Predicting plug loads with occupant count data through a deep learning approach

Zhe Wang, Tianzhen Hong*, Mary Ann Piette

Building Technology and Urban Systems Division
Lawrence Berkeley National Laboratory, One Cyclotron Road, Berkeley, CA 94720, USA
*Corresponding author: thong@lbl.gov, (+1) 510-486-7082

ABSTRACT

Predictive control has gained increasing attention for its ability to reduce energy consumption and improve occupant comfort in buildings. The plug loads prediction is a key component for the predictive building controls, as plug loads is a major source of internal heat gains in buildings. This study proposed a novel method to apply the Long-Short-Term-Memory (LSTM) Network, a special form of Recurrent Neural Network, to predict plug loads. The occupant count and the time have been confirmed to drive the plug load profile and thus selected as the features for the plug load prediction. The LSTM network was trained and tested with ground truth occupant count data collected from a real office building in Berkeley, California. Results from the LSTM network markedly improve the prediction accuracy compared with traditional linear regression methods and the classical Artificial Neural Network. 95% of 1-hour predictions from LSTM network are within ±1 kW of the actual plug loads, given the average plug loads during the office hour is 8.6 kW. The CV(RMSE) of the predicted plug load is 11% for the next hour, and 20% for the next eight hours. Lastly, we compared four prediction approaches with the office building we monitored: LSTM vs. ARIMA, with occupant counts vs. without occupant counts. It was found, the prediction error of the LSTM approach is around 4% less than the ARIMA approach. Using occupant counts as an exogenous input could further reduce the prediction error by 5%-6%. The findings of this paper could shed light on the plug load prediction for building control optimizations such as model-predictive control.

Keywords:

Plug loads; Prediction; Predictive control; Long Short Term Memory Network; Occupant count; Deep learning

1. Introduction

Commercial buildings are energy intensive, consuming 18% of the national energy consumption in the United States [1]. In commercial buildings, the plug load is one of the major energy consumers, which consumes 40% of the total building energy use in an office building in San Jose, California [2] and 60% in another office building in Berkeley, California [3]. On average, 20% of the commercial building energy consumption in the United States is consumed by the plug load [4], and this proportion is even higher in the United Kingdom [5]. The plug load plays a significant role in building energy efficiency, not only because of the high proportion of electricity it directly consumes but also because those consumed energies would be converted into heat and then dissipated to the indoor environment. As a major source of internal heat gains, the plug load would substantially affect the operation and energy efficiency of HVAC systems.

1.1 Predictive controls

The key point of predictive control is to optimize the control actions based on the prediction about future demands, disturbances, and inputs. A typical application of predictive control is the optimization of chilled water thermal storage systems [6] and the ice-storage systems [7]. By predicting the thermal demands and the utility price in the coming hours, the operation strategy of the thermal storage system could be optimized for minimizing operational costs. Since the prediction of future demands and disturbances is required for its optimization, the performance of predictive control is very sensitive to and highly dependent on the accuracy of predictions.

Model Predictive Control (MPC) is a representative of predictive controllers [8]. Compared with conventional building controllers which only focus on the current state of building and HVAC systems, MPC can include the predicted states and events into the loop, and accordingly outperforms alternative methods of multivariable control in terms of comfort and building efficiency [9]. Because of its potentially high performance, MPC starts to attract the attention of building energy efficiency researchers and operators, and has been applied to control HVAC systems [10], [11], [12], [13], [14].

As illustrated in the terminology, prediction is the prerequisite for predictive control. Three types of predicted information are needed to optimize building energy efficiency. First, the weather condition [15], which would influence the building thermal load and the efficiency of active or passive HVAC systems [16]; second, the occupant demands, for instance, predicting the fresh air demand by the occupant count forecasting; third, the heat disturbances, including the internal heat gains from the equipment (plug loads, lights) and the occupants. Sometimes, the utility rate prediction is needed to minimize utility bills. For either purpose of minimizing the energy consumption or the utility bills, predicting plug loads is needed in the predictive control. Considering the substantial amount of energy consumed by plug-in equipment [1], [3], [4], plug load is a major source of internal heat gains in office settings [17], [18]. Therefore, predicting plug loads is crucial in HVAC predictive controls.

1.2 Predicting plug loads in office buildings

Plug loads include the energy used by office equipment such as computers, printers, telephones and personal devices such as cellphone chargers and personal devices such as fans and portable heaters. It is reasonable to assume that the office workspace plug loads are linked to occupant count [19].

Kim and Srebric (2017) used a linear equation to regress the plug loads with the occupant count in four office and campus buildings in Philadelphia [1]. The correlation coefficients between plug loads and occupant counts could achieve 0.68 ~ 0.78, which are higher than the correlation coefficients between total building electricity consumption and occupant counts (in the range of 0.50 ~ 0.74). This higher correlation demonstrated that plug loads are more likely to be accurately predicted by the occupant counts, compared with other components of building energy consumption (e.g., HVAC). In addition to the linear regression, Mahdavi et al. applied Weibull distribution to regress plug loads with occupant counts and then compared its performance with linear regression [20]. It was found that the linear regression performs better in terms of predicting annual plug load consumption, while the Weibull approach performs better in terms of predicting the peak plug load and fitting the plug load distribution due to its ability to simulate the stochastic behavior of plug loads.

In addition to the occupant counts, another factor could be used to predict plug load is the time. Gunay et al. defined five different time periods: occupancy, intermediate breaks, weekday evenings, weekends, and vacations to fit plug loads [17]. Gandhi and Brager defined two time periods, working day and non-working day, to develop models to simulate plug load profiles [19].

Another widely used time-series prediction, such as building plug loads, is through the AutoRegressive Moving Average (ARMA) or AutoRegressive Integrated Moving Average (ARIMA) approach [21]. ARMA is a regression-based approach, which predicts a time-series variable with the autoregressive part and the moving average part of the historical data. As shown in Equation 1, the autoregressive (AR) part predicts the variable based on its own lagged value (X_{t-1}, …

X_{t-p}), while the moving average (MA) part predicts the variable based on previous regression error (ε_{t-1}, … ε_{t-q}). The integrated part (I) in the ARIMA was introduced to eliminate the non-stationarity [22]. Amjady utilized ARIMA to predict the 24-hour ahead building electricity usage and the daily peak load in Iran [23]. Espinoza et al. combined ARIMA and clustering technique to predict the residual electricity load and achieved a 90% R-square value in more than 92% cases he studied [24].

$$X_t = \alpha_1 X_{t-1} + \cdots + \alpha_p X_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \qquad \text{Equation 1}$$

It is also possible to add an exogenous variable in ARIMA models to improve prediction accuracy, as the time-series variable of interest might not only demonstrate seasonal, trend and residual behavior which could be captured by ARIMA, but also be influenced by exogenous variables. Yun et al. predict the 1-hour ahead building thermal load by developing an AR model which uses temperature as an exogenous variable [25]. Similarly, Newsham and Birt used occupancy data to improve the ARIMA-based electricity forecast in office building [26].

In addition to the ARIMA-based approach, different types of neural network-based approach have been applied to predict building electricity usage or thermal load. Kouhi and Keynia applied cascade Neural Network method to predict the electricity consumption of New York [27]. Guan et al. proposed a wavelet neural network method with data pre-filtering to facilitate short-term (5 minutes to 1 hour) electricity usage forecasting, which was tested in the New England region [28]. Similar to the regression-based approach discussed above, exogenous variables could be added in the neural network-based methods to improve prediction accuracy. Kusiak et al. developed a neural network ensemble with five multi-layer perceptrons using weather data as exogenous variables to predict steam consumption for heating [29].

1.3 Problem definition and research objectives

As summarized in Table 1, there are two types of problem to study the hourly plug load profile: the ***modeling*** problem and the ***prediction*** problem. ***Modeling*** problem refers to ***model*** the plug loads at the timestamp *t* with some features (for instance occupant counts, and day of week) at the timestamp *t*. ***Prediction*** problem refers to use the information currently available (at timestamps *t, t-1, t-2* …) to ***predict*** the plug load in the future (at timestamps *t+1, t+2* …).

Table 1 Distinctions between the modeling problem and the prediction problem

	Modeling	Prediction
Definition	Use the occupant counts to regress the plug loads at the same timestamp	Use the information we've already known to predict the plug loads we do not know yet
Inputs	Occ_t, Occ_{t-1}, Occ_{t-2} …	Occ_t, Occ_{t-1}, Occ_{t-2} … P_t, P_{t-1}, P_{t-2} …
Outputs	P_t	P_{t+1}, P_{t+2} …
Application	• To generate more realistic plug load schedule profiles for building simulation [30], [31] • Post-Occupancy Evaluation, energy audit to understand why building consumes more	• Predictive control: plug load is a key component of internal heat gains [34]. Therefore, the prediction of plug load could be used to optimize HVAC control, such as pre-cooling to avoid temperature

	energy than the designed value [32], [33]	over-shooting.

To implement the predictive control, ***modeling*** the plug loads at current timestamp is not enough, the ***prediction*** is needed to formulate the optimization problem for the search of optimized chilled water temperature, supply air temperature, etc. Actually, because of the tightening regulation on building insulation [35] and increasing usage of appliances [36], plug loads account for a higher proportion of building thermal load. Plug load prediction becomes increasingly important for predictive control of HVAC system [37].

From the literature review, it could be concluded: first, occupant count is an important feature to predict building plug load [1], [20] and electricity usage [26], which should not be ignored in plug load prediction. Second, we could find plenty of studies on building electricity usage prediction [23], [24], [26]; while we have not found plug load prediction research. Existing plug load studies (e.g., [1], [20]) belong to the modeling problem rather than the prediction problem. To fill in this research gap, this study proposes a model to predict the plug loads, which could facilitate and be directly plugged into predictive controllers. The key research questions of this study are:

- Which factors markedly influence the plug loads in office buildings? which features should be selected to predict the plug loads?
- Which algorithm is most suitable for the plug loads prediction and could reduce prediction errors?

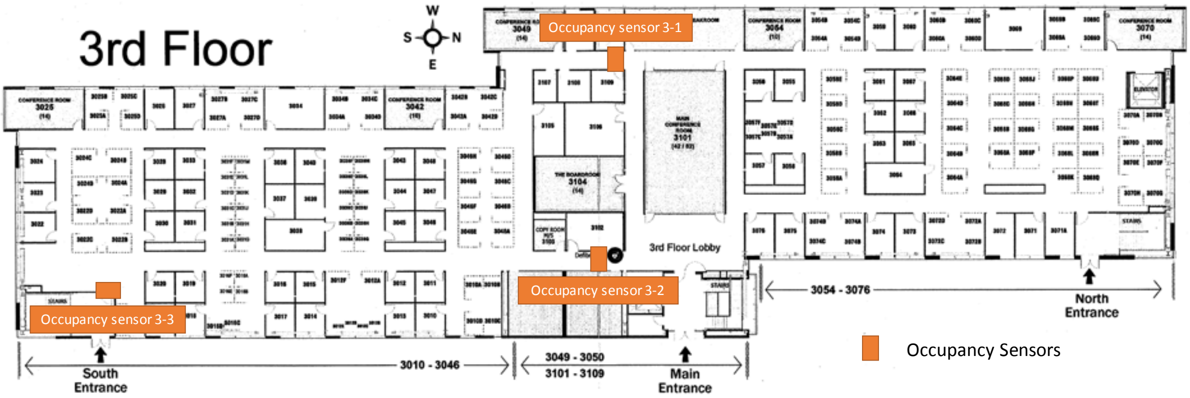
The remaining of this paper starts with the introduction of the data collection, as data is the fuel for a data-driven approach. Section 2 presents the test building (Section 2.1), sensors (Section 2.2), and the collected raw data (Section 2.3). The first research question is answered in Section 3 by examining how the plug load is correlated with occupant count (Section 3.1) and the time (Section 3.2). The second research question is explored in Section 4, by applying a novel approach - LSTM Network - to predict plug loads. We explain why LSTM is suitable for plug loads prediction (Section 4.1), introduce how the model is set up (Section 4.2), and finally compare the predicted results with the ground truth (Section 4.3). Two key issues, data availability (Section 5.1) and computational efficiency (Section 5.2) of implementing the LSTM network method in real buildings for predictive control are discussed in Section 5. Conclusions are drawn in Section 6.

The major contribution of this study is using a novel method to predict plug loads in office settings with the Long-Short–Term-Memory machine learning approach. This method has been tested by the ground truth data collected from a real office building, and could be directly integrated into building predictive controls.

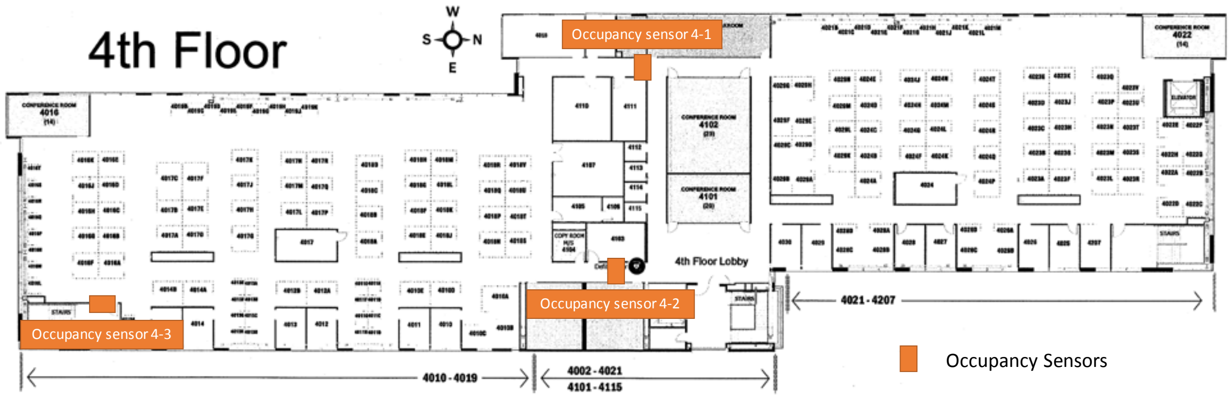
2. Data collection

2.1 Test building

An office building located in Berkeley, California was selected as the testbed for this study. The data were collected on the third and fourth floors of the test building, which both serve as office areas and have similar layouts. The floor plans of the third and fourth floors are presented in Figure 1.



(a) Floor map of the third floor



(b) Floor map of the fourth floor

Figure 1 Floor plans of the test building

The South half of the third and fourth floors are served by the same air handling unit. Meanwhile, the plug loads of the South half of the third and fourth floors are measured in one electrical circuit. We could not further separate the plug loads consumed by the South half of the third floor from the South half of the fourth floor. Actually, it is not necessary to further separate for HVAC control, since these two spaces are grouped into the same thermal zone and share the same cooling/heating source. Considering the status quo of the test building's HVAC system and energy management platform, this study's prediction problem is defined as *"to use the occupant count of the South part of the third and fourth floors to predict the plug loads of the same area."*

2.2 Occupancy detection

Researchers have proposed multiple methods to detect occupant count, such as CO_2 concentration based models [38], [39], Radio-Frequency Identification detection (RFID) systems [40], Wi-Fi connection based detection [41], [42], and camera-based sensors [43], [44]. Yang et al. (2016) summarized the advantages and disadvantages of the currently available occupancy detection approaches [45].

Figure 2 Camera-based sensor deployed in this study (https://www.trafsys.com/)

In this research, we chose camera-based sensors to detect occupant counts after considering issues including the detection accuracy, deployment cost and feasibility. We deployed occupant counters (Figure 2) manufactured by the TRAF-SYS company at the six entrances/exits of the investigated space, as illustrated in Figure 1. By counting and integrating how many occupants enter and leave the space through the entrances/exits, we calculated the number of occupants in the investigated space.

2.3 Raw data collected

Figure 3 illustrated the raw data we collected from the test building. The plug load data was collected from the building's sub-metering system; while the occupant count data was collected from camera-based sensors. The occupancy data was collected at the interval of one minute, and the plug load power was collected every 15 minutes. The missing rate of plug load power in July and August is high, undermining the data quality. Finally, we summed up the occupant counts of the South half of the third and fourth floors to ensure that the occupant counts and plug loads are spatially consistent.

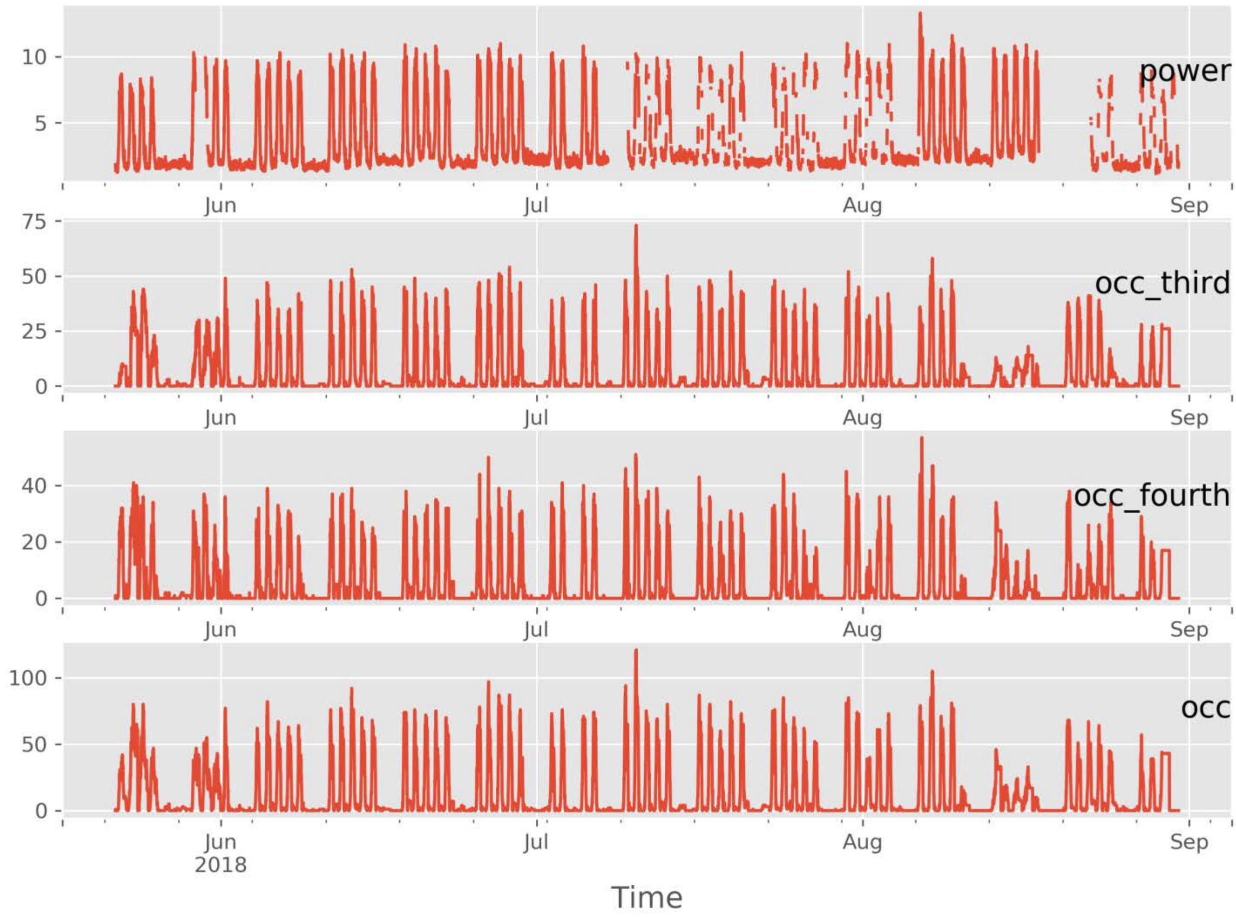


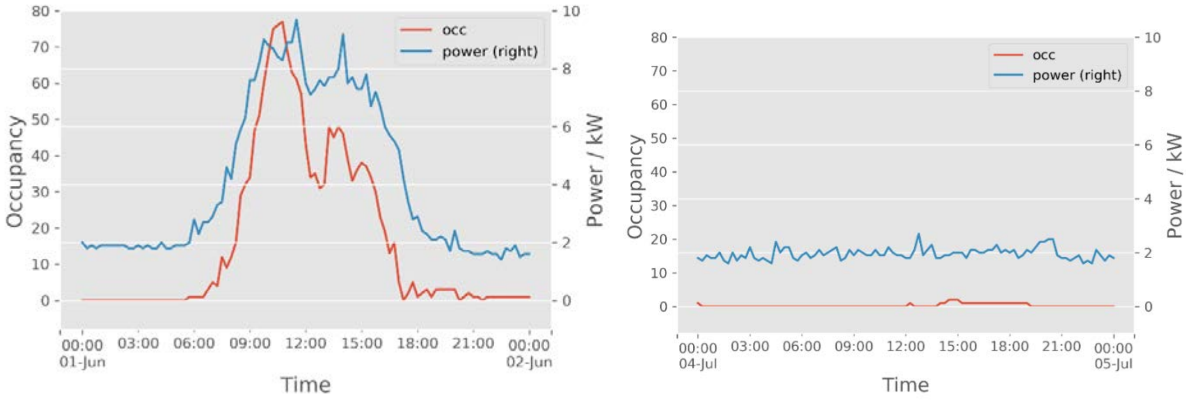
Figure 3 Raw data collected

3. Feature selections

The first research question, *"which factors markedly influence the plug loads in office buildings? which features should be selected to predict the plug loads?"* would be answered in this section. The relation of plug loads with occupant counts and time are explored by linear regression and Pearson Correlation Matrix separately.

3.1 Correlation with occupant count

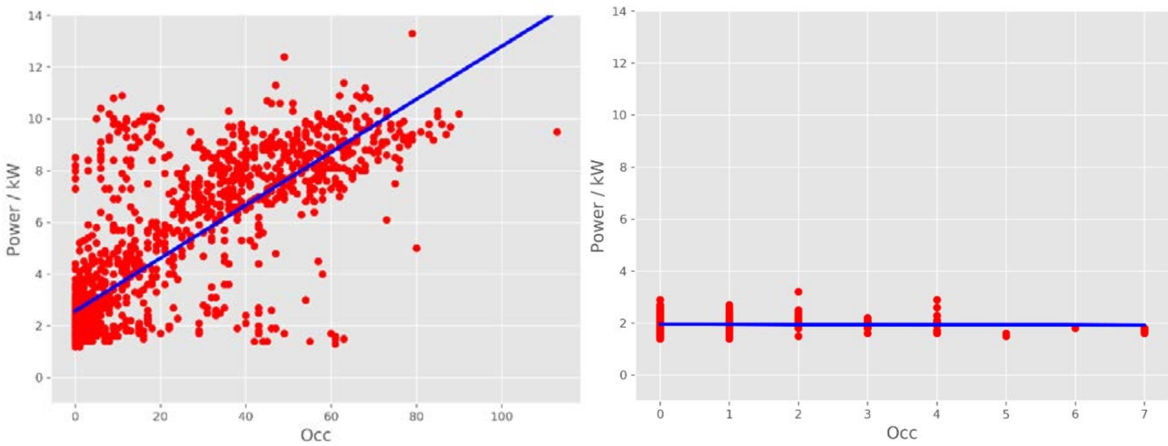
The occupant count and plug loads of a typical working day and a typical non-working day were shown in Figure 4. Different behaviors could be clearly identified during the working and non-working days. Therefore, we explore the correlation between plug loads and occupant counts during working days and non-working days separately.



(a) Working day (b) non-working day

Figure 4 Occupant count and plug loads on typical days

Figure 5 presents the linear regression results for working days and non-working days. The linear relation between plug load power and occupant count could be observed for working days but not for non-working days. The plug loads are almost a constant during non-working days, representing the energy consumed by office equipment that is always on and has nothing to do with occupancy.



(a) Working days (b) Non-working days

Figure 5 Fitting plug loads with occupant counts using linear regression

Root Mean Square Error (Equation 2) and R-square value are two commonly used metrics for regression model evaluation. Considering that Root Mean Square Error (RMSE) would bias the model comparison if the scales of two problems were different, we normalized the RMSE to calculate the Coefficient of Variation of the Root Mean Square Error, a.k.a. CV(RMSE). As shown in Equation 3, CV(RMSE) is defined by dividing the RMSE by the average of the measured value.

$$\mathrm{RSME} = \sqrt{\frac{\sum_{1}^{n}(\widehat{y_n} - y_n)^2}{n}} \qquad \text{Equation 2}$$

$$\mathrm{CV(RMSE)} = \mathrm{RSME}/\bar{y} \qquad \text{Equation 3}$$

Where, n is the sample size, y_n is the measured value, $\widehat{y_n}$ is the predicted value.

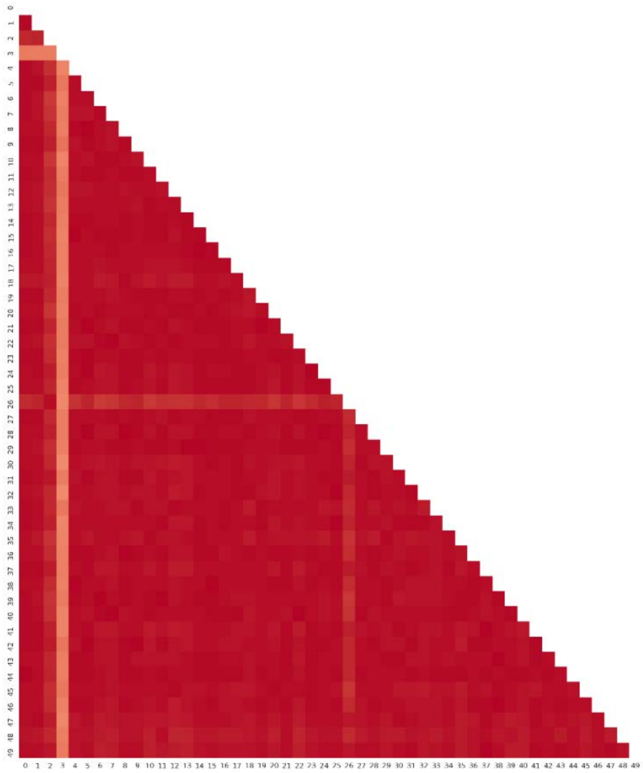
Table 2 Linear regression result of this study and previous studies

	This study		Kim and Srebric (2017) [1]	Mahdavi et al. (2016) [20]
	Working day	Non-working day		
Peak occupant counts	80	~0	100 ~ 300	8
Root Mean Square Error (kW)	1.72	0.27		0.13 ~ 0.16
CV(RMSE)	39%	14%		12.0% ~ 14.4%
R-Squared	0.65	0.01	0.68 ~ 0.78	

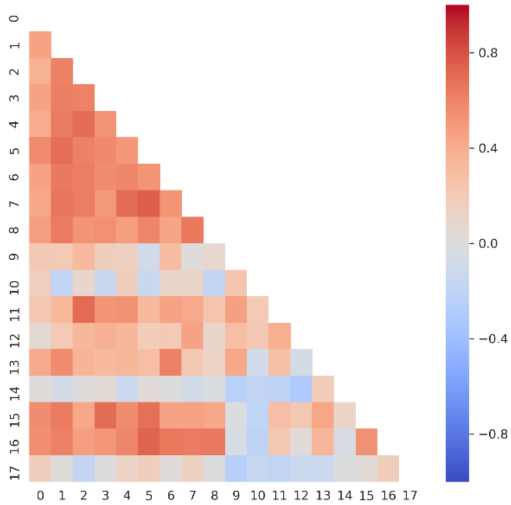
Table 2 compared the performance of our model with two previous studies on the same topic. The peak occupant count was used as a proxy for the scale of problems. The fitting goodness of this model is similar to the study [1], but a little bit worse than the study [20] in terms of the CV(RMSE), which might be because our model is built for a space with 80 occupants, while the model proposed by Mahdavi et al. (2016) [20] is used for a space with 8 occupants, which might have less variability. The high R-squared value (0.65) indicates that the occupant count is an important feature for plug load prediction. However, the relatively large regression error (39%) confirms the motivation to develop a more accurate plug load prediction model.

3.2 Correlations between daily plug load

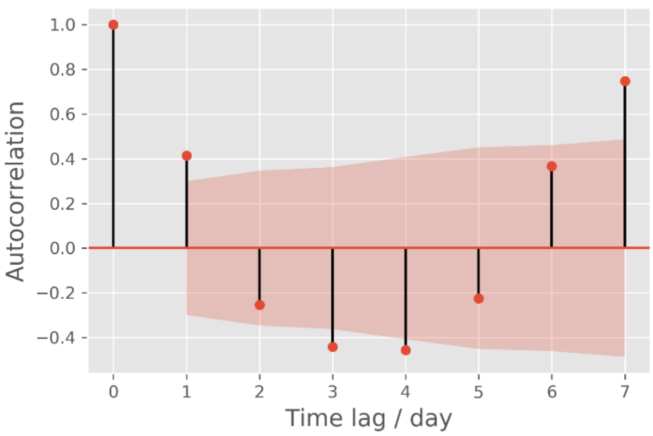
The Pearson Correlation Matrix was used to explore the influence of time on plug loads. We could consider the daily plug load profile as a 24-dimension vector, with the *kth* element representing the plug load at the *kth* hour. By checking the similarity of the plug load profile in two days, we could examine the correlation between plug loads and time. If the time has a strong influence on the plug loads, then the plug loads on different days would have similar patterns, i.e., the daily plug load vectors should be similar to each other. The Pearson Correlation Coefficient (PCC) is a commonly used metrics to quantify the similarity between vectors and accordingly used in this study. As shown in Figure 6(a), the Pearson Correlation Coefficients of the daily plug loads are high among the working day's group. The PCC of any two working days is higher than 0.4, proving that the plug loads have similar load shapes among working days. The plug loads of the previous working day might be a good predictor for the plug load of the next days. Contrarily, the plug load patterns among non-working days are not similar to each other, as illustrated in Figure 6(b).



(a) Pearson Correlation: Working day



(b) Pearson Correlation: Non-working days



(c) Autocorrelation figure

Figure 6 Correlations between daily plug loads

Another useful tool to investigate the correlations of time-series data is the autocorrelation figure, which plots the Pearson's correlation coefficient between P_t and P_{t-x}. As shown in Figure 6(c), the PCC between P_t and P_{t-7} is as high as 0.8, indicating a significant weekly periodic behavior: the same day of week has similar daily plug load.

In summary, we used linear regression to explore the relation of plug loads with the occupant count and then used Pearson Correlation Matrix to explore the relation of plug loads with the time. The R-square value for the linear regression between plug loads and occupant count is 0.65, which is relatively high for linear regression. However, the prediction error reaches 39%, which might not be considered acceptable for the control purpose. The correlation matrix reveals a similar pattern of plug loads among working days.

4. LSTM Network approach

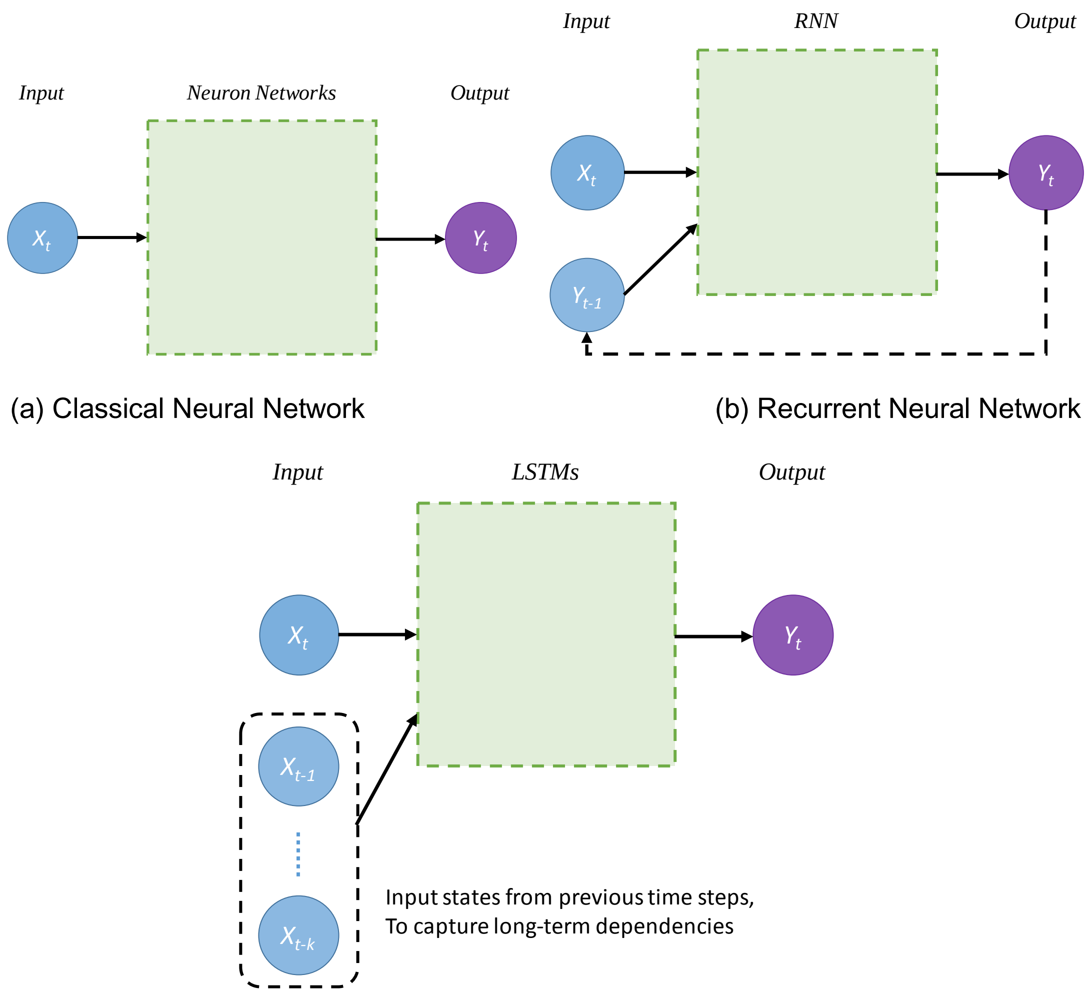
In this section, we aim to answer the second research question, i.e. *"which algorithm is most suitable for the plug load prediction and could reduce prediction errors?"*, by applying the LSTM Network to predict plug loads. LSTM is a special form of Artificial Neural Network (ANN). We will start by explaining why LSTM is selected and suitable to predict plug loads.

4.1 From ANN to LSTMs

The **Artificial Neural Network** is an extensively utilized machine learning algorithm that is inspired by the biological neural networks and functions by mimicking the human brains. ANN utilizes the input X_t at the timestamp t to predict the output h_t at the same timestamp, as shown in Figure 7(a). No information is exchanged between different timestamps.

However, in time-series data, the output h_t is not only influenced by the input at the current timestamp X_t, but also affected by what happened in the previous timestamp. To capture this time dependency, the **Recurrent Neural Network (RNN)** has been proposed, taking both the input at the current timestamp t and the state of the previous timestamp $t-1$ into consideration to predict the output h_t, as illustrated in Figure 7(b). Because of the ability to capture the time-dependencies of time-series data, RNN has been applied to a variety of problems: speech recognition, translation, the prediction of occupancy [46], etc.

RNN is powerful but not enough when the prediction tasks have long-term dependencies, which means the output of timestamp *t*, was not only influenced by the state of the previous timestamp *t-1*, but also by the state of *k* timestamps ago *t-k*. With the gap *k* growing, RNN struggles to learn and connect the information. To address this issue, the **LSTM** Network has been proposed [47]. As illustrated in Figure 7(c), all the state of the previous timestamp could be connected, no matter how long the time gap *k* is. However, to avoid the overwhelming of information, which consumes memory, and to reduce the model complexity, which slows down the model training process, a “forget gate layer” has been added to the model to decide which state is irrelevant and should be thrown away. Through such a mechanism, relevant information would be used for the prediction task, regardless of the length of the time gap.



(a) Classical Neural Network

(b) Recurrent Neural Network

(c) Long Short Term Memory Network

Figure 7 Systematic schemes of different neural networks

The plug loads prediction has the long-term dependencies. The high correlation between daily plug load vectors, as illustrated in Figure 6, indicates that the plug load 24 hours ago contains valuable information to infer the current plug load. If P_{t-1} and P_{t-24} were found to be the most relevant and useful information to predict P_t, then the information between the timestamp *t-23* and *t-2* would be “forgotten” to simplify and speed up the model. LSTMs’ capability to automatically identify, memorize, and use such long-term dependency information makes LSTMs a suitable tool to predict plug loads.

4.2 Model setup

Data cleaning

The raw data was collected at the 15-minute interval for the plug load power and at the 1-minute interval for the occupant count. We resampled and averaged the data at 1-hour interval for both the plug load and occupant count. The data is resampled for three reasons. First, the plug loads would not be influenced by short-term occupancy variability, e.g., the inhabitants would not turn off their desktop computers when they temporarily leave for the restroom or short meetings. By using the average hourly value, short-term variability could be removed. Second, the plug load prediction is for the control-oriented heat balance. The heat balance in buildings does not require a very high temporal resolution due to the effect of building thermal mass. Third, up-sampling the data could help to relieve the problem of the high missing rate observed in Figure 3. If one measurement is missing, we could use the average of the other three measurements as the plug load of that hour.

After the resampling, the whole month of June is free of missing data, which will be used to train and test our LSTM network model. We further divided June into two halves. The first two weeks is used to train the model, while the second two weeks is used as the test dataset.

Feature selection

Considering the significant different plug load shape on working and non-working days, we added one more feature as a proxy for working/non-working days. We chose the *day-of-week* as the proxy variable: 0 for Monday, 1 for Tuesday, … 6 for Sunday. Such a coding could inform the predictor not only the working/non-working day, but also the hour of the day. Since the predictor could infer the time of next hour by counting how many previous hours have the same *day-of-week* value. For instance, if the *day-of-week* of the previous 8 hours were the same, then the next hour would be 9 AM.

The next question is how many hours of the previous data should be input into the predictor for prediction. Considering the periodical behavior of plug loads, we chose to use the data of the previous 24 hours as the input to predict the plug loads in the coming eight hours. After the feature selection, we normalized the values of the selected features into the range of 0 to 1, as normalization is generally required by deep learning to improve prediction performance.

Hyper-parameter tuning

Table 3 presents the hyper parameters we chose in this model. We set the number of the hidden layer as one, and the number of neurons in the hidden layer as 50, since there is no evidence to support the existence of complex non-linear behaviors in the plug load prediction problem. Our hyper-parameter tuning process found adding extra hidden layers or increasing the number of neuron units would not increase the prediction accuracy, but lead to longer training time. As the training dataset is not too large, it is not necessary to use the Stochastic Gradient Descent or mini-batch Gradient Descent to speed up the training process. Therefore we passed the entire training set in a single iterative batch. As for the number of epochs, we plotted the Train and Test Loss against the number of epochs in Figure 8, and set the number of epochs as 300 for two reasons. First, the Test Loss becomes stable and not further decreases after 300 epochs. Second, the Train and Test Loss starts to diverge after 300 epochs, indicating a risk of overfitting if the number of epoch is set too large.

Table 3 Hyper-parameter selection

	Value selected
Number of hidden layers	1
Number of neurons in the hidden layer	50
Batch size	336 (the entire training set)
Number of Epochs	300

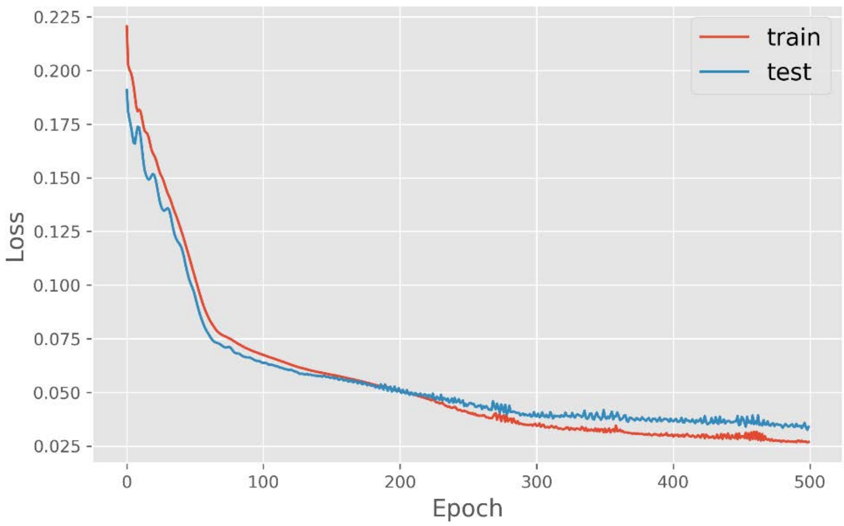


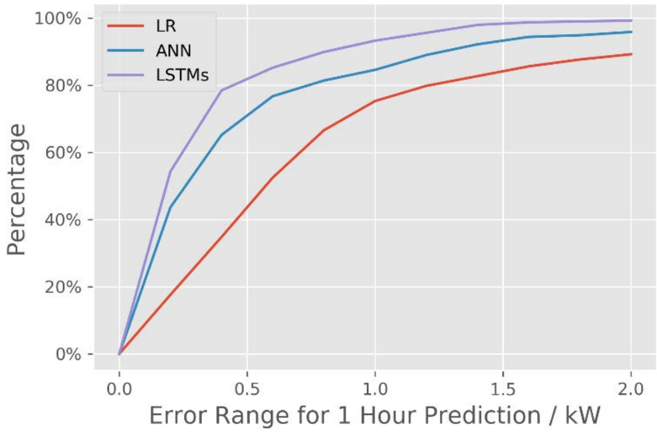
Figure 8 the Train and Test Loss

4.3 Results

We first compared the prediction errors of the three methods: Linear Regression (LR), classical Artificial Neural Network (ANN), and the Long Short Term Memory Networks (LSTMs). As shown in Figure 9, LSTMs could achieve the lowest prediction error, which has been significantly reduced from 40% (of linear regression) or 23.7% (of classical neural network) to 11.5%. As for the absolute error, more than 80% of LSTMs' predictions are within the range of ±0.5kW of the ground truth, while only 70% and 45% of ANN's predictions and LR's predictions could achieve such accuracy.

	LR	ANN	LSTMs
Train RMSE	1.46 kW (39.3%)	0.66 kW (or 18.4%)	0.39 kW (or 10.8%)
Test RMSE	1.49 kW (40.0%)	0.88 kW (or 23.7%)	0.42 kW (or 11.5%)

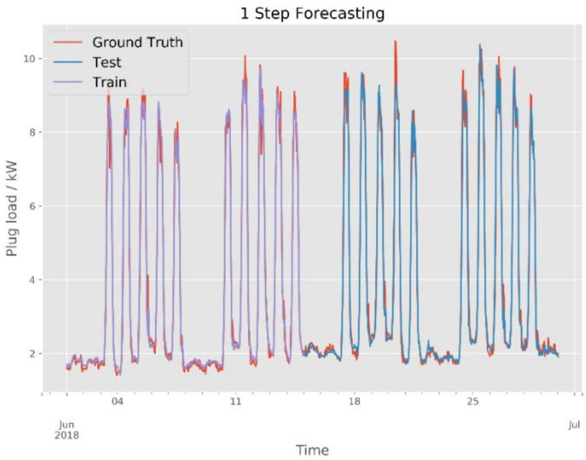
(a) Root Mean Square Error

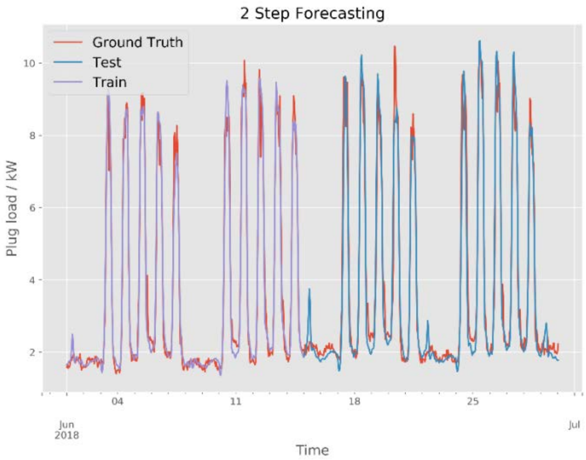


(b) Absolute Error on the Test Dataset

Figure 9 Comparison between ANN and LSTMs

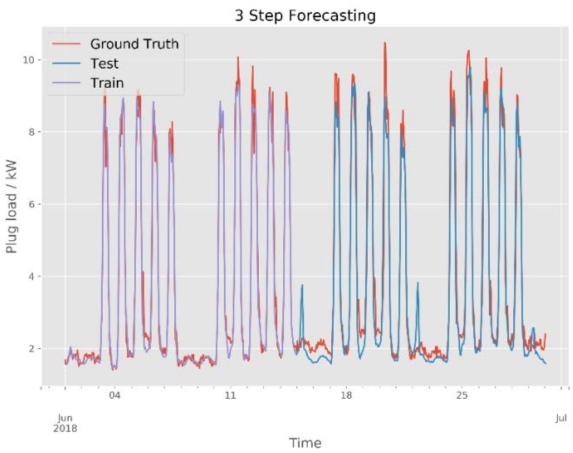
Because of the improved prediction accuracy, we chose LSTMs as the algorithm for the plug loads prediction. Figure 10 presented the prediction results for the next 1 to 8 hours. The predicted plug loads for the next hour fits well with the ground truth data. Though the prediction error starts to raise when the prediction timestamps increase but the plug load pattern could be generally followed, except for an incorrectly predicted plug load spike on Saturday morning in the 2, 3, 4, 5 and 6-hour forecasting. Another problem which happens in all of the eight predictions is the forecasted plug loads tend to underestimate the peak load; this underestimation might be as large as 20%. The major reason is the plug load in the training dataset (the first and second week) is lower than in the test dataset (the third and fourth week). Therefore, the model trained by the training data would fail to predict the peak load in the test dataset.

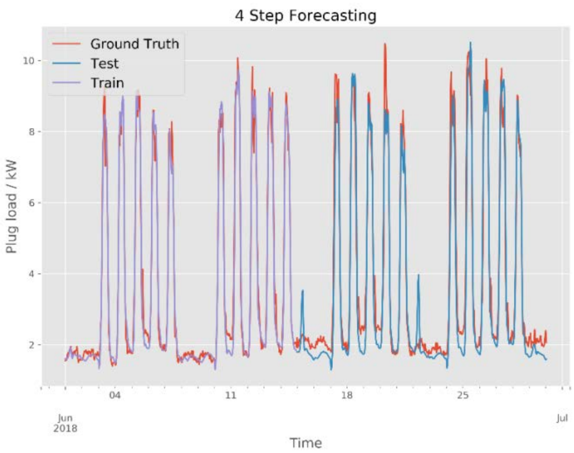




(a) 1-hour ahead forecasting

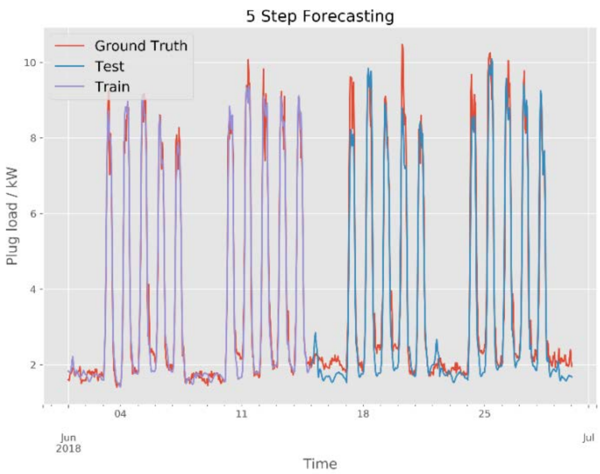
(b) 2-hour ahead forecasting

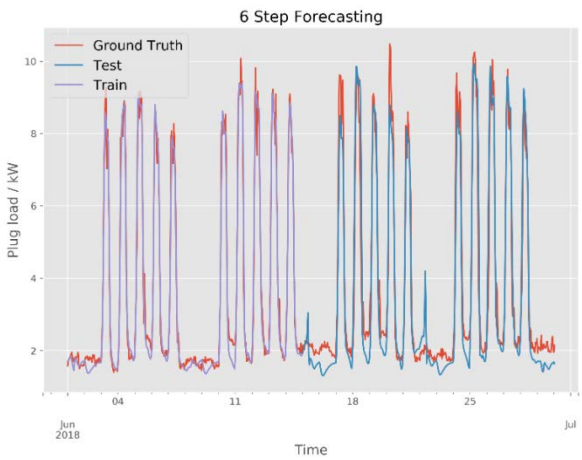




(c) 3-hour ahead forecasting

(d) 4-hour ahead forecasting



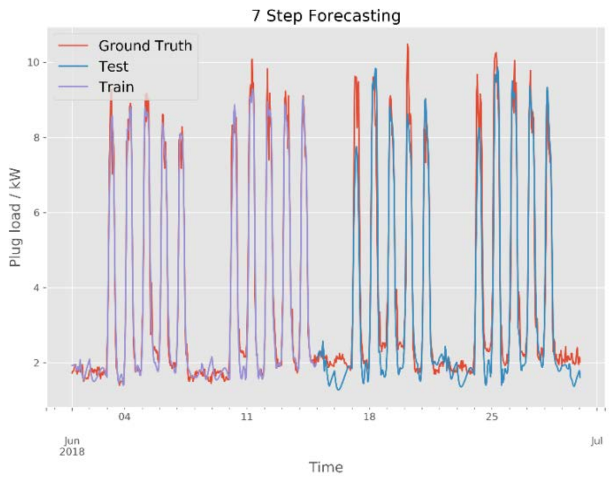


(e) 5-hour ahead forecasting

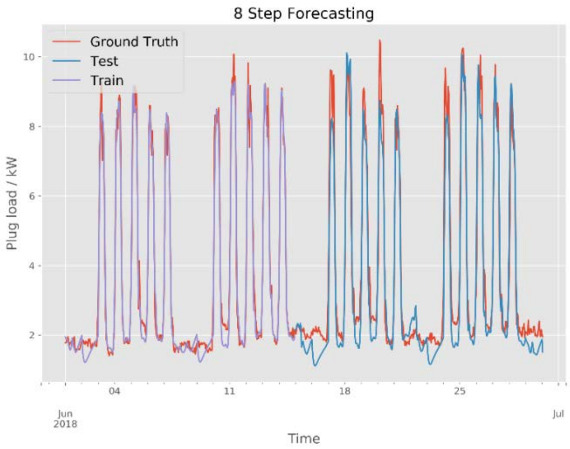
(f) 6-hour ahead forecasting

(e) 5-hour forecasting

(f) 6-hour forecasting



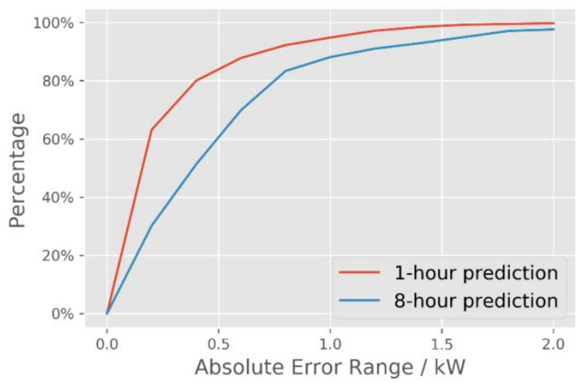
(g) 7-hour ahead forecasting



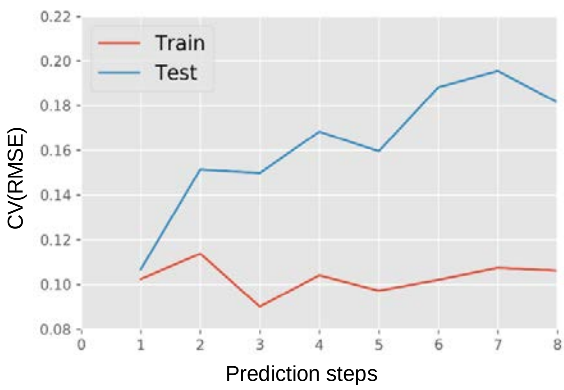
(h) 8-hour ahead forecasting

Figure 10 Prediction result

We used the normalized Root Mean Square Error, as defined in Equation 2, to quantify the prediction error. Figure 11(a) presents the absolute error of the 1-hour and 8-hour prediction. 94.8% of 1-hour predictions are within ±1 kW of the actual plug load. The prediction error is larger for the 8-hour prediction, 88.1% of prediction are within ±1 kW of the actual plug load and 97.6% are within ±2 kW. Considering the average plug load during the office hour of the area investigated is 8.6 kW, this prediction error is acceptable for predictive control.



(a) Absolute Error



(b) CV(RMSE)

Figure 11 Root Mean Square Error

Figure 11(b) compared the CV(RMSE) of different time-steps prediction. It would be more challenging if you try to predict something long time from now. As a result, the prediction error on the test set inflated with an increase of predicting steps. The prediction error for the next hour's plug load is 11%, which increases to 20% for predicting plug loads in 8 hours from now. However, either 11% or 20% is a marked improvement compared with the 39% prediction error of the linear regression method.

5. Discussion

5.1 Different approaches to forecast plug load

Time-series forecasting has many applications in buildings, such as electricity usage forecasting for flexible grid operation and thermal load prediction for HVAC control optimization. There is a wide range of approaches could be used for time-series forecasting, which could basically be categorized by two dimensions: the first dimension is whether the regression-based or NN-based approach is being used; the second dimension is whether exogenous variables are included or not, as shown in Figure 12.

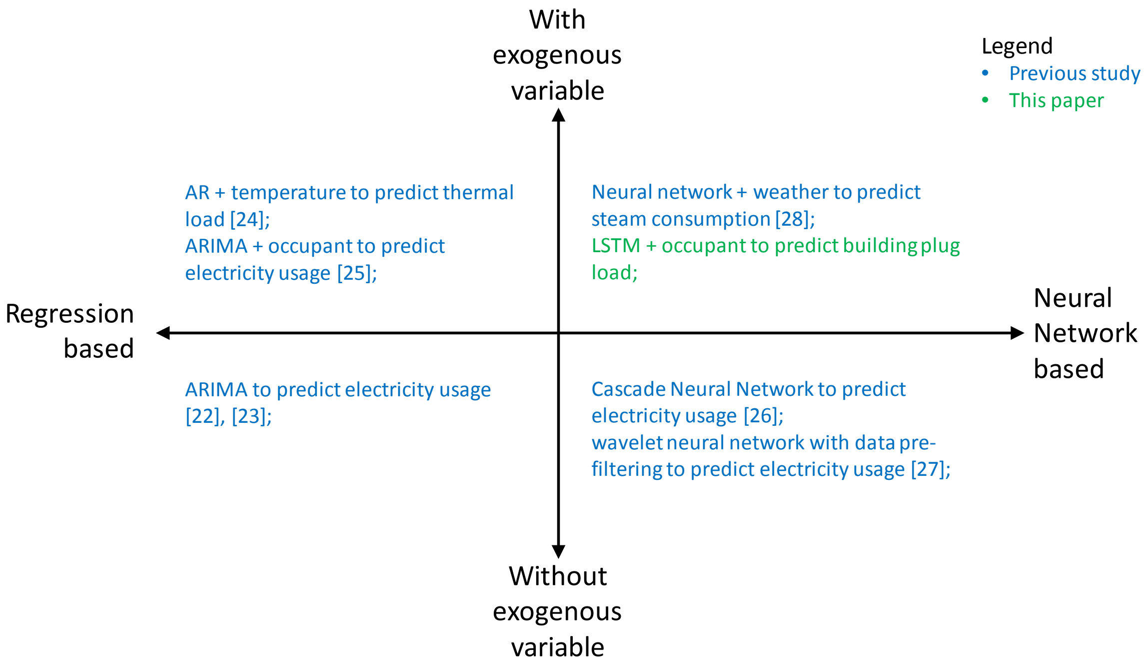
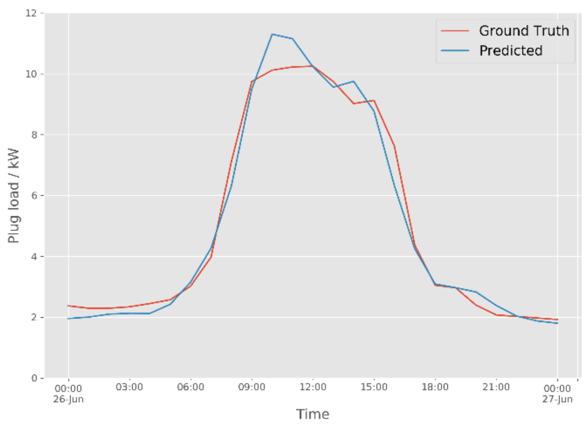


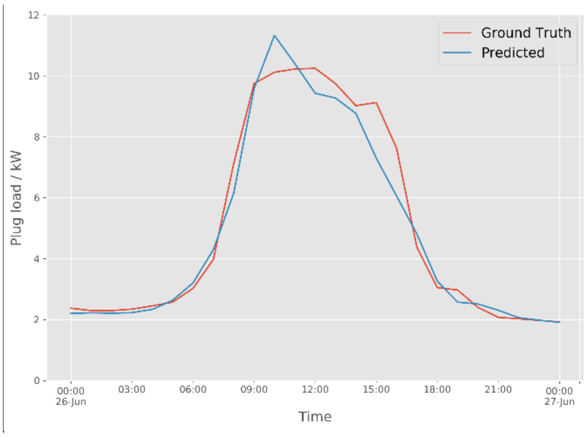
Figure 12 Different approaches for time-series forecasting for buildings applications

The regression-based approaches include ARIMA, using the steady-periodic weekly plug load profile, and regressing the residuals of this steady-periodic plug load profile with time-series modelling. The regression-based approach is easy to implement. However, the linear regression formula (as shown in Equation 1) is unable to map the complex input and output relationship, and accordingly might produce higher forecast error [48]. On the contrary, the NN-based approach could capture the complex input/output relationship by increasing the number of hidden layers and neuron units, and reflect non-linearity by implementing a non-linear activation function, such as *relu*. Meanwhile, the classical NN model could be reformed to facilitate different functions, such as the wavelet NN in [28] or the LSTM in this paper.

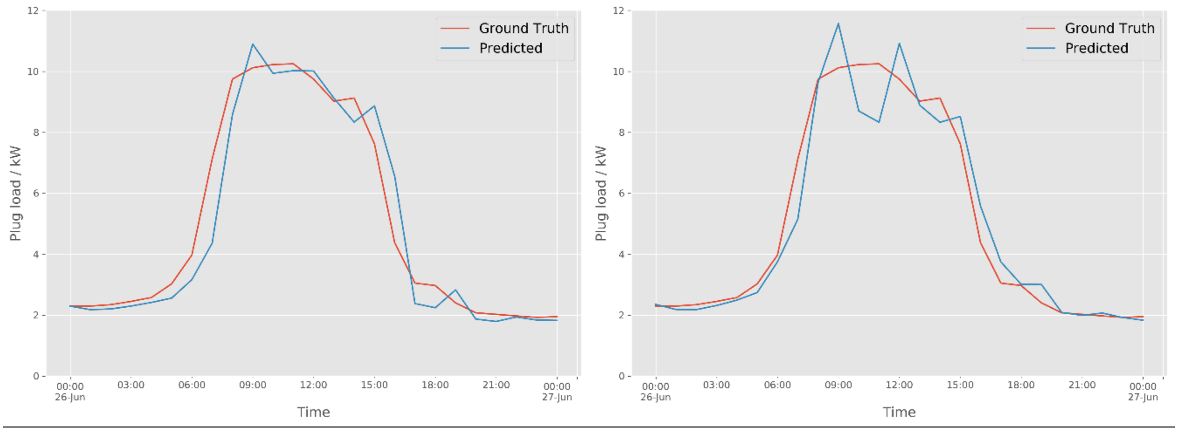
Another way to distinguish approaches is whether exogenous variables are included. Generally speaking, the more data available, the higher accuracy prediction could be achieved, especially when the added exogenous variables/features could reflect the variation that is unable to be captured by the feature of time. For example, for special and irregular events such as holding a seminar with many attendants, the plug load consumption jump could not be predicted by looking at historical data and decomposing the seasonal trend components only. Because the sudden power jump is not periodical and could not be captured by the feature of time. To deal with those special events, if the occupant count data is available as an input feature, the model's prediction accuracy could be enhanced.

To compare these four approaches in a quantitative way, we developed four models: LSTM with an exogenous variable (occupant counts), LSTM without the exogenous variable, ARIMA with an exogenous variable (occupant counts), ARIMA without the exogenous variable; and compared the 1-hour time step prediction accuracy with the data we collected. In this study, through grid search, we found the algorithm provides the best prediction when the AR order and MA order equal three. Therefore, we used *ARIMA(3,0,3)* model for this comparison. Figure 13a – 13d plot the actual plug load with the predicted plug load on a typical working day for the four prediction methods. Figure 13e compares the prediction error of different approaches on the training and test dataset. As the comparison benchmark, we used the Typical Load Profile (TLP) approach, i.e., using the average load of the same *day type (working/non-working)* and the same *hour of day* in the training set as the prediction.

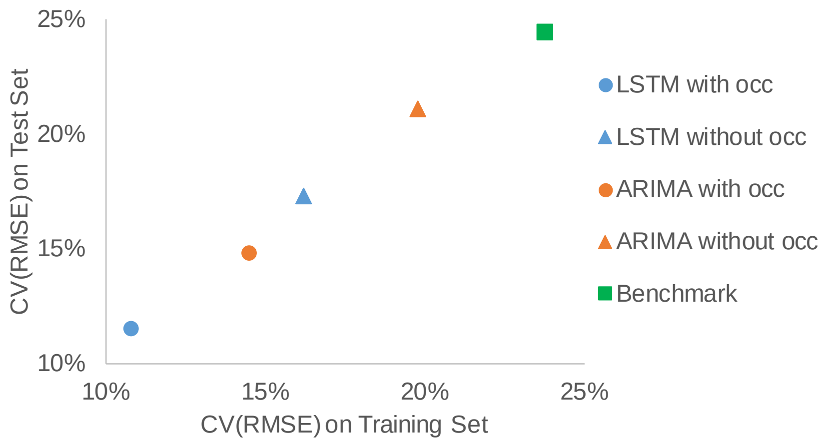




(a) LSTM with an exogenous variable (occupant count) (b) LSTM without the exogenous variable



(c) ARIMA with an exogenous variable (occupant count) (d) ARIMA without the exogenous variable



(e) Comparison on the prediction error[1]: blue for LSTM, orange for ARIMA; circle for the prediction methods with an exogenous variable (occupant count in this case), triangle for the prediction methods without the exogenous variable; the benchmark is the TLP method, i.e., using the average plug load of the same *day type (working/non-working)* and the same *hour of day* in the training set as the prediction

Figure 13 Comparison of plug load prediction accuracy for four prediction approaches

[1] The Root Mean Square Error (RMSE) is normalized by the daily average plug load

It could be observed that LSTM performs better than ARIMA in tracking the plug load variation, by comparing Figure 13a with Figure 13c, or comparing Figure 13b with Figure 13d, especially during the morning warm-up period. In terms of the comparison between methods with and without exogenous variables, for either LSTM or ARIMA, using occupant counts as exogenous variables could improve prediction accuracy. The reason could be explained by comparing Figure 13a with Figure 13b, or comparing Figure 13c with Figure 13d. Typically the plug load consumption would decrease during lunch time when people leave the building for lunch. However, on this specific day, the actual plug load did not decrease (as shown in the red line), which might be because there was a lunch-time seminar held on that specific day. If we only used the historical plug load data for plug load prediction, there is little chance to predict irregular events (for example lunch-time seminar in this case) and its influence on plug load consumption. As a result, the predicted plug load decreased during lunch time, as shown in Figure 13b and Figure 13d, which unfortunately was not what actually happened. Contrarily, if the external variable of occupant counts were used for plug load prediction, the algorithm is capable of detecting that some irregular events might be going on today, and then corrects its prediction accordingly. As a result, the plug load decrease during the lunch time in Figure 13a and Figure 13c is not as significant as that in Figure 13b and Figure 13d, which is the benefit of using exogenous variables (in this case, occupant counts) in the plug load prediction.

The difference we observed from Figure 13a – 13d is reflected in the prediction error in Figure 13e. LSTM approaches (represented by the blue color) could reduce the prediction error by around 4% compared with the ARIMA approach (represented by the orange color). Using occupant counts as an exogenous input variable could reduce the plug load prediction error by around 5%-6%, which is more significant than using different prediction algorithms.

In real buildings, occupant count data is currently not always available, since the occupant count sensors are expensive and not commonly deployed in buildings. The LSTM approach could be applied to cases when the occupant counts data is not available, as shown in Figure 13b. However, the absence of occupant counts data would result in a higher prediction error. For the specific building we tested, the prediction error is 17.3% for LSTM without occupant count information, while it is only 11.5% when the occupant count data is available and used.

As there are increasing research interests to explore a non-intrusive approach to detect occupant counts by using the existing information infrastructure in buildings, such as Wi-Fi [49], the occupant count data might become increasingly available in the future, which could help improve the plug load prediction accuracy.

5.2 Computational resource in implementation for building predictive control

The implementation of plug load prediction should balance the prediction accuracy and the requirement for computation resources. Since the plug load prediction is only one part of the predictive controller, the valuable computational resources need to cover other tasks including but not limited to the prediction of occupancy, model identification, and control optimization.

Computation is needed in two stages: the training stage and the prediction stage. With the current hyper-parameter settings listed in Table 3, the training process of the LSTM network on 2-week data (336 points) would take 576 seconds on a desktop equipped with 4-Core Intel Xeon CPU E5-1630 v4 @ 3.70GHz. The prediction process is almost instantaneous. If it is required to predict the plug load for a longer period, for instance for the next 24 hours, more computational resources are required for three reasons. First, a longer period of time-series input is needed. It might require to use the past 72 hours' data to predict the plug load for the next 24 hours. Second, more hidden layers or more number of neurons in the hidden layers might be needed as well for this more complex problem. Third, a larger training dataset is required to avoid overfitting, since the model becomes more complex and more parameters are to be trained. The three factors mentioned above would together markedly raise the training time needed if the predictive controller requires a longer timestamp prediction. If we do not adjust our settings and provide more computational resources, the prediction accuracy would rapidly attenuate with the increase of prediction time steps, as shown in Figure 14.

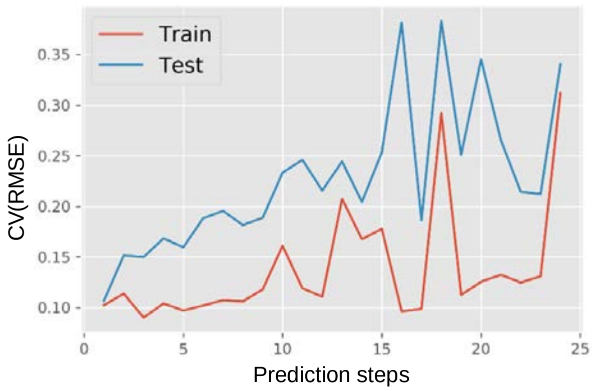


Figure 14 Prediction Error for a longer period of prediction

In this case, it is suggested, to conserve computational resources, that the LSTM network could be trained weekly or even monthly, rather than daily. The only computation needed on a daily basis is to run the prediction with the trained network, which is almost instantaneous. Keep in mind that the major computational resources are consumed on the training, rather than the prediction process.

6. Conclusions

Predictive control in buildings has gained increasing attention to reduce building energy consumption. While predicting the plug loads, a major source of internal heat gains in buildings, is a prerequisite for the predictive control of buildings, the current research on this topic focused on the regression problem, rather than the prediction problem. In this study, we used a novel method to apply Long Short Term Memory Networks to predict plug loads, which demonstrates a higher prediction accuracy than the Linear Regression and the classical Artificial Neural Network methods, and accordingly improves predictive building controls. The proposed method was tested with data collected from a real office building in Berkeley, California.

The occupant count and the time have been confirmed to have a strong influence on the plug load profile, and accordingly selected as the features for the plug load prediction. LSTM network demonstrates a high accuracy, 95% of 1-hour predictions from LSTM network are within ±1 kW of the actual plug load, given the average plug loads during the office hour is 8.6 kW. The CV(RMSE) of plug load prediction is 11% for the next hour, and 20% for the next eight hours, which is much improved from the prediction error of 39% from linear regression method.

Lastly, we compared four prediction approaches with the benchamark TLP approach: LSTM vs. ARIMA, with occupant counts vs. without occupant counts. Both LSTM and ARIMA outperform the benchamark TLP method. The LSTM approach is found to perform better, which could reduce the prediction error by around 4% compared with the ARIMA approach. LSTM could be applied when occupant count data is not available. However, when the occupant count is available and input as exogenous variables, the plug load prediction error could be reduced from 17.3% to 11.5%.

Acknowledgments

This research was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Building Technologies of the United States Department of Energy, under Contract No. DE-AC02-05CH11231. The authors appreciate the technical support on the LSTM network from Wanni Zhang and Xuan Luo, as well as data collection effort of David Blum and Baptise Ravache.

References

[1] Y.-S. Kim and J. Srebric, “Impact of occupancy rates on the building electricity consumption in commercial buildings,” *Energy Build.*, vol. 138, pp. 591–600, Mar. 2017.

[2] D. Kaneda, I. Brad Jacobson, E. Peter Rumsey, and R. Engineers, “Plug Load Reduction: The Next Big Hurdle for Net Zero Energy Building Design,” *ACEEE Summer Study Energy Effic. Build. Pac. Grove CA*, Jan. 2010.

[3] K. Marini, G. Ghatikar, and R. Diamond, “Using Dashboards to Improve Energy and Comfort in Federal Buildings,” Lawrence Berkeley National Lab. (LBNL), Berkeley, CA (United States), LBNL-4283E, Feb. 2011.

[4] G. Ghatikar, I. Cheung, S. Lanzisera, B. Wardell, M. Deshpande, and J. Ugarkar, “Miscellaneous and Electronic Loads Energy Efficiency Opportunities for Commercial Buildings: A

Collaborative Study by the United States and India," Lawrence Berkeley National Lab. (LBNL), Berkeley, CA (United States), LBNL-6287E, Apr. 2013.

[5] A. C. Menezes, A. Cripps, R. A. Buswell, and D. Bouchlaghem, "Benchmarking small power energy consumption in office buildings in the United Kingdom: A review of data published in CIBSE Guide F," *Build. Serv. Eng. Res. Technol.*, vol. 34, no. 1, pp. 73–86, Feb. 2013.

[6] Y. Ma, F. Borrelli, B. Hencey, A. Packard, and S. Bortoff, "Model Predictive Control of thermal energy storage in building cooling systems," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, Shanghai, China, 2009, pp. 392–397.

[7] W.-S. Lee, Y. –Ting Chen, and T.-H. Wu, "Optimization for ice-storage air-conditioning system using particle swarm algorithm," *Appl. Energy*, vol. 86, no. 9, pp. 1589–1595, Sep. 2009.

[8] C. E. García, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice—A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, May 1989.

[9] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, Dec. 2014.

[10] S. Yuan and R. Perez, "Multiple-zone ventilation and temperature control of a single-duct VAV system using model predictive strategy," *Energy Build.*, vol. 38, no. 10, pp. 1248–1261, Oct. 2006.

[11] S. Prívara, J. Široký, L. Ferkl, and J. Cigler, "Model predictive control of a building heating system: The first experience," *Energy Build.*, vol. 43, no. 2, pp. 564–572, Feb. 2011.

[12] H. Karlsson and C.-E. Hagentoft, "Application of model based predictive control for water-based floor heating in low energy residential buildings," *Build. Environ.*, vol. 46, no. 3, pp.

556–569, Mar. 2011.

[13] I. Hazyuk, C. Ghiaus, and D. Penhouet, “Optimal temperature control of intermittently heated buildings using Model Predictive Control: Part II – Control algorithm,” *Build. Environ.*, vol. 51, pp. 388–394, May 2012.

[14] B. Paris, J. Eynard, S. Grieu, T. Talbert, and M. Polit, “Heating control schemes for energy management in buildings,” *Energy Build.*, vol. 42, no. 10, pp. 1908–1917, Oct. 2010.

[15] H. B. Gunay, J. Bursill, B. Huchuk, W. O’Brien, and I. Beausoleil-Morrison, “Shortest-prediction-horizon model-based predictive control for individual offices,” *Build. Environ.*, vol. 82, pp. 408–419, 2014.

[16] Z. Liu, W. Li, Y. Chen, Y. Luo, and L. Zhang, “Review of energy conservation technologies for fresh air supply in zero energy buildings,” *Appl. Therm. Eng.*, vol. 148, pp. 544–556, Feb. 2019.

[17] H. B. Gunay, W. O’Brien, I. Beausoleil-Morrison, and S. Gilani, “Modeling plug-in equipment load patterns in private office spaces,” *Energy Build.*, vol. 121, pp. 234–249, Jun. 2016.

[18] A. C. Menezes, A. Cripps, R. A. Buswell, J. Wright, and D. Bouchlaghem, “Estimating the energy consumption and power demand of small power equipment in office buildings,” *Energy Build.*, vol. 75, pp. 199–209, 2014.

[19] P. Gandhi and G. S. Brager, “Commercial office plug load energy consumption trends and the role of occupant behavior,” *Energy Build.*, vol. 125, pp. 1–8, Aug. 2016.

[20] A. Mahdavi, F. Tahmasebi, and M. Kayalar, “Prediction of plug loads in office buildings: Simplified and probabilistic methods,” *Energy Build.*, vol. 129, pp. 322–329, Oct. 2016.

[21] J. G. De Gooijer and R. J. Hyndman, “25 years of time series forecasting,” *Int. J. Forecast.*, vol. 22, no. 3, pp. 443–473, Jan. 2006.

[22]S. Makridakis and M. Hibon, “ARMA Models and the Box–Jenkins Methodology,” *J. Forecast.*, vol. 16, no. 3, pp. 147–163, 1997.

[23] N. Amjady, “Short-term hourly load forecasting using time-series modeling with peak load estimation capability,” *IEEE Trans. Power Syst.*, vol. 16, no. 4, pp. 798–805, Nov. 2001.

[24] M. Espinoza, C. Joye, R. Belmans, and B. D. Moor, “Short-term load forecasting, profile identification, and customer segmentation: a methodology based on periodic time series,” *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1622–1630, Aug. 2005.

[25] K. Yun, R. Luck, P. J. Mago, and H. Cho, “Building hourly thermal load prediction using an indexed ARX model,” *Energy Build.*, vol. 54, pp. 225–233, Nov. 2012.

[26] G. R. Newsham and B. J. Birt, “Building-level Occupancy Data to Improve ARIMA-based Electricity Use Forecasts,” in *Proceedings of the 2Nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, New York, NY, USA, 2010, pp. 13–18.

[27] S. Kouhi and F. Keynia, “A new cascade NN based method to short-term load forecast in deregulated electricity market,” *Energy Convers. Manag.*, vol. 71, pp. 76–83, Jul. 2013.

[28] C. Guan, P. B. Luh, L. D. Michel, Y. Wang, and P. B. Friedland, “Very Short-Term Load Forecasting: Wavelet Neural Networks With Data Pre-Filtering,” *IEEE Trans. Power Syst.*, vol. 28, no. 1, pp. 30–41, Feb. 2013.

[29] A. Kusiak, M. Li, and Z. Zhang, “A data-driven approach for steam load prediction in buildings,” *Appl. Energy*, vol. 87, no. 3, pp. 925–933, Mar. 2010.

[30] D. Yan *et al.*, “Occupant behavior modeling for building performance simulation: Current state and future challenges,” *Energy Build.*, vol. 107, pp. 264–278, 2015.

[31] P. Hoes, J. L. M. Hensen, M. G. L. C. Loomans, V. de, and D. Bourgeois, “User behavior in

whole building simulation," *Energy Build.*, vol. 41, no. 3, pp. 295–302, 2009.

[32] B. Bordass, R. Cohen, M. Standeven, and A. Leaman, "Assessing building performance in use 3: Energy performance of the Probe buildings," *Build. Res. Inf.*, vol. 29, no. 2, pp. 114–128, 2001.

[33] R. M. Tetlow, D. van, C. P. Beaman, A. A. Elmualim, and K. Couling, "Identifying behavioural predictors of small power electricity consumption in office buildings," *Build. Environ.*, vol. 92, pp. 75–85, 2015.

[34] S. Frank, L. G. Polese, E. Rader, M. Sheppy, and J. Smith, "Extracting Operating Modes from Building Electrical Load Data," in *2011 IEEE Green Technologies Conference (IEEE-Green)*, 2011, pp. 1–6.

[35] A. M. Papadopoulos, "Forty years of regulations on the thermal performance of the building envelope in Europe: Achievements, perspectives and challenges," *Energy Build.*, vol. 127, pp. 942–952, Sep. 2016.

[36] S. Frank, L. G. Polese, E. Rader, M. Sheppy, and J. Smith, "Extracting Operating Modes from Building Electrical Load Data," in *2011 IEEE Green Technologies Conference (IEEE-Green)*, 2011, pp. 1–6.

[37] Z. Wang, T. Hong, and M. A. Piette, "Data fusion in predicting internal heat gains for office buildings through a deep learning approach," *Appl. Energy*, vol. 240, pp. 386–398, Apr. 2019.

[38] C. Jiang, M. K. Masood, Y. C. Soh, and H. Li, "Indoor occupancy estimation from carbon dioxide concentration," *Energy Build.*, vol. 131, pp. 132–141, Nov. 2016.

[39] T. Ekwevugbe, N. Brown, V. Pakka, and D. Fan, "Real-time building occupancy sensing using neural-network based sensor network," in *2013 7th IEEE International Conference on*

Digital Ecosystems and Technologies (DEST), Menlo Park, CA, USA, 2013, pp. 114–119.

[40] N. Li, G. Calis, and B. Becerik-Gerber, "Measuring and monitoring occupancy with an RFID based system for demand-driven HVAC operations," *Autom. Constr.*, vol. 24, pp. 89–99, Jul. 2012.

[41] Y. Wang and L. Shao, "Understanding occupancy pattern and improving building energy efficiency through Wi-Fi based indoor positioning," *Build. Environ.*, vol. 114, pp. 106–117, Mar. 2017.

[42] C. Martani, D. Lee, P. Robinson, R. Britter, and C. Ratti, "ENERNET: Studying the dynamic relationship between building occupancy and energy consumption," *Energy Build.*, vol. 47, pp. 584–591, Apr. 2012.

[43] Z. Li and B. Dong, "Short term predictions of occupancy in commercial buildings—Performance analysis for stochastic models and machine learning approaches," *Energy Build.*, vol. 158, pp. 268–281, Jan. 2018.

[44] J. Zou, Q. Zhao, W. Yang, and F. Wang, "Occupancy detection in the office by analyzing surveillance videos and its application to building energy conservation," *Energy Build.*, vol. 152, pp. 385–398, Oct. 2017.

[45] J. Yang, M. Santamouris, and S. E. Lee, "Review of occupancy sensing systems and occupancy modeling methodologies for the application in institutional buildings," *Energy Build.*, vol. 121, pp. 344–349, Jun. 2016.

[46] W. Wang, J. Chen, T. Hong, and N. Zhu, "Occupancy prediction through Markov based feedback recurrent neural network (M-FRNN) algorithm with WiFi probe technology," *Build. Environ.*, vol. 138, pp. 160–170, Jun. 2018.

[47] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[48] M. Q. Raza and A. Khosravi, “A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings,” *Renew. Sustain. Energy Rev.*, vol. 50, pp. 1352–1372, Oct. 2015.

[49] K. Akkaya, I. Guvenc, R. Aygun, N. Pala, and A. Kadri, “IoT-based occupancy monitoring techniques for energy-efficient smart buildings,” in *2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, 2015, pp. 58–63.