estimate the large gaps of missing values as displayed in Fig. 1 instead of using conventional imputation techniques [22]. Commonly, the electricity load time-series dataset contains patterns that are complex and nonlinear due to several influencing factors such as meteorological influences, distinct occupancy behaviour regarding electricity consumption [23]. Researchers have recently used the CNN-LSTM neural network [24,25] to forecast electricity load data and achieve high predictive efficiency through the use of multivariate time series datasets. It expected that multivariate time series models could produce more precise forecasts than univariate models [26]. In our case, we have univariate time-series dataset in which only one predictor/ dependent variable is available corresponding to the time intervals, which is a specific day of a month.

Recently, recurrent neural network models like LSTM and hybrid CNN-LSTM gain great concern in forecasting electricity load time-series datasets. These models are mainly used to predict sequence data like time-series. For this reason, we conduct a comparison experiment between hybrid CNN-LSTM with its single variants such as CNN and LSTM neural networks for the prediction of missing values large gaps that are with the ranges of 7, 15, and 30 time-steps. A suitable hyperparameters determined by performing hyperparameter tuning on each model used in a comparison experiment. Experiment results indicate that the hybrid CNN-LSTM neural network achieves overall high predictive accuracy and efficiency.

The rest of the paper organized as follows. Section 2 briefly describes the hybrid CNN-LSTM model architecture. Section 3 describes our proposed framework. Section 4 describes the experimental setup, and Section 5 describes the performance analysis. Finally, the conclusion is drawn.

2. The Hybrid CNN-LSTM Model

The CNN-LSTM model explicitly developed to accommodate very long input sequences. The CNN model can interpret input sequences as blocks or subsequences, and the LSTM will combine the interpretations of these sub-sequences. The hybrid CNN-LSTM neural network architecture contains CNN layers that extract significant features from one-dimensional input data combined with LSTM that interpret them to support sequence prediction. Fig. 2 represents the hybrid CNN-LSTM model architecture to predict one-dimensional univariate time-series data [27].

The hybrid CNN-LSTM model can effectively predict nonlinear univariate time-series data that involves high volatility and uncertainty in the data patterns.

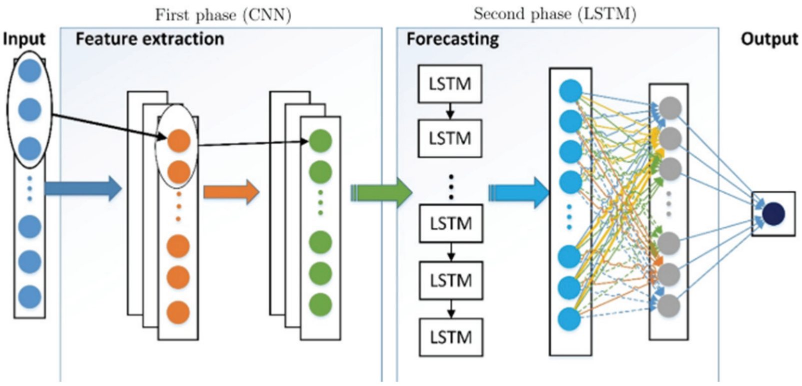


Fig. 2. The hybrid CNN-LSTM neural network internal architecture [27].

3. Proposed Framework

This section discusses our proposed framework, designed to predict the large gaps of missing values in the beLyfe appliances electricity consumption time-series dataset. Our proposed framework architecture shown in Fig. 3 developed using PyThon version 3.3.7 and TensorFlow version 1.12.0. The input data has been collected from the beLyfe SHS server database by the given application programming interface (APIs). The framework follows a series of steps required to transform the data and predict the following missing values in the dataset. Each of these steps briefly described below.

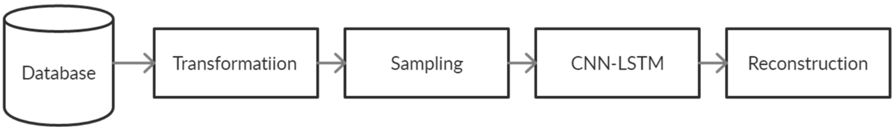
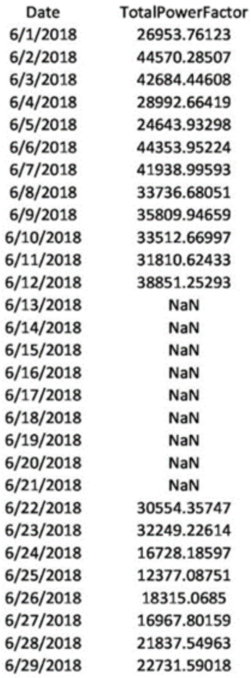


Fig. 3. Overview of a proposed framework architecture.

3.1 Transformation

In the transformation phase, the source data collected and transformed into a standard format. Since in the raw input data, the consecutive missing dates are not present. First, we need to identify consecutive missing dates from electricity consumption time-series data and fill them with not-a-number (NaN) values, as shown in Fig. 4.



Date	TotalPowerFactor
6/1/2018	26953.76123
6/2/2018	44570.28507
6/3/2018	42684.44608
6/4/2018	28992.66419
6/5/2018	24643.93298
6/6/2018	44353.95224
6/7/2018	41938.99593
6/8/2018	33736.68051
6/9/2018	35809.94659
6/10/2018	33512.66997
6/11/2018	31810.62433
6/12/2018	38851.25293
6/13/2018	NaN
6/14/2018	NaN
6/15/2018	NaN
6/16/2018	NaN
6/17/2018	NaN
6/18/2018	NaN
6/19/2018	NaN
6/20/2018	NaN
6/21/2018	NaN
6/22/2018	30554.35747
6/23/2018	32249.22614
6/24/2018	16728.18597
6/25/2018	12377.08751
6/26/2018	18315.0685
6/27/2018	16967.80159
6/28/2018	21837.54963
6/29/2018	22731.59018

Fig. 4. Transformation of the give input appliance time-series dataset.

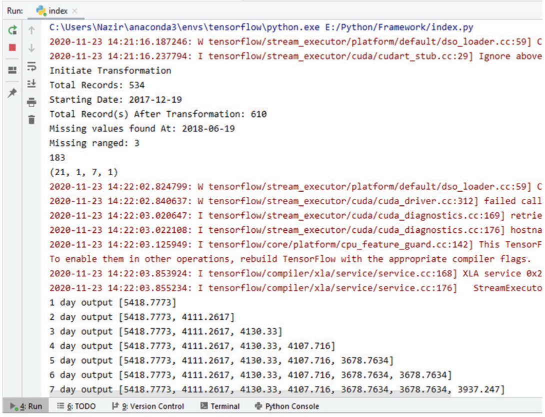
3.2 Sampling

In the sampling phase, the time-series data re-framed into the input and output patterns called samples. This process of sampling depends on the range of consecutive missing values detected in time-series data. In our case, the consecutive missing values are within the range of 7, 15, and 30 observations. The CNN-LSTM model one-dimensional input vector needs to be reshaped into samples, subsequences, time-steps, and features, as shown in Eq. (1), to predict missing values within these define ranges. However, based on the number of missing values detection ranges, the sampling part collects and reshape the input for the CNN-LSTM model.

$$Input\ Vector = [samples, subsequences, timesteps, features] \tag{1}$$

3.3 Predictive CNN-LSTM Model

In this phase, the hybrid CNN-LSTM neural network model predicts the detected ranges of missing values from the input time-series dataset. The missing detected values should be within the defined ranges such as 7, 15, and 30 time-steps. Hybrid CNN-LSTM model input sampled before prediction. A CNN-LSTM model generates a single one-step forecast, and this forecast value is fed further at the end of actual input for subsequent forecasts, as shown in Fig. 5. This forecasting process will continue until the one-step predicted values length of the CNN-LSTM model reaches the defined detected range of missing values.



```
C:\Users\Nazir\anaconda3\envs\tensorflow\python.exe E:/Python/Framework/index.py
2020-11-23 14:21:16.187246: W tensorflow/stream_executor/platform/default/dso_loader.cc:59] C
2020-11-23 14:21:16.237794: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above
Initiate Transformation
Total Records: 534
Starting Date: 2017-12-19
Total Record(s) After Transformation: 610
Missing values found At: 2018-06-19
Missing ranged: 3
183
(21, 1, 7, 1)
2020-11-23 14:22:02.824799: W tensorflow/stream_executor/platform/default/dso_loader.cc:59] C
2020-11-23 14:22:02.840637: W tensorflow/stream_executor/cuda/cuda_driver.cc:312] failed call
2020-11-23 14:22:03.020647: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrie
2020-11-23 14:22:03.022108: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostna
2020-11-23 14:22:03.125949: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorF
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2020-11-23 14:22:03.853924: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x2
2020-11-23 14:22:03.855234: I tensorflow/compiler/xla/service/service.cc:176]   StreamExecuto
1 day output [5418.7773]
2 day output [5418.7773, 4111.2617]
3 day output [5418.7773, 4111.2617, 4130.33]
4 day output [5418.7773, 4111.2617, 4130.33, 4107.716]
5 day output [5418.7773, 4111.2617, 4130.33, 4107.716, 3678.7634]
6 day output [5418.7773, 4111.2617, 4130.33, 4107.716, 3678.7634, 3678.7634]
7 day output [5418.7773, 4111.2617, 4130.33, 4107.716, 3678.7634, 3678.7634, 3937.247]
```

Fig. 5. Running steps of our proposed framework.

3.4 Reconstruction

In this phase, the predicted values of the CNN-LSTM model from the given input time-series dataset replaced by the detected missing value gaps. When we run our proposed framework by providing the air-condition appliance electricity consumption time-series missing values dataset as an input, the framework follows all the above steps represented in the framework architecture shown in Fig. 3. The running output of our proposed framework shown in Fig. 5.

The proposed framework predicts large gaps of missing values from the given input time-series missing values dataset. Figs. 6 and 7 represent the given input missing and generated output predicted values of our proposed framework.

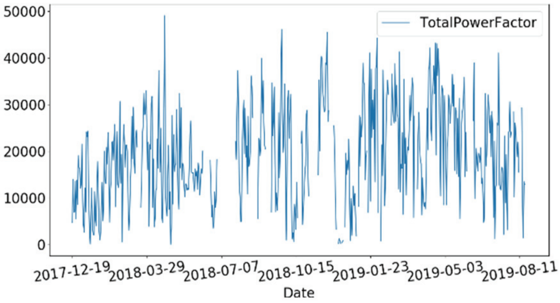


Fig. 6. Input time-series missing values to the proposed framework.

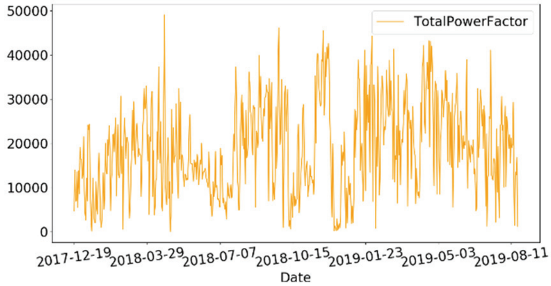


Fig. 7. Output time-series missing values predicted by the proposed framework.

4. Experiment

The experiment aims to identify an efficient deep neural network (DNN) sequence model for our proposed framework, which will predict the large gaps of missing values from the beLyfe SHS appliances electricity consumption time-series datasets. In this experiment, we compare the performance of CNN, LSTM, and the hybrid CNN-LSTM models on the prediction of air-condition appliance time-series dataset. We compare each model predictive accuracy and computational complexity on three different time-series data ranges, such as 7, 15, and 30 time-step. The dataset used in the experiment only contains total active power (TAP) feature values corresponding to a specific date. The TAP values measured in kilowatt (kW) unit and its values ranges have been summarized in a boxplot, as shown in Fig. 8. This time-series data is nonlinear in form and contain abrupt shifts. Fig. 9 represents the histogram and kernel density estimate (KDE) plot of our time-series dataset used in this experiment. We can determine from Fig. 9 that the distribution of values is not consistent or smooth.

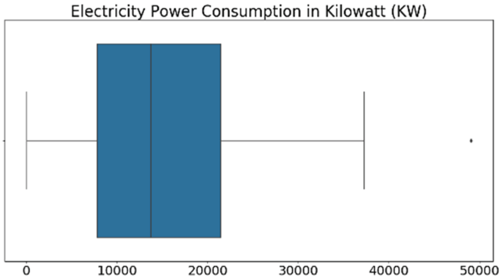


Fig. 8. A boxplot of four-month time-series data.

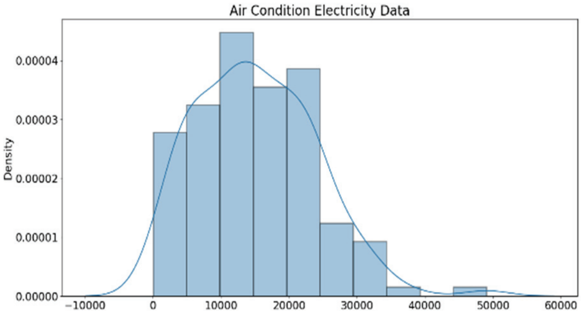


Fig. 9. A histogram and KDE plot of 4-month time-series data.

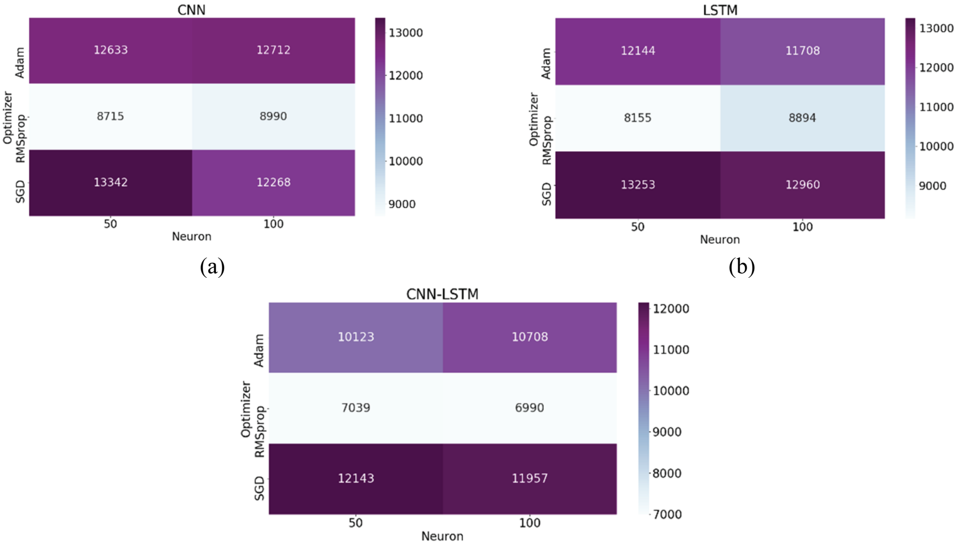
The experiment implemented on PyThon 3.3.7 with TensorFlow 1.12.0 deep learning library. Initially, the significant hyperparameters used in the training configuration of CNN, LSTM, and hybrid CNN-LSTM models chosen carefully. These chosen hyperparameters are the number of cells/neurons, the activation function, and the optimization algorithm. The suitable hyperparameters discovery is an essential part of training DNN models on the required input datasets [28]. The forecasting performance of CNN, LSTM, and hybrid CNN-LSTM will increase if the model best fits the training dataset, which is possible only when the selected hyperparameters values chosen according to the data characteristic.

We performed hyperparameter tuning on the chosen hyperparameters different values to select suitable hyperparameters for each model. These chosen hyperparameter different values on which we performed tuning to get the best combination between these hyperparameters values, which are 50, 100 cells/ neurons, the ADAM, SGD, and root mean square propagation (RMSprop) optimization algorithms and the ReLU activation function.

These chosen hyperparameter values of different combinations are tested on each model to determine a single combination that generates lower predictive errors on the various appliances time-series datasets. The tuning results on different hyperparameter combination values on each model are represented in Fig. 10, as shown below.

Fig. 10 shows the tuning results, which indicate that the RMSprop optimization algorithm with the 50 number of cells/neurons would generate the least prediction errors on appliances electricity consumption time-series datasets.

The batch size and the number of epochs are also carefully selected for each model to achieve the best fit train model on the input data set that takes the least training computation time. The batch size defines the whole dataset segmentation into the predefined number of parts called samples, independently processed by the neural network. The epochs define the number of time the training dataset is processed in the network optimization algorithm to update internal model parameters. For this experiment, the chosen batch size and epochs are 32 and 500, respectively.



(a) (b)

(c)

Fig. 10. The RMSE scores on two different hyperparameters, such as neuron size and optimization algorithm: (a) CNN, (b) LSTM, and (c) CNN-LSTM.

An early stopping method [29] is used in this experiment to avoid overfitting and stop training once the model performance stops improving. A split validation parameter with a value between 0 and 1 required for early stopping is defined as 0.67, indicating that the models use 67.7% input data for training and 33.3% for validation. Each model prediction performance compares on three separate prediction ranges, such as 7, 15, and 30, so the model historical data size used as an input set to four times more extensive than the defined prediction range.

A recursive forecast technique applied in this experiment, in which the models make one-step predictions and the output feed as inputs for the subsequent predictions. The data is normalized and scaled between 0 and 1 before the experiment to increase the models training efficiency. Eq. (2) used to scale and normalized the input time-series data for this experiment.

$$y = \frac{(x - min)}{(max - \mathrm{min})} \tag{2}$$

Each model training hyperparameter configurations are the same in this comparison experiment to evaluate predictive performance under the same conditions.

5. Performance Analysis

This section discussed the performance comparison of each predictive model used in the experiment. Each model prediction accuracy on three different prediction ranges measured by root mean square error (RMSE) residual errors measure. Eq. (3) represents the RMSE residual error measurement matric.

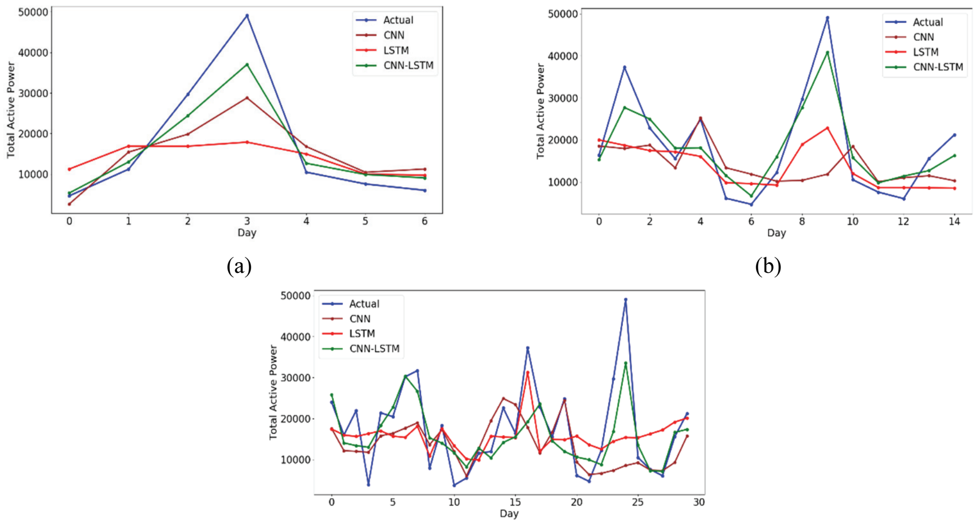
$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

Table 1 shows that the hybrid CNN-LSTM model prediction accuracy is better than the single CNN and LSTM models. The average prediction errors of the CNN-LSTM model on three different time-steps prediction ranges is 48.4% lower than CNN and 48.8% lower than LSTM. The prediction results of CNN, LSTM, and the hybrid CNN-LSTM model on three different time-step prediction ranges, such as 7, 15, and 30, are shown in Fig. 11(a), (b), and (c), respectively.

Table 1. Performance comparison of three diverse neural network

Time intervals	**RMSE**		
	CNN	**LSTM**	**CNN-LSTM**
7	9281	13356	5284
15	12886	10109	4908
30	10888	9827	6844
Average	11018	11097	5678

We also determine that the hybrid CNN-LSTM model is less computationally expensive and a good fit on the input dataset while training three different time-step prediction ranges such as 7, 15, and 30 time-steps. CNN-LSTM model fitness determined by evaluating the training and validation loss plots over a range of epoch cycles taken during training. When the model fit function called during compilation, it returns the model training history, including training and validation losses. Train and validation losses calculated at every epoch during model training, which allows us to diagnose model training fitness on input datasets.



(a)

(b)

(c)

Fig. 11. The prediction results of each DNN model on three different time-steps: (a) 7 time-steps, (b) 15 time-steps, and (c) 30 time-steps.

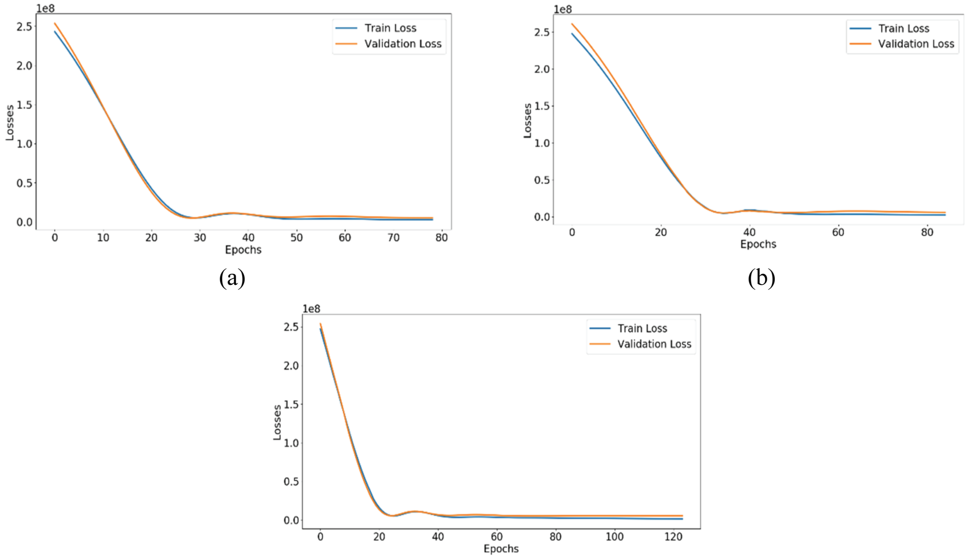
The DNN models fitness on the given input dataset is primary classified into three categories: a good-fit, over-fit, and under-fit. If training and validation losses decrease and stabilize around at same point is considered a good-fit model. If the validation loss of a model keeps decreasing at a certain level and afterwards starts increasing, this is considered overfitting. Furthermore, if model validation loss is higher than train loss and tends to improve further, it is considered an under-fit model.

The hybrid CNN-LSTM model fits each of three different time-step prediction ranges shown in Fig. 12. It observed that training and validation loss from the hybrid CNN-LSTM model indicates a good-fit training condition on each of three distinct time-step prediction ranges. The model reaches 120 maximum range of epoch cycles only at 30 days' time-step predictions, as shown in Fig. 9. The number of epoch cycles on remaining time-steps prediction ranges such as forecast intervals of 7 and 15 days is below 100, suggesting that the model is less expensive in computation and good fits on an input training dataset.

Overall, the CNN-LSTM model average prediction error score for three different time intervals is comparatively lower than the single CNN and LSTM models. Furthermore, the CNN-LSTM average computational time measured in seconds is also comparatively lower than the single CNN and LSTM models, as shown in Fig. 12(c).

Fig. 13 represents a boxplot of each model training computation time for three different prediction ranges. We can determine from the experiment that the hybrid CNN-LSTM sequence model could be used to predict large gaps of missing values in our proposed framework.

Prediction of missing values large gaps in the electricity consumption time-series dataset is difficult due to the nonlinear data patterns that contain abrupt shifts. The recurrent neural network models in deep learning, such as LSTM and hybrid CNN-LSTM, are explicitly designed to predict data in the form of sequences such as time series. These models preserve the contextual information in the memory cells of previous input observations and effectively map an input to the output.



(a)

(b)

(c)

Fig. 12. CNN-LSTM model train and validation loss graph on three different time intervals of input data training: (a) 7 days, (b) 15 days, and (c) 30 days.

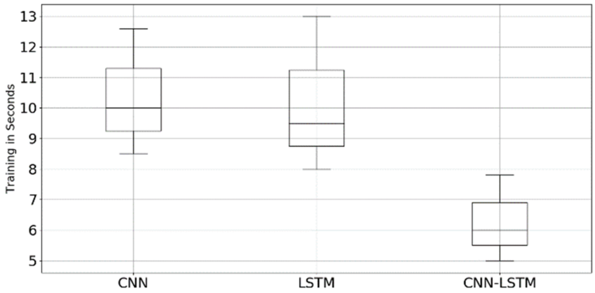


Fig. 13. The average training computation time of CNN, LSTM, and hybrid CNN-LSTM sequence models.

The average RMSE values of each model on three different time-steps are exceptionally high. The major problem is the lack of observed time-series data used to forecast and fulfil the specified range of missing values. The traditional statistical and machine learning methods, such as autoregressive integrated moving average (ARIMA) [30] and LSSVM [31], used in forecasting time-series data, would not be appropriate in this case. These models perform well when the time-series data patterns involve some sort of trend and seasonality.

Furthermore, the proposed framework is compatible with other time-series missing values dataset rather than electricity load data. The specified prediction ranges of this framework could extend according to the requirements. Our proposed framework prediction accuracy will further enhance if we incorporate multiple input features to support multivariate forecasting instead of univariate.

6. Conclusion

Missing values is one of the significant data quality issues in IoT-based SHS. If the collected data is missing or incomplete, it could eventually be useless. This paper proposed a novel framework for predicting large missing values gaps observed in IoT based home appliances electricity consumption time-series datasets. Missing values in SHS appliances datasets observed at 7, 15, and 30 time-steps, so we have defined these predictive ranges in our proposed framework to forecast the missing values within these ranges. The proposed framework automatically detects, predicts, and reconstructs the missing values within the defined ranges—a hybrid CNN-LSTM neural network used in the proposed framework to forecast the detected ranges of missing values. The CNN-LSTM model input historical data for training is four-times more extensive than the defined detected range after sampling.

A comparison experiment performed to analyze the forecasting performance of the CNN-LSTM model with its single variant such as CNN and LSTM on the defined missing values ranges. A tuning on crucial selected hyperparameters performed to determine the best hyperparameters combination for each model training configuration in this experiment. It observed that the CNN-LSTM model generates fewer prediction errors on average and takes less time to train the input datasets. The experiment results confirm the selection of a hybrid CNN-LSTM neural network as a predictive model for our proposed framework.

Acknowledgement

The research has been supported by Telecom Malaysia (TM) Research & Development (R&D) Institute under the grant number MMUE/190007.02.

References

[1] N. H. Motlagh, M. Mohammadrezaei, J. Hunt, and B. Zakeri, “Internet of Things (IoT) and the energy sector,” *Energies*, vol. 13, no. 2, article no. 494, 2020. https://doi.org/10.3390/en13020494

[2] M. A. Rahman and A. T. Asyhari, “The emergence of Internet of Things (IoT): connecting anything, anywhere,” vol. 8, no. 2, article no. 40, 2019. https://doi.org/10.3390/computers8020040

[3] C. Paul, A. Ganesh, and C. Sunitha, “An overview of IoT based smart homes,” in *Proceedings of 2018 2nd International Conference on Inventive Systems and Control (ICISC)*, Coimbatore, India, 2018, pp. 43-46.

[4] T. Banerjee and A. Sheth, “IoT quality control for data and application needs,” *IEEE Intelligent Systems*, vol. 32, no. 2, pp. 68-73, 2017.

[5] H. Kang, “The prevention and handling of the missing data,” *Korean Journal of Anesthesiology*, vol. 64, no. 5, pp. 402-406, 2013.

[6] A. N. Baraldi and C. K. Enders, “An introduction to modern missing data analyses,” *Journal of School Psychology*, vol. 48, no. 1, pp. 5-37, 2010.

[7] D. Sovilj, E. Eirola, Y. Miche, K. M. Bjork, R. Nian, A. Akusok, and A. Lendasse, “Extreme learning machine for missing data using multiple imputations,” *Neurocomputing*, vol. 174, pp. 220-231, 2016.

[8] P. E. Bunney, A. N. Zink, A. A. Holm, C. J. Billington, and C. M. Kotz, “Orexin activation counteracts decreases in nonexercise activity thermogenesis (NEAT) caused by high-fat diet,” *Physiology & Behavior*, vol. 176, pp. 139-148, 2017.

[9] J. Poulos and R. Valle, "Missing data imputation for supervised learning," *Applied Artificial Intelligence*, vol. 32, no. 2, pp. 186-196, 2018.

[10] X. Xu, W. Chong, S. Li, A. Arabo, and J. Xiao, "MIAEC: missing data imputation based on the evidence chain," *IEEE Access*, vol. 6, pp. 12983-12992, 2018.

[11] I. Lana, I. I. Olabarrieta, M. Velez, and J. Del Ser, "On the imputation of missing data for road traffic forecasting: new insights and novel techniques," *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 18-33, 2018.

[12] S. Sridevi, S. Rajaram, C. Parthiban, S. SibiArasan, and C. Swadhikar, "Imputation for the analysis of missing values and prediction of time series data," in *Proceedings of 2011 International Conference on Recent Trends in Information Technology (ICRTIT)*, Chennai, India, 2011, pp. 1158-1163.

[13] T. A. Mohamed, N. El Gayar, and A. F. Atiya, "Forward and backward forecasting ensembles for the estimation of time series missing data," in *ANNPR 2014: Artificial Neural Networks in Pattern Recognition*. Cham, Switzerland: Springer, 2014, pp. 93-104.

[14] S. F. Wu, C. Y. Chang, and S. J. Lee, "Time series forecasting with missing values," in *Proceedings of 2015 1st International Conference on Industrial Networks and Intelligent Systems (INISCom)*, Tokyo, Japan, 2015, pp. 151-156.

[15] A. S. Dhevi, "Imputing missing values using inverse distance weighted interpolation for time series data," in *Proceedings of 2014 6th International Conference on Advanced Computing (ICoAC)*, Chennai, India, 2014, pp. 255-259.

[16] E. P. Caillault, A. Lefebvre, and A. Bigand, "Dynamic time warping-based imputation for univariate time series data," *Pattern Recognition Letters*, vol. 139, pp. 139-147, 2020.

[17] N. Bokde, M. W. Beck, F. M. Alvarez, and K. Kulat, "A novel imputation methodology for time series based on pattern sequence forecasting," *Pattern Recognition Letters*, vol. 116, pp. 88-96, 2018.

[18] Z. Ding, G. Mei, S. Cuomo, Y. Li, and N. Xu, "Comparison of estimating missing values in IoT time series data using different interpolation algorithms," *International Journal of Parallel Programming*, vol. 48, pp. 534-548, 2020.

[19] A. Chaudhry, W. Li, A. Basri, and F. Patenaude, "A method for improving imputation and prediction accuracy of highly seasonal univariate data with large periods of missingness," *Wireless Communications and Mobile Computing*, vol. 2019, article no. 4039758, 2019. https://doi.org/10.1155/2019/4039758

[20] T. Kim, W. Ko, and J. Kim, "Analysis and impact evaluation of missing data imputation in day-ahead PV generation forecasting," *Applied Sciences*, vol. 9, no. 1, article no. 204, 2019. https://doi.org/10.3390/app9010204

[21] N. Al-Milli and W. Almobaideen, "Hybrid neural network to impute missing data for IoT applications," in *Proceedings of 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, Amman, Jordan, 2019, pp. 121-125.

[22] K. Zor, O. Celik, O. Timur, H. B. Yildirim, and A. Teke, "Simple approaches to missing data for energy forecasting applications," in *Proceedings of the 16th International Conference on Clean Energy (ICCE)*, Gazimagusa, Turkey, 2018.

[23] X. Cao, S. Dong, Z. Wu, and Y. Jing, "A data-driven hybrid optimization model for short-term residential load forecasting," in *Proceedings of 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, Liverpool, UK, 2015, pp. 283-287.

[24] T. Y. Kim and S. B. Cho, "Predicting residential energy consumption using CNN-LSTM neural networks," *Energy*, vol. 182, pp. 72-81, 2019.

[25] X. Shao, C. S. Kim, and P. Sontakke, "Accurate deep model for electricity consumption forecasting using multi-channel and multi-scale feature fusion CNN–LSTM," *Energies*, vol. 13, no. 8, article no. 1881, 2020. https://doi.org/10.3390/en13081881

[26] J. Du Preez and S. F. Witt, “Univariate versus multivariate time series forecasting: an application to international tourism demand,” *International Journal of Forecasting*, vol. 19, no. 3, pp. 435-451, 2003.

[27] K. Yan, X. Wang, Y. Du, N. Jin, H. Huang, and H. Zhou, “Multi-step short-term power consumption forecasting with a hybrid deep learning strategy,” *Energies*, vol. 11, no. 11, article no. 3089, 2018. https://doi.org/10.3390/en11113089

[28] M. Massaoudi, S. S. Refaat, I. Chihi, M. Trabelsi, H. Abu-Rub, and F. S. Oueslati, “Short-term electric load forecasting based on data-driven deep learning techniques,” in *Proceedings of the 46th Annual Conference of the IEEE Industrial Electronics Society (IECON)*, Singapore, 2020, pp. 2565-2570.

[29] M. Li, M. Soltanolkotabi, and S. Oymak, “Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks,” 2019 [Online]. Available: https://arxiv.org/abs/1903.11680.

[30] C. Nichiforov, I. Stamatescu, I. Fagarasan, and G. Stamatescu, “Energy consumption forecasting using ARIMA and neural network models,” in *Proceedings of 2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE)*, Galati, Romania, 2017, pp. 1-4.

[31] F. Kaytez, M. C. Taplamacioglu, E. Cam, and F. Hardalac, “Forecasting electricity consumption: a comparison of regression analysis, neural networks and least squares support vector machines,” *International Journal of Electrical Power & Energy Systems*, vol. 67, pp. 431-438, 2015.

Syed Nazir Hussain **https://orcid.org/0000-0003-4922-0330**

He received a B.S. degree in Computer Science from Usman Institute of Technology, affiliated with Hamdard University, Karachi, Pakistan in 2017. Since November 2019, he is with the Faculty of Engineering from Multimedia University, Malaysia, as an M.Eng.Sci. candidate. His current research interests include data analysis and data quality improvement. He had previously worked as a software engineer for Golpik Inc., a software company.

Azlan Abd Aziz **https://orcid.org/0000-0003-2255-0831**

He is a TM staff and currently attached to the Faculty of Engineering and Technology. He has been in the telecommunication industry for more than 15 years, with a couple of years in TM R&D working on next-generation wireless networks.

Md. Jakir Hossen **https://orcid.org/0000-0002-9978-7987**

He is currently working as a senior lecturer in the Department of Robotics and Automation, Faculty of Engineering and Technology, Multimedia University, Malaysia. His research interests are artificial intelligence techniques in data analytics, robotics control, data classification and predictions.

Nor Azlina Ab Aziz **https://orcid.org/0000-0002-2119-6191**

She is a lecturer in the Faculty of Engineering and Technology at Multimedia University, Melaka. Her focus is in the area of swarm intelligence and nature-inspired optimization algorithm.

G. Ramana Murthy **https://orcid.org/0000-0002-3556-9294**

He 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 Ph.D. from Multimedia University, Malaysia and secured the grant from TM R&D Telekom Malaysia in 2019. Currently he is working as a 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, and evolutionary algorithms.

Fajaruddin Bin Mustakim **https://orcid.org/0000-0001-5204-7319**

He received a bachelor’s degree of Civil Engineering from (UiTM) University, and subsequently receiving an M.Sc. in Construction Management from Universiti Teknologi Malaysia (UTM). He was awarded a Ph.D. Engineering in Scientific and Engineering Simulation from the Nagoya Institute of Technology (NIT), Japan. Currently appoints as consultant project entitled “TMR Asynchronous V2V with NLoS Vehicular Sensing (V2V)” funded by TM R&D at Multimedia University.