IET Smart Cities

Research Article

PowerNet: a smart energy forecasting architecture based on neural networks

eISSN 2631-7680
Received on 3rd February 2020
Revised 7th October 2020
Accepted on 7th October 2020
doi: 10.1049/iet-smc.2020.0003
www.ietdl.org

Yao Cheng[1], Chang Xu[2], Daisuke Mashima[3] ✉, Partha P. Biswas[3], Geetanjali Chipurupalli[3], Bin Zhou[3], Yongdong Wu[4]

[1]*Institute for Infocomm Research, A*STAR, Singapore*
[2]*School of Computing Engineering, Nanyang Technological University, Singapore*
[3]*Advanced Digital Sciences Center, Singapore, Singapore*
[4]*Department of Computer Science, Jinan University, Guangzhou, People's Republic of China*
✉ *E-mail: daisuke.m@adsc-create.edu.sg*

Abstract: Electricity demand forecasting is a critical task for efficient, reliable and economical operation of the power grid, which is one of the most essential building blocks of smart cities. Accurate forecasting allows grid operators to properly maintain the balance of supply and demand as well as to optimize operational cost for generation and transmission. This article proposes a novel neural network architecture *PowerNet* which can incorporate multiple heterogeneous features such as historical energy consumption data, weather data and calendar information for the demand forecasting task. Using real-world smart meter dataset, we conduct an extensive evaluation to show the advantages of *PowerNet* over recently-proposed machine learning methods such as Gradient Boosting Tree (GBT), Support Vector Regression (SVR), Random Forest (RF) and Gated Recurrent Unit (GRU). *PowerNet* demonstrates notable performance in reducing both the median and worst-case prediction errors when forecasting demands of individual residential households. We further provide empirical results concerning the two operational considerations that are crucial when using *PowerNet* in practice: the time horizon the model can predict with a decent accuracy and the frequency of training the model to retain its modeling capability. Finally, we briefly discuss a multi-layer anomaly/ electricity-theft detection approach based on *PowerNet* demand forecasting.

1 Introduction

The smart grid is an integral part of modern smart cities. It is the enhanced electrical grid that takes advantage of sensing and information communication technologies to improve the efficiency, reliability and security of traditional power grid. Compared to the traditional power grid, entities in the smart grid are able to obtain timely power grid status of many kinds. Smart metering, which is a major improvement brought by the smart grid, facilitates real-time metering and reporting of electricity consumption data. One resulting benefit is that the accurate, fine-grained *power demand forecasting* can be carried out based on such meter measurement. Such forecasting based on the historical data facilitates power generation scheduling and power dispatching in a future period.

Demand forecasting is important in demand management for both power companies and electricity customers [1]. Power companies can allocate proper resources to balance the supply and demand based on the demand forecasting results. They can also adjust the demand response strategy such as dynamic pricing to shape the load so as to avoid the infrastructure capacity strain or to avoid additional cost for starting plants operating near to their peak. Furthermore, the utilities can detect abnormal meter measurements, caused either by unexpected meter failures or deliberate meter manipulation, by identifying those measurements that do not conform to the predicted/expected values. For the electricity customers, power demand forecasting provides them with their expected power consumption and cost in a future period under dynamic pricing strategy, so that they can adjust their usage schedule accordingly to achieve a lower cost. Therefore, the importance of accurate demand forecasting for effective and efficient management of the smart grid in a smart urban set up is paramount.

Although demand forecasting has been widely studied for years, attaining high accuracy in forecasting is a challenge as power demand is dependent upon various factors which may have discriminative capability in influencing the demand. With this challenge in mind, we propose a novel forecasting neural network architecture named as *PowerNet*. We take into account a set of features from three heterogeneous dimensions – the historical consumption data, the weather information and the calendar information, all of which are considered influential on electricity customers' power consumption patterns. In each dimension, a set of features is developed. Thereafter, we introduce our model *PowerNet* which is capable of incorporating all the designed features. The key property of *PowerNet* is the ability to model both sequential data (i.e. historical consumption data) and non-sequential data (i.e. weather and calendar information) in a unified manner. The underpinning idea lies in the use of recurrent neural network (RNN) for encoding dependencies implied in sequential data and multilayer perceptron (MLP) network for capturing correlations between non-sequential features and predictions. In order to evaluate the effectiveness of our model, we compare *PowerNet* with four state-of-the-art demand forecasting techniques which are gradient boosting tree (GBT) [2], support vector regression (SVR) [3], random forest (RF) [4, 5] and gated recurrent unit (GRU) [6]. We show that the performance of our proposed *PowerNet* model is competitive in the case studies of predicting demands of smart apartments in a smart city. Furthermore, we tackle two crucial questions that need to be answered when operating *PowerNet* in practice: how far in the future the model can forecast with a reasonable accuracy and how often should we train the forecasting model to retain its modelling capability? Lastly, we discuss a multilayer data-driven anomaly detection approach based on *PowerNet*.

The contributions of this work are summarised below:

- We propose *PowerNet*, a novel power demand forecasting neural network that captures heterogeneous features in a unified way.
- We compare *PowerNet* with four representative models adopted in recent research works, i.e. GBT, SVR, RF and GRU. The results show that *PowerNet* can reduce and bound the error over

these competitors, even in the prediction of individual apartment energy demand.
- We further evaluate the forecasting model under different forecasting duration and re-training frequency using publicly available datasets.
- We provide brief discussion on potential application of *PowerNet* for anomaly/electricity-theft detection.

The rest of this paper is organised as follows. In Section 2, we discuss the features to be incorporated into power demand forecasting. Section 3 elaborates the design of *PowerNet*. We discuss evaluation results, including comparison with state-of-the-art techniques and empirical results to answer the aforementioned questions for practical operation in Section 4, followed by a brief discussion about the application for anomaly detection in Section 5. Related work is discussed in Section 6 and we conclude the paper in Section 7.

2 Feature design and dataset

Power consumption patterns are affected by a variety of factors. Thus demand forecasting mechanism should incorporate such factors as features, in addition to historical energy consumption data. We focus on weather and calendar data. Below, we elaborate these features and the dataset we have utilised in this paper.

2.1 Energy usage dataset

We use the publicly available dataset provided by the University of Massachusetts [7]. It includes two parts, the apartment dataset and the weather dataset.

The apartment dataset contains consumption data for 114 single-family apartments located in Western Massachusetts for the period of year 2014 to 2016. The dataset records the demand of every single apartment in fixed temporal frequency. Given the metering interval is fixed, power values are able to represent the power consumption. The metering frequency is once every 15 min for the year 2014 and 2015 (till December 15), and once every 1 min for the year 2016. The data is in *.csv* files, each of which records the power consumption details for one apartment within 1 year with apartment ID as its file name.

Together with the power consumption data, hourly weather information during the record period from 2014 to 2016 is also available. Fourteen meteorological attributes are included in the weather dataset, including weather summary, temperature, humidity, cloud cover, wind speed, wind bearing, visibility, pressure and so on. In our experiment, we use the data of 2016 because of its finer granularity in recording frequency as well as the latest consumption pattern it may reflect.

2.2 Feature design

The features used by the existing forecasting models fall into three categories in terms of privacy issue, i.e. publicly available information (e.g. weather information and calendar information), household private information (e.g. demography) and quasi-private information (e.g. historical consumption data acquired by power utility companies). The quasi-private information here is defined as privacy-related but not public available data. For example, the historical electricity consumption data can be used to infer certain private household characteristics [8, 9], but it is only available to the authorised personnel within power utility companies instead of to the public.

Although it is natural that private household data would have a direct influence on the household power demand, e.g. more people living in the house leads to more power demand, in this work, we limit the predictors to non-private information due to the following reasons. First of all, we would like to involve no household-specific data in forecasting procedure other than power meter readings due to user privacy concern. Secondly, some utility companies may have access to household private data such as locations. However, it is not common for utility companies to have other private information, for example, the demography information. Thirdly, the forecasting model independent of the house specific data can be applied to larger scales easily, such as building level or area level.

Table 1 Features for the power demand forecasting task

Category	Detail
historical consumption data	consumption data in past n_d time slots
weather information	weather summary, weather representation icon name, temperature, apparent temperature, cloud cover, precipitation probability, precipitation intensity, visibility, wind speed, wind bearing, humidity, pressure, dew point
calendar information	day of the month, day of the week, hour of the day, period of the day (i.e. daytime and night time), is weekend (Boolean value)

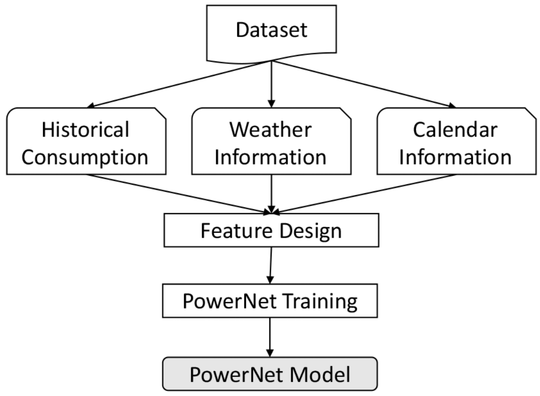

Fig. 1 *Approach overview*

We develop three categories of features from the dataset, i.e. historical consumption data, weather information and calendar information. Historical consumption data is the actual observation of the prediction target, which directly reflects the consumption pattern. Power utility companies can get this data by reading power meters. Weather information has an influence on the power demand since some appliances are sensitive towards weather conditions. For example, the use of air conditioner depends on the temperature and humidity. Calendar information, such as weekday or weekend, shapes the user consumption behaviour in terms of different living/ working styles. It indicates the consumption pattern according to the calendar feature and cycle.

Our features based on the above three categories are summarised in Table 1. There are $n_d + 18$ features in total, among which, n_d features are from historical consumption data, 13 are from weather information and 5 are designed from calendar information. The historical data involves a large number of data points. Therefore, it is necessary to find out n_d historical data points that are most correlated with the target forecasting value. To solve this problem, we use the autocorrelation function, which can quantify the correlation between data points in the same time series of various lags, to find out the most related number of lag values as n_d.

3 PowerNet

3.1 Overview

Our approach to forecasting power demand is by modelling the relationship between the target power demand and a set of indicative features. Fig. 1 illustrates the high-level process sketching our approach. First, we extract several suites of features discussed in Section 2.2 from the datasets. Then, we train the proposed *PowerNet* model by feeding the feature vectors as input, supervised by the signals from the power demand ground-truth. Fig. 2 shows the architecture of *PowerNet*, which consists of two major components. The left component (in blue) is designed to model the historical consumption time series data. The idea is to

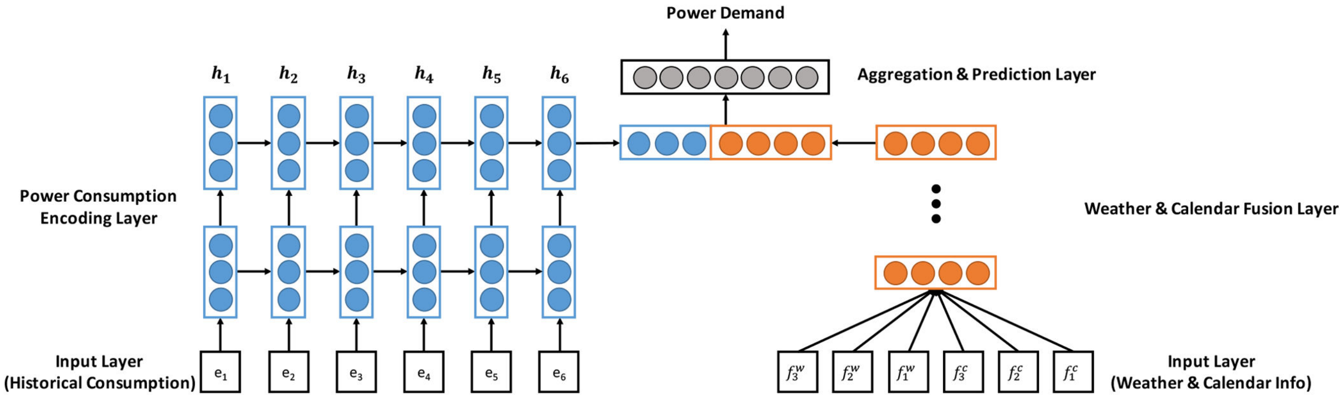

Fig. 2 *Architecture of PowerNet*

capture the temporal effects of power consumption as future consumption may be correlated to consumption in the recent past. Here, we utilise the long short-term memory (LSTM) [10] network to *encode* the correlations between consecutive power demands across time. The right component (in orange) is a MLP [11] model for modelling the non-linearity between the weather and calendar features, and the target power demand. Finally, we aggregate the outputs of these two components and make the final prediction of the target power demand via a prediction layer. In the following, we dissect each component of *PowerNet*.

3.2 Input layer

To process the sequential historical energy consumption data and the non-sequential weather and calendar data, the input layer of *PowerNet* consists of two components for each of the data types. The first is a sequence of historical power consumption values denoted by $\boldsymbol{E} = \{e_1, \ldots, e_t, \ldots, e_{|E|}\}$ where $|\boldsymbol{E}|$ is the cardinality of $\boldsymbol{E}$ (i.e. the number of meter readings), with each value $e_t \in \mathbb{R}^+$ is a real-valued non-negative power meter reading at time t. The second is a vector of weather and calendar features, denoted by $\boldsymbol{F} = [F^w; F^c]$, where $\boldsymbol{F}^w = \{f_1^w, \ldots, f_{|w|}^w\}$, $\boldsymbol{F}^c = \{f_1^c, \ldots, f_{|c|}^c\}$, $[;]$ represents vector concatenation, and $|w|$ and $|c|$ are the numbers of the weather and calendar features, respectively, already introduced in Section 2.2.

3.3 Power consumption encoding layer

The utility of this layer is to encode the power consumption time series data by using an LSTM network, a widely-used variant of RNN that can learn long-term dependencies. Different from traditional neural networks that can only take historical energy consumption readings as input, LSTM allows unlimited history information to persist with an internal loop mechanism while avoiding the gradient vanishing problem [12]. Therefore, it has been successfully applied to various areas, e.g. continual prediction [13], language modelling [14] and translation [15]. The core of LSTM is a memory cell that can maintain information across time via gating mechanism. The LSTM cell maintains a cell status $\boldsymbol{s}$ based on both current input e_t and previous output $\boldsymbol{h}_{t-1}$ (i.e. the recurrent input) and then decides what information to be dropped and what to be passed on (i.e. $\boldsymbol{h}_t$). We do not detail the gating mechanisms here which can be found in previous literature [10]. We use LSTM$(\cdot)$ to represent the cell function.

Specifically, we apply a stacked LSTM to every time step of the power consumption time series data $\boldsymbol{E}$

$$\begin{aligned} [\boldsymbol{h}_1\ \boldsymbol{s}_1] &= \mathrm{LSTM}^{\text{stack}}(e_1, \boldsymbol{h}_0, \boldsymbol{s}_0) \\ &\cdots \\ [\boldsymbol{h}_t\ \boldsymbol{s}_t] &= \mathrm{LSTM}^{\text{stack}}(e_t, \boldsymbol{h}_{t-1}, \boldsymbol{s}_{t-1}) \\ &\cdots \\ [\boldsymbol{h}_{|E|}\ \boldsymbol{s}_{|E|}] &= \mathrm{LSTM}^{\text{stack}}(e_{|E|}, \boldsymbol{h}_{|E|-1}, \boldsymbol{s}_{|E|-1}) \end{aligned} \tag{1}$$

where $\boldsymbol{h}$ and $\boldsymbol{s}$ are the hidden and cell states of LSTM, respectively. Then, the output of LSTM$^{\text{stack}}$ at the last time step $\boldsymbol{h}_{|E|} \in \mathbb{R}^n$ is used as a final encoding of the entire power consumption series, where n is the LSTM memory size.

3.4 Weather and calendar fusion layer

Here we deal with the input from the weather and calendar feature vectors $\boldsymbol{F}^w$ and $\boldsymbol{F}^c$. Specifically, we jointly model these two feature vectors through an MLP network as follows:

$$\boldsymbol{h}_{wc} = \mathrm{ReLU}(\boldsymbol{W}_2\mathrm{ReLU}(\boldsymbol{W}_1[\boldsymbol{F}^w; \boldsymbol{F}^c] + \boldsymbol{b}_1) + \boldsymbol{b}_2) \tag{2}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{d_1 \times m}, \boldsymbol{W}_2 \in \mathbb{R}^{d_2 \times d_1}, \boldsymbol{b}_1 \in \mathbb{R}^{d_1}, \boldsymbol{b}_2 \in \mathbb{R}^{d_2}$ are trainable weights, $m = |\boldsymbol{F}^w| + |\boldsymbol{F}^c|$, d_1 and d_2 are the sizes of hidden units, $[;]$ denotes vector concatenation by column and $h_{wc} \in \mathbb{R}^{d_2}$ is the output encoding of this MLP. ReLU [16] is used as the activation function for introducing non-linearity.

3.5 Aggregation and prediction layer

Having both power consumption history and weather and calendar information encoded, we finally aggregate the obtained encoding representations and make the final power demand predictions. Concretely, we concatenate the two encoding representations $\boldsymbol{h}_{|E|}$ and h_{wc} and feed the result into a final feed-forward regression network

$$\hat{y} = \boldsymbol{W}_4\mathrm{ReLU}(\boldsymbol{W}_3[\boldsymbol{h}_{|E|}; h_{wc}] + \boldsymbol{b}_3) + b_4 \tag{3}$$

where $\boldsymbol{W}_3 \in \mathbb{R}^{d_3 \times (|E| + d_2)}, \boldsymbol{b}_3 \in \mathbb{R}^{d_3}, \boldsymbol{W}_4 \in \mathbb{R}^{1 \times d_3}, b_4 \in \mathbb{R}$ are trainable parameters and d_3 is the hidden size of the inner layer. Note that both $\boldsymbol{W}_4$ and b_4 of the outer layer have only one hidden unit for producing the final predicted reading value. $\hat{y} \in \mathbb{R}$ is the predicted power consumption reading value.

3.6 Optimisation

For model training, we use mean squared error loss ((4)) with dropout regularisation [17]

$$L(\boldsymbol{W}_*, \boldsymbol{b}_*) = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \tag{4}$$

where $\hat{y}_i$ and y_i are the ith predicted and actual energy consumption values, respectively, N is the number of training examples, $\boldsymbol{W}_*, \boldsymbol{b}_*$ are all the aforementioned trainable parameters in our model. In addition, all trainable parameters in the fully-connected layers are regularised by L2 norm. Finally, Adam (Adaptive Moment Estimation) [18] is used as the optimiser for stochastic gradient descent.

Table 2 MAPE and MSE on prediction of individual apartment consumption

Apartment (season)	Model	MAPE, %	MSE
69 (spring)	PowerNet	7.98	0.017
	SVR	8.69	0.018
	GBT	8.84	0.019
	RF	8.86	0.019
	GRU	9.80	0.026
91 (summer)	PowerNet	13.82	0.014
	SVR	106.75	0.016
	GBT	22.41	0.013
	RF	21.38	0.013
	GRU	21.00	0.015
39 (autumn)	PowerNet	16.73	0.213
	SVR	19.62	0.408
	GBT	19.85	0.368
	RF	22.83	0.449
	GRU	22.83	0.449

4 Evaluation

This section first compares *PowerNet* with several representative models used in recent works in terms of two quantitative metrics. Then, we evaluate *PowerNet* under different settings, including the forecasting frequencies, forecasting periods and the newness of *PowerNet*.

4.1 Preparation

4.1.1 Baseline: We select four prediction methods utilised in recent works as our baseline models in this work: GBT [2], SVR [3], RF [4, 5] and GRU [6]. For a fair comparison, we apply these models to the same public dataset as described in Section 2.1.

GBT is adopted by Bansal *et al.* [2] to forecast power consumption. GBT is a supervised learning predictive model which can be used for classification and regression purposes [19, 20]. GBT builds the model, i.e. a series of trees, in a step-wise manner. In each step, it adds one tree and maintains the existing trees unchanged. The added tree is the optimal tree by minimising a predefined loss function. In summary, the prediction model of GBT is formed with the ensemble of weaker prediction models following the core idea of the gradient.

Support vector machine (SVM) is used in the work by Yu *et al.* [3] to forecast power usage. SVM is a supervised machine learning algorithm for solving both classification and regression problems [21]. SVM performs classification by seeking the hyperplane that differentiates the two classes to the largest extent, i.e. maximising the margin. Similarly, regression using SVM is called SVR [22], and it is used to seek and optimise the generation bounds by minimising the predefined error function. The regression can be conducted in both linear and non-linear manner. For the non-linear SVR, it needs to transform the data into a higher dimensional space so that it is possible to perform the linear separation.

RF, an ensemble machine learning method consisting of many decision trees (DTs), has been widely used for classification and regression problems. A DT, also termed as classification and regression tree (CART), has many nodes where each node stores a test function to apply on incoming data [4]. In RF, the bagging principle is incorporated in which observations of a certain sample size (called bootstrap samples) are randomly selected from the training set to fit a regression tree. While single CART is sensitive to data noise, the bootstrap aggregation is immune to it to a large extent. RF shows competitive performance in time series forecasting as observed in recent studies on load demand forecasting [4, 5].

GRU is based on the framework of RNN. Conventional RNN network uses gradient descent method in back-propagation for learning. However, vanishing gradient in dealing with long time sequence is the common problem encountered in RNN. The vanishing gradient is countered by adding control gates for information buffer in both LSTM and GRU. In place of the hidden units in RNN, LSTM network uses LSTM cells consisting of the input gate, output gate and forget gate. GRU is simpler compared to LSTM as it combines the input gate and forget gate with a single gate called update gate. Wang *et al.* [6] performed short-term load forecasting using GRU and including factors such as weather, temperature, day and so on.

4.1.2 Evaluation metric: We introduce two metrics to evaluate the accuracy of the forecasting model, i.e. mean square error (MSE) and mean absolute percentage error (MAPE). The smaller the error is, the more accurate the model prediction is.

MSE measures the average of the squared errors/deviations as directed by (5). N_v is the total number of forecasting values, A_t denotes the actual value at time t and F_t denotes the forecasting value at time t. A smaller MSE value signifies better prediction

$$\text{MSE} = \frac{1}{N_v}\sum_{t=1}^{N_v}(A_t - F_t)^2 \tag{5}$$

Different from MSE, MAPE measures the error proportion to the absolute value. It expresses the error as a percentage and can be calculated using the following equation:

$$\text{MAPE} = \frac{100\%}{N_v}\sum_{t=1}^{N_v}\left|\frac{A_t - F_t}{A_t}\right| \tag{6}$$

MSE is more useful in comparison among identical test data as it is the absolute square error value, which depends on the scale of actual values. Compared to MSE, MAPE is more indicative in the comparison between different data since it represents the error in a percentage manner.

4.2 Comparison with baselines

In this sub-section, we present empirical results to demonstrate the advantage of *PowerNet* over the four baseline models. Our *PowerNet* uses a two-layered LSTM network. The cell memory size for every layer is tuned from the set {64, 128, 256, 512} using grid search. Early stopping is employed when there is no further improvement on the validation set.

Similarly, the parameters for baseline models are also automatically tuned in the same way. For GBT, three parameters are involved, i.e. the number of boosting stages to perform *n_estimators*, maximum depth of the individual regression estimators *max_depth* and learning rate *learning_rate*. Its parameter grid is constructed using *n_estimators*: {50, 100, 150, 200, 250, 300, 350, 400, 450, 500}, *max_depth*: {1, 2, 3, 4, 5} and *learning_rate*: {0.001, 0.01, 0.1, 1}. For SVR, three parameters *C*, *kernel* and *gamma* are involved. We construct the parameter grid using *C*: {0.001, 0.01, 0.1, 1}, *kernel*: {rbf, linear, poly, sigmoid} and hence *gamma* is automatically set to the corresponding kernel coefficient or the reciprocal of the number of features. For RF, we set the number of trees in the forest *n_trees*, maximum depth of the tree *max_depth* and minimum number of samples for a leaf, *min_samples_leaf*. Like other algorithms, parameters are automatically tuned from the search range *n_trees*: {50, 100, 150, 200, 250, 300, 350, 400, 450, 500}, *max_depth*: {80, 90, 100, 110} and *min_samples_leaf*: {3, 4, 5}. GRU is similar to LSTM network. The cell memory size for every layer is tuned from the set {32, 64, 96, 128, 256} using grid search.

We use the power consumption data of past 26 days, i.e. 624 h as the training set to train all the models and the next 48 h data, i.e. day 27-28 as the validation set. Finally, we make predictions on the test data of day 29-30. Due to space limitation, we demonstrate the results of our model and the four baselines on the data of only few randomly chosen apartments from different seasons (No. 69 in Spring, No. 91 in Summer and No. 39 in Autumn as seen in Table 2). The plots of predicted consumption patterns against the real consumption for the first two apartments are found in Figs. 3 and 4. From the figures we can see that *PowerNet* model captures

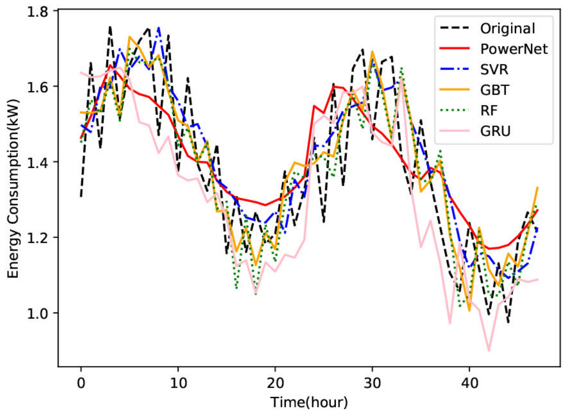

Fig. 3 *Forecasting results of apartment 69 (spring)*

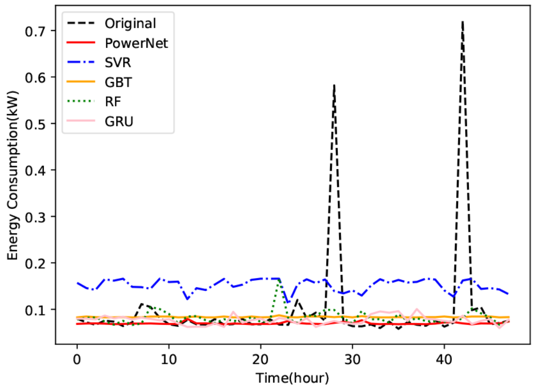

Fig. 4 *Forecasting results of apartment 91 (summer)*

trends well and offers accuracy improvement in MAPE for the selected datasets by 8, 34 and 14%, respectively, compared to the second-best model.

We further conduct experiments with more apartment data to compare the accuracy in terms of distribution of MAPE. We use summer season consumption data for randomly picked 50 apartments and perform the same experiment for each apartment to calculate MAPE. The result is summarised in Fig. 5. As can be seen, overall the MAPE of *PowerNet* is lower than the other competitors. In general, as predicting demand of individual household level is challenging, MAPE is often high. However, MAPE of *PowerNet* is still bounded below 100% and the median is below 50%.

Lastly, we conduct experiments with aggregated energy consumption data. We evaluate accuracy with 2 different aggregation levels, 16 apartments and 114 apartments (i.e. all apartments available in the dataset). For each case, we predict for 48 h as done in the experiments for individual apartments, and calculate MAPE and MSE. As seen in Table 3, when the aggregation level is low, *PowerNet* has advantage over the others. The corresponding plot is found in Fig. 6. On the other hand, when consumption values of all apartments are aggregated, all prediction models except GRU perform reasonably well. Based on these results as well as the results of individual apartment experiments discussed earlier, *PowerNet* exhibits competitive performance along with mostly better accuracy over the other models evaluated in our set-up. However, we admit that our results may have bias caused by the specific dataset we use in this study and evaluation with other datasets for generality will be part of our future work.

4.3 Forecasting period of PowerNet

In general, the accuracy of power demand forecasting deteriorates as the prediction horizon moves farther. Therefore, it is crucial for grid operators to know how much ahead in time the *PowerNet* can predict the demand without significant drop in accuracy. In this section, we provide empirical results on forecasting accuracy against different forecasting periods using the real-world electricity

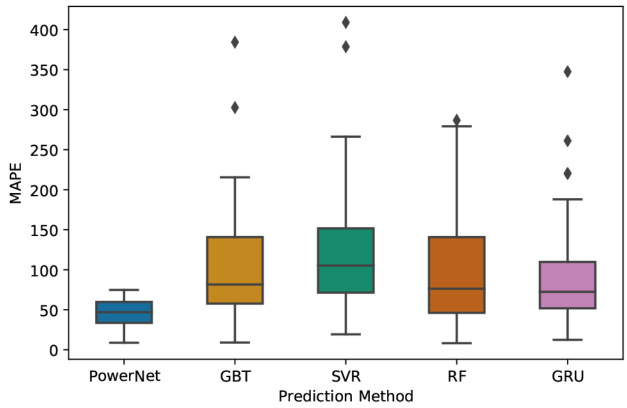

Fig. 5 *Comparison in distribution of MAPE*

Table 3 MAPE and MSE on prediction of aggregated consumption

No. of apartment	Model	MAPE, %	MSE
16	PowerNet	13.98	0.024
	SVR	15.61	0.034
	GBT	14.18	0.028
	RF	15.03	0.032
	GRU	16.03	0.036
114	PowerNet	10.00	0.012
	SVR	10.94	0.014
	GBT	8.48	0.009
	RF	9.71	0.012
	GRU	15.75	0.036

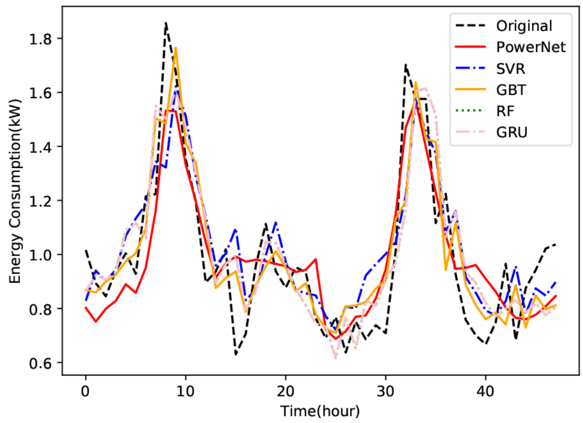

Fig. 6 *Forecasting results of aggregated consumption of 16 apartments (spring)*

consumption data. By doing so, grid operators can evaluate whether *PowerNet* is suitable for certain tasks that require different lengths of prediction period, such as bidding in the day-ahead electricity market and day-ahead electricity scheduling which require the one day-ahead forecasting results [23].

Some features for predicting the power demand in the far future may not be available at the time of prediction. For example, the power consumption of the previous 1 h is an important feature to predict the power demand for the next hour. If we predict beyond 1 h at once, we would not know the actual consumption value for every 'previous' hour. Therefore, the prediction in the far future relies on the predicted values prior to that. The fact has an inherent risk of error accumulation.

In this experiment, we predict the power demand for the future 30 days at once based on current historical data. We train the model on the aggregated historical data in July and predict the power demand for the following 30 days. The forecasting results are shown in Fig. 7 in red. We can see that the red line follows the original peaks and valleys well at the beginning. However, starting

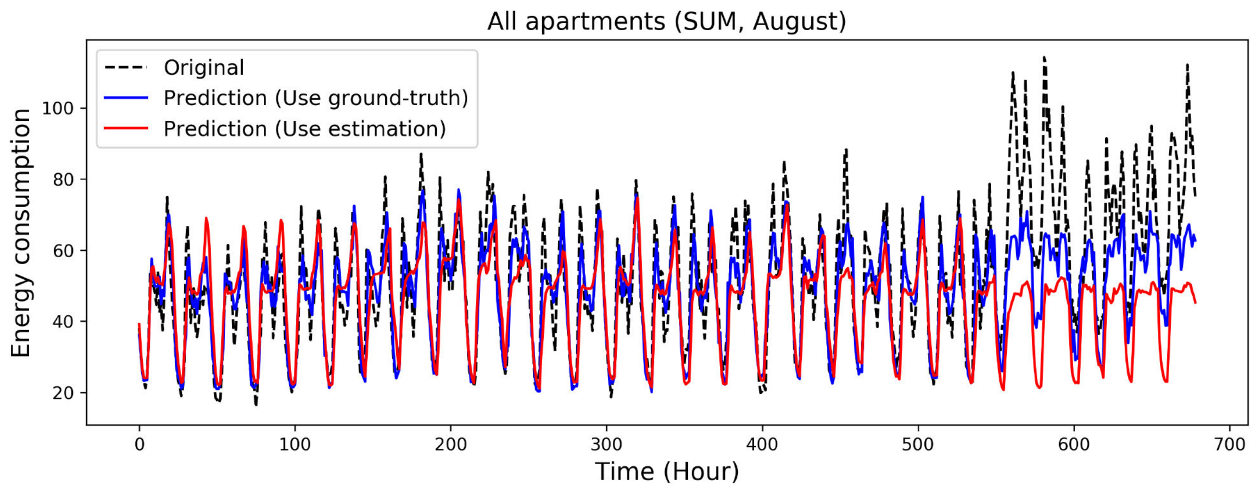

Fig. 7 *Forecasting results using predicted and actual values*

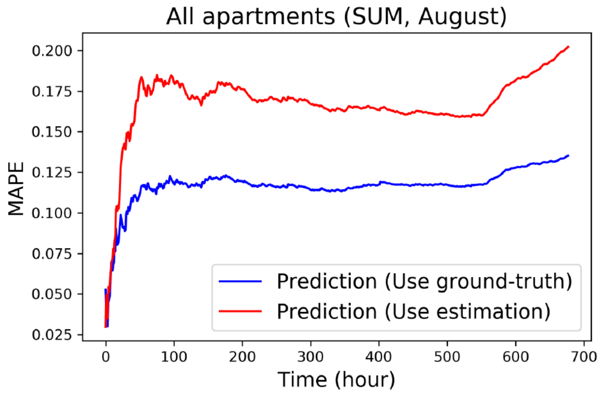

Fig. 8 *Forecasting MAPE using predicted and actual values*

from a point around 550 on the *x*-axis, the red line totally loses track of the original values. In order to understand the error quantitatively, we plot MAPE in Fig. 8 in red. We can see from the MAPE plot that the error increases as it goes farther into the future. Specifically, before 24 on the *x*-axis, the MAPE is at a low level, <10%. Thereafter, the MAPE rises a regional peak 18% at 52 on the *x*-axis. Subsequently, the MAPE declines a bit to 16% and maintains the value till 550 on the *x*-axis, the point from which the error increases sharply. Given the experimental results, we can infer that the model is suitable for forecasting in the day-ahead bidding task and day-ahead electricity scheduling.

4.4 Model retraining interval

For any data-driven model, it is necessary to keep the model up to date by retraining the model using fresh data. In particular, power consumption patterns are not fixed and the trained model would become obsolete over time; a fact which would result in lower forecasting accuracy. Thus, the timing for retraining is a crucial tuning parameter in real-world operation. Retraining is desired when degradation in prediction is noticed. This subsection is to empirically investigate appropriate model retraining interval and to find how long a trained *PowerNet* model can be used with acceptable accuracy. It also provides us with insight on how often *PowerNet* should be trained to capture the new power demand characteristics evolved with time.

This experiment is different from the previous experiment in Section 4.3. The experiment in Section 4.3 focuses on exploring the accuracy fluctuation caused by different lengths of forecasting periods and it forecasts the power demand for a period at once based on the data on hand at that moment. Differently, the experiment in this section uses actual data, which eliminates the error accumulation caused by forecasting using estimated feature values. We use the model trained in Section 4.3 and test it using the actual data in August.

The results are shown in Fig. 7 using the blue line. Generally, the prediction based on actual values (the blue line) is better than the prediction based on predicted values (the red line), which is reasonable and in line with the expectation. From the MAPE plot in Fig. 8, the same conclusion can be drawn. We can see that both ‘prediction (use ground-truth)’ and ‘prediction (use estimation)’ show almost same errors till 15 h on the *x*-axis. The latter digresses significantly afterwards. The ‘prediction (use ground-truth)’, i.e. the blue line maintains itself around 10% MAPE at 36 h and around 11% till 550 on the *x*-axis. At the very end, it reaches the largest error of around 13%. In practice, depending on the error tolerance of the prediction task, we can adjust our model by re-training the model with new data. For example, we can re-train the model every 36 h to capture the new characteristics of the data generated during the 36 h. Generally, the model can maintain an MAPE of around 11% for more than 3 weeks in the future prediction horizon.

5 PowerNet for anomaly detection

Anomaly detection is to identify patterns in data that do not conform to the defined normal behaviour [24]. Anomaly detection in smart grids focuses on the non-technical loss which is not caused by the intrinsic loss (technical loss, e.g. transmission loss) in a power system. Electricity theft is the most focused non-technical loss that causes anomalies. Data-driven anomaly detection can be done by modelling the normal consumption behaviour and defining a normal region. Any consumption that does not fall within the normal region is considered as an anomaly and it potentially indicates a problem in the smart grid. The forecasting results from *PowerNet* can be interpreted differently depending on the tasks, e.g. the power demand at some time in the future or the expected normal consumption at that time. In the latter sense, *PowerNet* can be used to define the normal consumption behaviour based on which further anomaly detection can be carried out.

Normally, for a consumer u, the reported consumption M_r should be roughly equal to the actual consumption M_u. However, an attacker may be able to manipulate M_r aiming at reducing the electricity bill by making $M_r < M_u$. We conduct a preliminary experiment to understand the performance of *PowerNet* to capture electricity theft. We here consider ‘forecasting using predicted values’ approach and evaluate the deviation from the prediction as the criteria for detection. In order to prevent manipulated consumption data from affecting the prediction, this is a reasonable design.

We artificially reduce the power consumption by different theft percentages in the test data to simulate different electricity theft scenarios. Fig. 9 shows the forecasting MAPE results under different theft percentages and Fig. 10 magnifies the first 30% of the *x*-axis in Fig. 9.

We can see from the magnified view (Fig. 10) that when theft percentage is small, the MAPE grows linearly as the percentage of

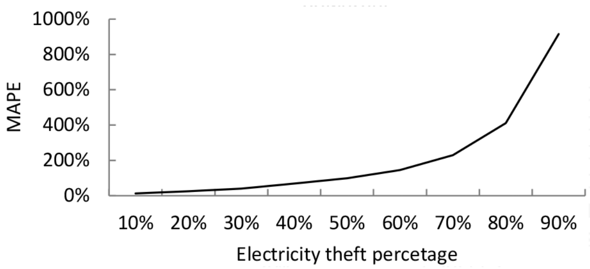

Fig. 9 *MAPE predictions over different electricity theft scenarios characterised by the theft percentage from 10 to 90%*

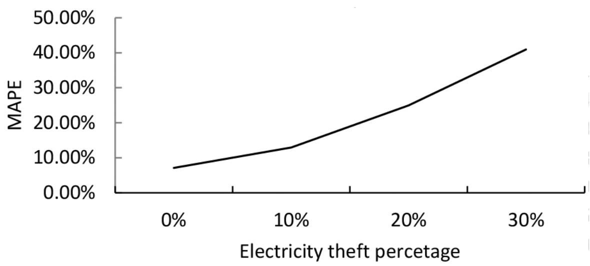

Fig. 10 *MAPE predictions over different electricity theft scenarios characterised by the theft percentage from 10 to 30%*

theft grows. However, from the experimental results in Fig. 9, we can see that the overall MAPE increases in an exponential manner. It means that the more the user steals, the larger the deviation between the predicted value M_p and the reported value M_r is. In other words, the more the user steals, the more obvious the deviation is. A reasonable threshold that would trigger an alarm can be inferred from the historical data as well as the tolerance of theft.

Anomaly detection can be deployed in both substation layer and individual consumer layer. We discuss how *PowerNet* can be utilised to detect such anomalies in both layers.

Anomaly detection in substation layer: At the substation level, there is a master meter which is a meter to measure the aggregated consumption of the whole supply region. The reading of master meter is denoted as M_s. So, we have $M_s = \sum_{i=1}^{N_c} M_u^i + \mathrm{TL}$, where N_c is the number of consumers in the supply region and TL is the technical loss. The substation can observe M_r^i which is the reported consumption of consumer i. We can obtain TL through $\mathrm{TL} = M_s - \sum_{i=1}^{N_c} M_u^i$. In normal case where $M_r^i = M_u^i$, we have $\mathrm{TL} = M_s - \sum_{i=1}^{N_c} M_r^i$. In order to detect the anomaly where $M_r^i \neq M_u^i$, we use *PowerNet* to model the indirectly observed TL_o. In the attack case where $M_r^i \neq M_u^i$, a deviation would be observed between the predicted TL_p and the observed TL_o. Hence, *PowerNet* is able to detect the anomaly under a substation supply region by constructing one model for one substation.

Anomaly detection in individual consumer layer: Anomaly detection at substation level can detect the anomaly but cannot pinpoint which consumer is suspicious. At the individual consumer level, with the help of the *PowerNet*, we can build a model for the consumer u based on his/her historical M_u. Once the attacker reduces his/her M_r to make $M_r \neq M_u$, we shall notice that there is a deviation between his/her M_r and M_p which is predicted by *PowerNet*. In this sense, anomaly detection at individual consumer layer can work as a complementary to anomaly detection in substation layer, which is able to locate the consumer who is suspiciously reporting false readings.

6 Related work

The existing works on power demand forecasting can be generally classified into two categories – classic statistical models and modern machine learning algorithms.

In terms of statistical models, time-series models have been used to capture the time-series characteristics of power demand, e.g. ARMA [25, 26], ARIMA [27–29]. Beside time-series models, Hong *et al.* [30] adopt multiple linear regression to model the hourly energy demand using seasonality (regarding year, week and day) and temperature information. Their results indicate that complex featuring of the same information results in a more accurate forecasting. Fan and Hyndman [31] use the semi-parametric additive model to explore the non-linear relationship between energy usage data and variables, i.e. calendar variables, consumption observations and temperatures in the short-term time period. Their model demonstrates sensitivity towards the temperature. In addition, conditional kernel density estimation is applied to the power demand forecasting area which performs well on the data with strong seasonality [32]. However, these models have limitations in incorporating heterogeneous features in a unified way. Differently, the design of *PowerNet* creates a neural network that can encode sequential features and single-value features simultaneously.

Regarding the machine-learning models, there are three models widely used for demand forecasting tasks, viz., DT [2, 33, 34], SVM [3, 35–37] and artificial neural network (ANN) [38–41]. DT is used to predict building energy demand levels [34] and analyse the electricity load level based on hourly observations of the electricity load and weather [33]. Later, Bansal *et al.* [2] use the boosted DT to model and forecast energy consumption so as to create personalised electricity plans for residential consumers based on the usage history. There are also works using SVR, the regression based on SVM and so on, to forecast power consumption in combination with other techniques such as fuzzy-rough feature selection [37], particle swarm optimisation algorithms [36] and chaotic artificial bee colony algorithm [35]. The SVR-based prediction has demonstrated good prediction results [3]. For the third model ANN, Gajowniczek and Zabkowski choose ANN because they believe that time-series analysis is not suitable for their work since they observe high volatility in the data [38]. Zufferey *et al.* [39] apply time delay neural network and find out that the individual consumer's consumption is harder to predict than an aggregation of multiple consumers. Recently, researchers take the advantage of LSTM to forecast building energy load using historical consumption data [40]. Historical load data and ambient temperature are utilised to build a prediction model based on ANN in [41]. Cheng *et al.* [42] further manage to feed the concatenation of historical data and influence features as a sequential input to the LSTM network. Since they only use the LSTM network, all data are treated as sequential data. Short-term demand forecasting using LSTM network based on historical load data and weather information has been proposed in [43]. In this paper, historical data are input to the LSTM layer and output of the LSTM layer is combined with weather data which is the output of a fully connected neural network. A similar study is performed in [44] where categorical features like time of the day, holiday flag and so on, are incorporated in addition to the weather data to enhance prediction accuracy for short-term load demand using LSTM network. Despite the extensive research carried out in power demand forecasting area, to the best of our knowledge, there is no such neural network architecture that takes consideration of heterogeneous features to the extent the *PowerNet* does.

Another stem of related work is the anomaly detection in smart grids for non-technical loss such as electricity theft. Bandim *et al.* [45] introduced an observer meter to observe the meter consumption of a set of users and further identified the tampered meter using the deterministic and statistical approach. Later, Krishna *et al.* [46] discussed the detection capability based on such extra meters on different attacks. Other than these, linear regression [47], cluster outlier [48, 49] and SVM [50, 51] are also used to detect the anomaly in smart girds. Biswas *et al.* [52] performed correlation analysis to pinpoint electricity thieves among a large pool of domestic consumers. Furthermore, Mashima and Cárdenas [26] evaluated the effectiveness of several anomaly detection models including the average detector, ARMA-GLR, and non-parametric statistics and local outlier factor. In this work, we discuss how *PowerNet* can be used in multiple anomaly detection layers.

7 Conclusion

In this paper, we propose *PowerNet*, a power demand forecasting model based on modern RNN and MLP network, which is capable of incorporating heterogeneous influencing factors in a unified way. It demonstrates improvement in prediction accuracy compared to four state-of-the-art approaches, viz., GBT, SVR, RF and GRU. Further, evaluation under different settings with the real-world dataset is carried out to better understand the model capability and crucial operational considerations in practice, mainly the length of the forecasting period and the model retraining interval. Based on our evaluation results, *PowerNet* shows advantages in terms of prediction accuracy when the prediction is made for individual and aggregated consumption of a group of households which is often challenging in practice. Finally, we briefly discuss the usability of *PowerNet* in anomaly detection task in the smart metering processes.

Our potential future work includes the evaluation with other smart meter datasets, such as datasets of commercial/industrial electricity consumers. Moreover, development and evaluation of anomaly detection scheme based on *PowerNet* under more sophisticated attacker models is also an interesting research direction.

8 Acknowledgments

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under the Energy Programme and administrated by the Energy Market Authority (EP Award No. NRF2017EWT-EP003-047 and NRF2014EWT-EIRP002-040). It is also in part supported by the National Natural Science Foundation of China (Grant No. 61932011), Guangdong Provincial Key R&D Plan (Grant No. 202020022911500032) and Guangdong Provincial Key R&D Plan (Grant No. 202020022911500032, 2019B1515120010).

9 References

[1] Siano, P.: 'Demand response and smart grids – a survey', *Renew. Sust. Energy Rev.*, 2014, **30**, pp. 461–478

[2] Bansal, A., Rompikuntla, S.K., Gopinadhan, J., *et al.*: 'Energy consumption forecasting for smart meters', arXiv preprint arXiv:151205979, 2015

[3] Yu, W., An, D., Griffith, D., *et al.*: 'Towards statistical modeling and machine learning based energy usage forecasting in smart grid', *ACM. SIGAPP. Appl. Comput. Rev.*, 2015, **15**, (1), pp. 6–16

[4] Lahouar, A., Slama, J.B.H.: 'Day-ahead load forecast using random forest and expert input selection', *Energy Convers. Manage.*, 2015, **103**, pp. 1040–1051

[5] Moon, J., Kim, Y., Son, M., *et al.*: 'Hybrid short-term load forecasting scheme using random forest and multilayer perceptron', *Energies*, 2018, **11**, (12), p. 3283

[6] Wang, Y., Liu, M., Bao, Z., *et al.*: 'Short-term load forecasting with multisource data using gated recurrent unit neural networks', *Energies*, 2018, **11**, (5), p. 1138

[7] Risinger, E.: 'Umass smart* dataset - 2017 release', 2017. Available at http://traces.cs.umass.edu/index.php/Smart/Smart

[8] Anderson, B., Lin, S., Newing, A., *et al.*: 'Electricity consumption and household characteristics: implications for census-taking in a smart metered future', *Comput. Environ. Urban Syst.*, 2017, **63**, pp. 58–67

[9] Mashima, D., Serikova, A., Cheng, Y., *et al.*: 'Towards quantitative evaluation of privacy protection schemes for electricity usage data sharing', *ICT Express*, 2018, **4**, (1), pp. 35–41

[10] Hochreiter, S., Schmidhuber, J.: 'Long short-term memory', *Neural Comput.*, 1997, **9**, (8), pp. 1735–1780

[11] Hornik, K., Stinchcombe, M., White, H.: 'Multilayer feedforward networks are universal approximators', *Neural Netw.*, 1989, **2**, (5), pp. 359–366

[12] Greff, K., Srivastava, R.K., Koutník, J., *et al.*: 'LSTM: a search space odyssey', *IEEE Trans. Neural Netw. Learn. Syst.*, 2017, **28**, pp. 2222–2232

[13] Gers, F.A., Schmidhuber, J., Cummins, F.: 'Learning to forget: continual prediction with LSTM', *Neural Comput.*, 2000, **12**, (10), pp. 2451–2471

[14] Mikolov, T., Karafiát, M., Burget, L., *et al.*: 'Recurrent neural network based language model'. INTERSPEECH, Makuhari, Japan, 2010, vol. 2, p. 3

[15] Sutskever, I., Vinyals, O., Le, Q.V.: 'Sequence to sequence learning with neural networks'. Advances in Neural Information Processing Systems, Montreal, Quebec, Canada, 2014, pp. 3104–3112

[16] Nair, V., Hinton, G.E.: 'Rectified linear units improve restricted Boltzmann machines'. Proc. of the 27th Int. Conf. on Machine Learning (ICML-10), Haifa, Israel, 2010, pp. 807–814

[17] Srivastava, N., Hinton, G.E., Krizhevsky, A., *et al.*: 'Dropout: a simple way to prevent neural networks from overfitting', *J. Mach. Learn. Res.*, 2014, **15**, (1), pp. 1929–1958

[18] Kingma, D., Ba, J.: 'Adam: a method for stochastic optimization', arXiv preprint arXiv:14126980, 2014

[19] Friedman, J.H.: 'Greedy function approximation: a gradient boosting machine', *Ann. Stat.*, 2001, **29**, pp. 1189–1232

[20] Friedman, J.H.: 'Stochastic gradient boosting', *Comput. Stat. Data Anal.*, 2002, **38**, (4), pp. 367–378

[21] Boser, B.E., Guyon, I.M., Vapnik, V.N.: 'A training algorithm for optimal margin classifiers'. Proc. of the Fifth Annual Workshop on Computational Learning Theory, Pittsburgh, Pennsylvania, USA, 1992, pp. 144–152

[22] Müller, K.R., Smola, A.J., Rätsch, G., *et al.*: 'Predicting time series with support vector machines'. Int. Conf. on Artificial Neural Networks, Lausanne, Switzerland, 1997, pp. 999–1004

[23] Conejo, A.J., Plazas, M.A., Espinola, R., *et al.*: 'Day-ahead electricity price forecasting using the wavelet transform and ARIMA models', *IEEE Trans. Power Syst.*, 2005, **20**, (2), pp. 1035–1042

[24] Chandola, V., Banerjee, A., Kumar, V.: 'Anomaly detection: a survey', *ACM Comput. Surv.*, 2009, **41**, (3), p. 15

[25] Gross, G., Galiana, F.D.: 'Short-term load forecasting', *Proc. IEEE*, 1987, **75**, (12), pp. 1558–1573

[26] Mashima, D., Cárdenas, A.A.: 'Evaluating electricity theft detectors in smart grid networks'. Int. Workshop on Recent Advances in Intrusion Detection, Amsterdam, Netherlands, 2012, pp. 210–229

[27] Alberg, D., Last, M.: 'Short-term load forecasting in smart meters with sliding window-based ARIMA algorithms'. Asian Conf. on Intelligent Information and Database Systems, Kanazawa, Japan, 2017, pp. 299–307

[28] Cho, M., Hwang, J., Chen, C.: 'Customer short term load forecasting by using ARIMA transfer function model'. 1995 Int. Conf. on Energy Management and Power Delivery, 1995. Proc. of EMPD'95, Singapore, 1995, vol. 1, pp. 317–322

[29] Nepal, B., Yamaha, M., Yokoe, A., *et al.*: 'Electricity load forecasting using clustering and ARIMA model for energy management in buildings', *Jpn. Archit. Rev.*, 2020, **3**, pp. 62–76

[30] Hong, T., Gui, M., Baran, M.E., *et al.*: 'Modeling and forecasting hourly electric load by multiple linear regression with interactions'. IEEE Power and Energy Society General Meeting, Minneapolis, Minnesota, USA, 2010, pp. 1–8

[31] Fan, S., Hyndman, R.J.: 'Short-term load forecasting based on a semi-parametric additive model', *IEEE Trans. Power Syst.*, 2012, **27**, (1), pp. 134–141

[32] Arora, S., Taylor, J.W.: 'Forecasting electricity smart meter data using conditional kernel density estimation', *Omega*, 2016, **59**, pp. 47–59

[33] Gładysz, B., Kuchta, D.: 'Application of regression trees in the analysis of electricity load', *Bad. Oper. Decyzje*, 2008, **4**, pp. 19–28

[34] Yu, Z., Haghighat, F., Fung, B.C., *et al.*: 'A decision tree method for building energy demand modeling', *Energy Build.*, 2010, **42**, (10), pp. 1637–1646

[35] Hong, W.C.: 'Electric load forecasting by seasonal recurrent SVR (support vector regression) with chaotic artificial bee colony algorithm', *Energy*, 2011, **36**, (9), pp. 5568–5578

[36] Qiu, Z.: 'Electricity consumption prediction based on data mining techniques with particle swarm optimization', *Int. J. Database Theory Appl.*, 2013, **6**, (5), pp. 153–164

[37] Son, H., Kim, C.: 'Forecasting short-term electricity demand in residential sector based on support vector regression and fuzzy-rough feature selection with particle swarm optimization', *Procedia Eng.*, 2015, **118**, pp. 1162–1168

[38] Gajowniczek, K., Ząbkowski, T.: 'Short term electricity forecasting using individual smart meter data', *Procedia Comput. Sci.*, 2014, **35**, pp. 589–597

[39] Zufferey, T., Ulbig, A., Koch, S., *et al.*: 'Forecasting of smart meter time series based on neural networks'. Workshop Data Analytics for Renewable Energy Integration (DARE), European Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD), Riva del Guarda, 2016, pp. 19–23

[40] Marino, D.L., Amarasinghe, K., Manic, M.: 'Building energy load forecasting using deep neural networks'. 42nd Annual Conf. of the IEEE Industrial Electronics Society, IECON 2016, Florence, Italy, 2016, pp. 7046–7051

[41] Fernandes, K.C., Sardinha, R., Rebelo, S., *et al.*: 'Electric load analysis and forecasting using artificial neural networks'. 2019 3rd Int. Conf. on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2019, pp. 1274–1278

[42] Cheng, Y., Xu, C., Mashima, D., *et al.*: 'PowerLSTM: power demand forecasting using long short-term memory neural network'. Int. Conf. on Advanced Data Mining and Applications, Singapore, 2017, pp. 727–740

[43] Kwon, B.-S., Park, R.-J., Song, K.-B.: 'Short-term load forecasting based on deep neural networks using LSTM layer', *J. Electr. Eng. Technol.*, 2020, **15**, pp. 1501–1509

[44] Hossain, M.S., Mahmood, H.: 'Short-term load forecasting using an LSTM neural network'. 2020 IEEE Power and Energy Conf. at Illinois (PECI), Champaign, IL, USA, 2020, pp. 1–6

[45] Bandim, C., Alves, J., Pinto, A., *et al.*: 'Identification of energy theft and tampered meters using a central observer meter: a mathematical approach'. 2003 IEEE PES Transmission and Distribution Conf. and Exposition, Dallas, TX, 2003, vol. 1, pp. 163–168

[46] Krishna, V.B., Lee, K., Weaver, G.A., *et al.*: 'F-DETA: a framework for detecting electricity theft attacks in smart grids'. 2016 46th Annual IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN), Toulouse, France, 2016, pp. 407–418

[47] Liu, X., Nielsen, P.S.: 'Regression-based online anomaly detection for smart grid data', arXiv preprint arXiv:160605781, 2016

[48] Menon, D.M., Radhika, N.: 'Anomaly detection in smart grid traffic data for home area network'. Int. Conf. on Circuit, Power and Computing Technologies (ICCPCT), 2016, Nagercoil, India, 2016, pp. 1–4

[49] Chen, C., Cook, D.J.: 'Energy outlier detection in smart environments', *Artif. Intell. Smarter Living*, 2011, **11**, p.07

[50] Nagi, J., Yap, K.S., Tiong, S.K., *et al.*: 'Nontechnical loss detection for metered customers in power utility using support vector machines', *IEEE Trans. Power Deliv.*, 2010, **25**, (2), pp. 1162–1171

[51] Jokar, P., Arianpoo, N., Leung, V.C.: 'Electricity theft detection in ami using customers' consumption patterns', *IEEE Trans. Smart Grid*, 2016, **7**, (1), pp. 216–226

[52] Biswas, P.P., Cai, H., Zhou, B., *et al.*: 'Electricity theft pinpointing through correlation analysis of master and individual meter readings', *IEEE Trans. Smart Grid*, 2020, **11**, pp. 3031–3042