Occupant-Centric Miscellaneous Electric Loads Prediction In Buildings Using State-of-the-art Deep Learning Methods

Anooshmita Das[1], Masab Khalid Annaqeeb[2], Elie Azar[3], Vojislav Novakovic[2], Mikkel Baun Kjærgaard[1]

Mærsk Mc-Kinney Møller Institute, University of Southern Denmark, Odense 5230M, Denmark

Department of Energy and Process Engineering, Norwegian University of Science and Technology, Trondheim NO-7491, Norway

Khalifa University of Science and Technology, PO Box 127788, Abu Dhabi, United Arab Emirates (UAE)

Abstract

Buildings have emerged as one of the dominant sectors when it comes to worldwide energy consumption. While a large portion of this consumption is due to the Heating, Ventilation, and Air Conditioning (HVAC) loads, a significant portion is contributed through the use of standard equipment, also known as Miscellaneous Electric Loads (MEL). It is necessary to understand the consumption patterns to optimize the MELs of the occupants using the building and conduct accurate forecasts for building energy management. One of the methods to achieve that purpose is the employment of Deep Learning (DL) methods. This study provides an analysis using Long Short-Term Memory (LSTM) model as a baseline for predicting MELs. The predictions were conducted for a day-ahead and a week-ahead period. Furthermore, the results from the baseline model were then used in a comparative analysis with two other state-of-the-art DL models; Bidirectional Long Short-Term Memory

* Corresponding author

Email addresses: `adas@mmmi.sdu.dk` (Anooshmita Das), `masab.k.annaqeeb@ntnu.no` (Masab Khalid Annaqeeb), `elie.azar@ku.ac.ae` (Elie Azar), `vojislav.novakovic@ntnu.no` (Vojislav Novakovic), `mbkj@mmmi.sdu.dk` (Mikkel Baun Kjærgaard)

[1] Student
[2] Student
[3] Professor
[4] Professor
[5] Professor

(Bi-LSTM) and Gated Recurrent Units (GRU). The results from this study showed that both the Bi-LSTM and GRU models were significantly better than the LSTM model, especially when the prediction horizon was longer. The conclusions obtained can help implement these models in building energy management systems to draft strategic responses and schedules for more efficient energy usage.

Keywords: `Buildings`, `Consumption Patterns`, `Occupant Behavior`, `Miscellaneous Energy Loads (MEL)`, `Prediction Models`, `Plug Loads`, `Deep Learning`

---

1. Introduction

The building sector is a significant contributor to the massive and expanding energy demands which are responsible for greenhouse gases and carbon emissions [1]. Worldwide, this sector accounts for 35% to 40% of total energy consumption [2]. This ratio is even higher in countries with extreme climatic weather such as the United Arab Emirates (UAE), where buildings devour more than 70% of the power produced [3] [4]. While a significant portion of that consumption is due to the cooling loads, around 17% constitutes the load consumption from standard equipment in buildings. It is crucial to have an understanding of consumption patterns to achieve energy savings in office buildings or shared spaces [4]. To mitigate the growing energy demands of the building sector, the impact of occupant behavior has become a focus of recent studies. Various studies have investigated the impact of occupants on the energy consumption in buildings to qualitatively and quantitatively comprehend occupant behavior, foster energy efficiency, and minimize the gap between the actual and predicted energy consumption [5] [6].

Miscellaneous Electric Loads (MELs) are a critical hindrance for creating low-energy buildings, and are a significant contributor to the building's energy load. This has been documented by Roth et al. [7] in the case of residential buildings located in the United States of America (US), and in studies regarding US office buildings [8]. These loads can contribute up to 20% of the total building energy consumption, and are set to increase by 40% in the next two decades [9]. MELs in buildings get referred to as the diverse electric loads emanating from electronic devices not responsible for Heating, Ventilation, Air Conditioning (HVAC), or lighting [10] [11]. Syed and Hachem [3] considered MELs as a factor in a simulation study of a greenhouse-retail complex and found that MELs accounted for 23% of the building energy load. Burgett and Chini [12] conducted an analysis regarding the improvement of MEL prediction using occupant-centric methods. MELs are gaining attention as electronic devices get accumulated inside offices or buildings, and have become more sophisticated, thus generating upsurge in the miscellaneous electric usage [12][13]. MELs are one of the fastest expanding loads and are evolving to become one of the dominant load categories [3]. This growth reveals the fact that personal computers (PCs) and other office devices are penetrating office buildings, creating a sizeable base of installed computing equipment [1]. MELs comprise the vast majority of office equipment, while a fundamental part of them is plug loads related to Information and Communication Technologies (ICTs), such as desktops, monitors, and printers [14]. Plug load disaggregation is imperative for evaluating and investigating the underlying causes of energy wastage and developing strategies for energy reduction inside buildings [1] [3].

1.1. Challenges of Occupant-Centric Miscellaneous Electric Loads

The challenges in occupant-centric MELs energy research in buildings can get summarized as (1) unexplored and understudied as compared to design-focused research studies; (2) combined effects of uncertainty in numerous parameters responsible for the upsurge in MELs are neglected; and (3) particular types of buildings (e.g., educational facilities), or buildings exposed to extreme climatic settings are not considered, or inspected thoroughly.

Another challenge is that MELs are difficult to predict because individual occupants are in control of the electronic devices in a shared office or building. Burgett et al. [12] used an occupant based operational model to predict MELs and found that the results improved significantly when compared to the standardized occupant behavior models.

1.2. Benefits of predicting MELs

Considering the significance of MELs in the contribution to building energy, it is not only necessary to have an understanding of the consumption patterns, but also reliable estimates of those MELs. These estimates are important for decision-making processes [15], and establishing predictive control [16]. Also, accurate estimations are beneficial for predicting internal heat gains [16], and more importantly, for building performance simulations [17]. Building facility managers can benefit from an understanding of these patterns, and then look ahead into the future, for recommending energy-saving strategies, and determine the period of the peak loads.

However, these predictions often rely on typical load profiles and schedules, which are obtained from published benchmarks. A review of these benchmarks for offices in United Kingdom (UK) highlighted their inaccuracy, stemming from these benchmarks being outdated and unrepresentative of the equipment currently used in buildings (that contribute to the MELs)[18]. Only a handful of studies employ more sophisticated models for such predictions [19] [20]. For example, Menezes et al. [19] introduced two methodologies for predicting MELs, one of which made use of detailed monitored data, and the other was independent of it.

The use of deep learning models is rare, but some studies have made use of the Long Short-Term Memory (LSTM) algorithms to forecast electricity consumption in commercial and residential buildings. Wang et al. [16] applied an LSTM model to predict internal load in buildings, in order to facilitate predictive HVAC control. Marino et al. [21] employed the same technique in building energy load forecasting, while Rahman et al. [22] used it to predict electricity consumption. The motivation for using LSTM models is based on the transient nature of electricity consumption patterns. Arahal et al. [23], and Fan et al [24] explored this temporal dependency. Another problem with such predictions is the long term dependencies, which is not accounted for in vanilla Recurrent Neural Networks (vRNNs). Hochreiter and Schmidhuber [25] recommended LSTM algorithms to address both these issues. Based on these studies, the baseline model in this paper was taken as LSTM. Besides, newer models like Gated Recurrent Units (GRUs), or variants of LSTM (such as Bi-directional LSTM) have not yet been tested for these purposes, which motivated the comparative analysis in this research work.

1.3. Research contributions

This research work enables us to delve into the limitations in the previous studies by proposing a comparison between state-of-the-art prediction models to quantify the impact of occupants on MEL energy consumption. The prediction models apply time-series data analysis methods to capture potential synergetic repercussions of occupant actions and consumption patterns on building performance. In this paper, a case study is presented on a typical educational building located in the extreme hot climate of Abu Dhabi, United Arab Emirates (UAE). Such climates have not yet been the target of studies focusing on MEL predictions, signifying a notable gap in the literature.

In this research work, we try to find a solution to the following questions:

1) LSTM, Bi-LSTM, or GRU? Which model yields better results and achieve greater prediction accuracy on extrapolating consumption patterns for plug load devices using heterogeneous sensors?

2) How can we make accurate predictions on the number of occupants and energy consumption patterns in a non-intrusive and reliable manner? Does occupancy duration have more influence on prediction accuracy compared to the occupancy ratio? Does the duration of prediction (look ahead into the time horizon) have an impact on the prediction accuracy of the different models?

The significant contributions of this research work:

1) Present predictive models to forecast energy consumption more transparently and consistently and try to find out where the energy consumption is coming from, measuring in detail its energy footprint for the different plug load devices. Thus, it is interesting to identify and present state-of-the-art techniques regarding efforts to characterize, analyze, measure, and reduce the consumption of MELs.

2) Propose a comparison between state-of-the-art prediction models. Experimental results highlight that the Bi-LSTM model is slightly more accurate than the

GRU and the LSTM (baseline model). However, the GRU and Bi-LSTM converges around the same number of epochs than the baseline LSTM model.

3) Device-utilization patterns for multiple occupants get extrapolated inside the multi-utility test-space. A comparison between groundtruth and predicted patterns was carried out to demonstrate which model yields higher accuracy. Also, proposed reasonable strategies for the reduction of MELs inside buildings.

Besides the upsurge in energy consumption, MELs also influence the power quality of the network, generally by implanting harmonics and transients in the voltage signal. However, this is not the prime focus of this research work; the practitioners and model developers should consider it as another essential aspect that affects the power quality inside a building that requires further investigation [3].

The rest of the work in this paper is structured as follows: Section 2 highlights the related work in this field. Section 3 describes the problem formulation. Section 4 explicitly describes the case scenario and the experimental set-up. Section 5 illustrates the data acquisition to analysis methods in detail. Section 6 explicitly describes the deep neural network models as well as the implementation steps of the prediction models. Section 7 describes the hyper-parameter tuning and optimization strategies. Section 8 describes the experimental results for the case study, and Section 9 discusses the model evaluation results. The summary of the lessons learned is presented as a discussion in Section 10. Section 11 elucidates the applications of the prediction framework and also includes open research questions for future research directions. Finally, the conclusion is presented in Section 12.

2. Related Work

This section explores the related works found in literature, which includes the use of MEL for whole-building energy simulations, behavioral impacts on energy savings, and control systems for building energy management.

A significant amount of these studies obtained the loads from regional or global design standards [26] [4], regional surveys [12] [27] [26] or electricity meters [15] [13] [28] [29] installed in buildings. A broad range of building types are covered as well,

Table 1: Related Work

Author and Year	Data Used	Building Type	KG-CC	Methods	Results
Afshari et. al. 2014	Typical building profiles and data from Urban Planning Council	Mixed-use office	BWh	Life cycle cost analysis (LCA) Marginal abatement cost curves (MACC) EnergyPlus models	Proposed retrofits for a bussiness-as -usual (BAU) building
Burgett et. al. 2014	MEL data extracted from RECS surveys 12000 households (US EIA 2011)	Residential	Cfa/Aw	Regression model to predict MELs	Occupant characteristics predictors of MELs than building characteristics Achieved accuracy of 79%
Wang et. al. 2019	MELs, lighting, occupant counts, Wi-Fi connection counts	Office	Csb/Csa	LSTM	Reduced prediction errors compared to ASHRAE schedules
Mahdavi et. al. 2016	Occupancy (PIR) Plug loads (energy meters)	Office	Cfb	Aggregate estimation and stochastic model	Achieved NRMSE range of 12.6 - 20.3 % with stochastic models and 12 - 14% with simplified models.
Mahdavi et. al. 2017	Occupancy (PIR) Plug loads (energy meters)	Office	Cfb	Simplified and probabilistic models (using Weibull distributions)	Reduced previous RMSE values by including occupant diversity
Wang et. al. 2018	Electricity data (energy meters)	Office	Dfa	LMS nLMS RLS GMMR	RMSE (kw) for different models: LMS - 124.8 nLMS - 129.6 RLS - 122.0 GMMR - 45.2
Yalcintas 2008	Occupancy Hourly electricity measurements	Hotel	BSh	ANN	RMSE ranged from 6.81 to 16.4%
Lee et. al. 2001	Design criteria from ASHRAE Energy end-use survey by SRCI	Commercial	Cwa	Comparative analysis of design criteria and surveyed values	Recommended realistic design criteria, saving estimates of 6-22% of electricity consumption
Sarfraz et. al. 2018	Power consumption profiles Load factor profiles	Office	NA	Combined diversity factors and load fators	Recommended using office level load factors for overall load calculation
Ren et. al. 2013	Time use survey Metering data	Residential	Cfa/Cfb	AccuRate	Prediction error 6.5%
Christiansen et. al. 2015	Plug load measurements	Hospital	Cfb	Multiple regression	Error of less than 6%

most common ones being offices or commercial ones, but there is a notable dearth of academic buildings in these studies. The key findings of these related works, and the associated Köppen–Geiger climate classifications (KG-CC) of the cities they were conducted in, are presented in Table 1. The Köppen climate classification (KG-CC) system is one of the most commonly used systems, wherein the regions are represented by letters that indicate the amount of precipitation and the normal temperatures that region experiences. The regions are first divided into five main climate types; A (tropical), B (dry), C (temperate), D (continental), and E (polar). The second letter indicates the seasonal precipitation, while the third letter indicates the heat levels [30]. It can be identified from the table that there are various prediction methods used for the consumption patterns, including LSTM [16]. However, it can be noted that no works are making the use of Bi-LSTM or GRU.

3. Problem Formulation

The prediction of an occupant's energy consumption into the future at any time instant t gets represented by the occupant's presence status S and consumption patterns C. Occupancy varies throughout the week within the monitored space, and each occupant is perceived as a decision-making agent who considers multitudes of different environmental and contextual parameters before deciding which plug load devices to use. At any time instant t, the n^{th} occupant represented as (O_n) has a fixed desk position D, assigned to the occupant and plug load devices deployed at each desk P, where $P = p_0, \ldots, p_{n-1}$. This task gets perceived as a regression problem. The time-series data we observe of the occupant for different plug load devices are from time-stamp T_0 to $T_{obs-values}$. And, the predictions made from $T_{obs-values+h}$ to $T_{predict}$ with h as the desirable horizon ahead of the current time instant. h is called the windowing method, where the size of the window is a parameter that gets used to predict the miscellaneous load in the next time-stamp $t+1$, based on the current time-stamp t and past time-stamp $t-1$ and $t-2$. An occupant spawns an input consumption sequence that corresponds to the values observed, and the task of the prediction models is to generate an output sequence for the different MEL plug load devices predicting the occupant's future consumption patterns. The horizon of the prediction required (look ahead into the future) is flexible for an extension. It also depends on the demands which can range from an hour to day-ahead or week-ahead prediction. In this paper, we have a 24 hour-ahead (day-ahead) and week-ahead prediction from the different deep learning models.

4. Experimental Set-up and Design Scenario

The data collection was performed at a research facility located in the city of Abu Dhabi, UAE. The area used for data collection served as a shared work-space consisting of eight individual desks, which would be occupied by up to 6 graduate students at a time. The remaining two desks were designated as shared desks for the occupants. Also, another common table was present for shared use, as shown in Figure 1. Since students used the test-space, the specific occupancy of each desk was subject to change every semester, with incoming and outgoing students.

In total, eight different occupants were included in the study over the entire course of the data collection, which spanned over a period of eight months during April - November in the year 2017. Each desk contained a personal lamp for illumination, controlled by the occupant, while the area itself was illuminated by six different lighting fixtures, each controlled by localized motion sensors. Besides, each desk was equipped with six power outlets, including the power source for the lamp. While the university was operational from 8 am to 5 pm on weekdays, the area was open for access throughout the week. Moreover, it should be noted that the area of study was not a fully controlled environment, which meant that the occupants had flexible hours for work, and while there were six primary occupants in the area, there were no restrictions for visitors to access the area. Such visitors occasionally occupied the shared table. However, all these visitors were students or staff members of the institute.

5. Approach

In this section, we explicitly describe the methodology implemented in this research work. The overview is also highlighted in Figure 2.

5.1. Sensor Placement Strategy and Calibration

The occupants were monitored with regards to their presence, device consumption patterns (via plug loads), and the associated environmental parameters such as illuminance, temperature, and relative humidity. This was achieved with the use

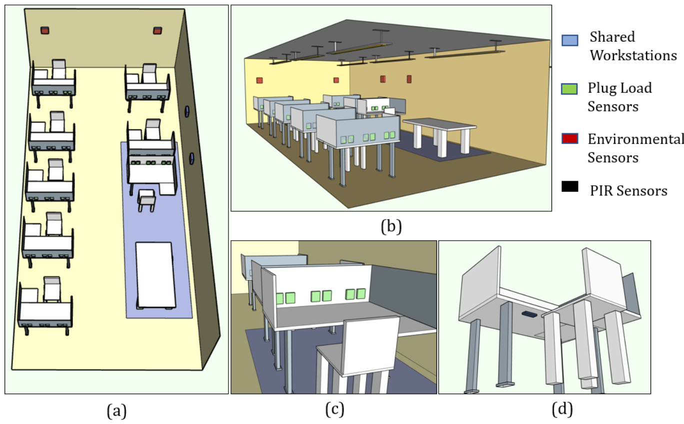

Figure 1: Experimental Set-up a) Bird's view angle of the space layout b) Environmental sensor placement in the space layout c) Six Plug-load Sensors on each desk (Miscellaneous, Dock, Monitor 1, Monitor 2, Laptop, Lamp) d) PIR sensors on each occupant desk

of three different kinds of sensing modalities: plug load sensors, PIR (Passive Infra Red) sensors for the occupant presence, and environmental sensors to record the illuminance (lighting status), temperature, and humidity. For more information about the specific sensors, the individual datasheet can be referred to [31] [32]. In total, nine PIR sensors from OccupEye [31] were deployed in the area, eight on the work desks, and one on the common table. The sensors were placed under the work desks, as shown in Figure 1 (d), and no other objects were placed in the area beneath the desk, so as to avoid occlusion and have a better range of detection. To find the optimal position of the PIR sensor at each desk, three factors were taken into consideration; first, the sensors needed to be able to detect the occupant's presence at all positions at the desk, and second, the optimal position should enable the sensor to avoid any triggers from passers-by near the desk. Thirdly, the sensors are mounted to ensure there is no occlusion in its Field of View (FoV). Different positions and scenarios were performed, recording the triggers from the sensor to determine the optimal position with respect to the three conditions.

For the plug loads, each desk was equipped with six sensors, and each power outlet was designated to be used by a specific device, as seen in Figure 1 (c). This

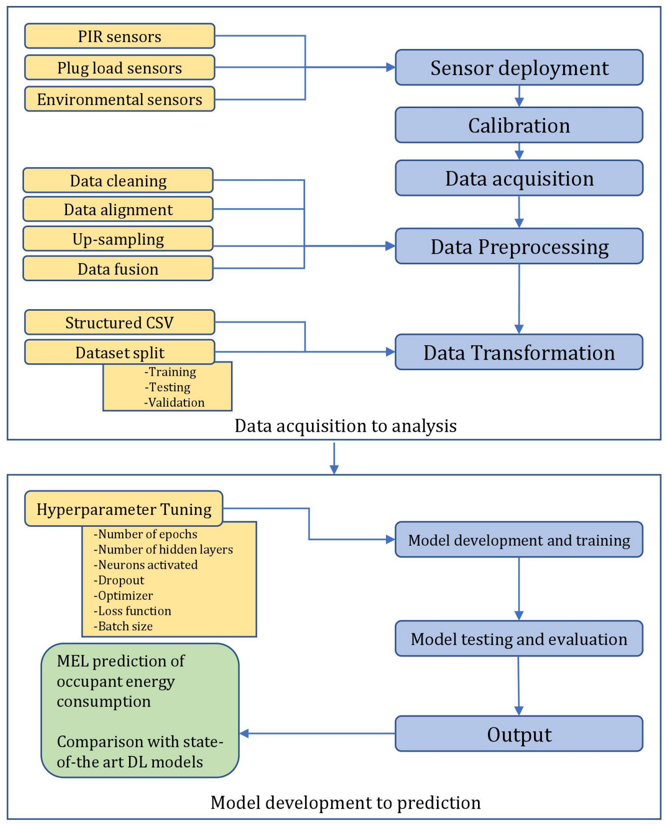

Figure 2: Overview of the methodology

was done in order to label and segment the plug load consumption according to the devices plugged in. The devices used for each socket and sensor were kept constant throughout the data collection process. To ensure the due procedure was followed, weekly checks were conducted.

As for the environmental sensors, one of the objectives of the study was to be independent of any Building Management System (BMS), which is why the data was not extracted from any BMS. Instead, the environmental sensors were used locally to monitor the environmental conditions around the occupant. The sensors were placed at different heights and positions near the lighting fixtures in order to accurately capture the luminance levels for each, without any significant interference from the others. Hence, six such sensors were used, one for each lighting fixture.

While the data collection included input from all the sensors, it should be noted that the environmental data was not used in the deep learning models in this research work. The environmental data was collected to understand the relationships and dependencies of occupants' visual and thermal comfort with the energy consumption patterns, which was not included in the scope of this study. However, there is a possibility to extend this work in that direction in the future. With regards to the data from the two remaining sensing modalities, the application of the PIR data was for the authors to understand the occupant presence status within the domain and context of the space. The deep learning models were modeled for the weeks wherein the authors had determined that the occupants were present in the area of study, and tested the models in the week with maximum occupancy. The plug load data represented the MEL consumption data for each device, and was the dataset used to develop the deep learning models.

5.2. Data Acquisition

The occupants were monitored for a period of 8 months (April – November 2017) in total. However, due to the malfunction of devices and the associated missing data, the time period taken into consideration was for around 6 months, or 167 days precisely. The data were individually collected from each sensor's respective data storage platform in the form of comma-separated values (CSV) files. Along with the routine checks to ensure the functionality of sensors, this procedure was conducted at weekly intervals. Among the sensors deployed, both PIR and environmental sensors were battery-operated and could collect data for up to 6 months at a stretch. The plug load sensors used the power source they were attached to for functioning.

The data collection process for the environmental and plug load sensors was continuous, reporting the associated device consumption and environmental parameters at 15-minute intervals. The PIR sensors, however, were event-based, recording triggers as soon as the occupant enters the detection range of the sensor. The sensor also registered the absence as soon as the occupant was out of the range. However, to account for cases where the occupant may have momentarily left the area, on occupancy duration of five minutes was selected. The five minutes was a wait-period before reporting for an 'Absent' reading; i.e., the sensor would have to detect the absence of the occupant for a continuous five minutes.

5.3. Data Pre-processing and Data Transformation

In this section, we explicitly describe the pre-processing steps for the collected data. Since the PIR sensor was event-based, it was not aligned to a consistent temporal resolution. Also, those sensors recorded 'blank' triggers when there was no change in the previously reported presence status. To counter these issues, all the 'blank' triggers had to be first substituted with the occupancy triggers they were indicative of. The data was then processed to sort and replicate the last recorded trigger for every fifteen minutes. After up-sampling, each sensor now had four readings every hour (one reading every fifteen minutes), totaling 96 readings per day.

For anomaly detection, the routine checks involved determining if data was being continuously reported by each sensor. In some cases, there was some missing data, frequently attributed to sensor malfunction. The malfunctions included sensors running out of battery and removal of sensors from designated positions by maintenance personnel.

Considering the nature of the study, it was necessary to have consistent data for each device and its user, and in order to maintain that consistency, weeks with missing data were discarded. That was the reason for the difference between the original monitored time of 8 months and the final duration of 6 months after data pre-processing. The final data set consisted of 16,032 rows (96 readings x 167 days) of data for each occupant. After the pre-processing step, the data was transformed into a structured and consistent format to work with and build the prediction models.

The sufficient amount of data needed for modeling depends both on the complexity of the problem at hand and on the complexity of the chosen algorithm. However, one heuristic about the relation between the number of features and training samples is that the number of training data should be 10 times more than the number of features. Our MEL dataset comprises of consumption data (via plug loads) for 6 devices (Miscellaneous, Monitor1, Monitor2, Docking Station, Laptop and Lamp).

The dataset was split into training, testing and validation sets in the ratio 60 %: 20 %: 20 % respectively, which consists of 57,715 training samples, 19,238 testing samples and 19,239 validation samples. There are 6 features in the dataset. Thus, it is ensured that the training data, with approximately around 60,000 samples and 6 features in the dataset is sufficient to build the predictive models.

5.4. Privacy Handling and Data Suppression

This section details the steps that were taken to ensure the privacy of the occupants that were monitored. Each occupant was monitored on a voluntary basis, and was provided with a consent form detailing the data that was being collected, and the duration and aims of the study. This consent form was approved by the Human Research Ethics Committee of the host university. In order to avoid revealing the identification of the occupant, the data was labeled as Occupant 1, 2, etc. No names or other personal details of the occupants were recorded.

The most sensitive part of this dataset is the data collected outside of the opening hours since those readings are most likely collected from employees working in the monitored area. This part of the data can be used by the employer to estimate the work performance of the employees in the area. Data suppression is generally referred to as the process of withholding or ad-hoc removal of a selected piece of information from the data to protect the identity, privacy, and revealing confidential information of any individual occupant. This is a crucial step when sharing data publicly or with third parties to protect the privacy of each occupant. Due to this, for the future release of the datasets, we will not release any of the readings collected outside of the opening hours. Furthermore, the occupants in the monitored area also have the right to privacy. We have decided to protect the identity of the days, by not including the dates as part of the dataset. Furthermore, re-ordering the days by mapping the date component, thus creating a random permutation of the days.

Similarly, for the weeks, we have a Week ID indicator. These precautions make it significantly more difficult for adversaries to perform data linkage attacks upon the released data, and hence identifying/revealing the identity of the occupants or the location of the monitored area. In the dataset, we have introduced a workday indicator that accounts for weekends and national holidays. Furthermore, we included a national holiday indicator. The time-stamp remains in the up-sampled form (15 minutes). Table 2, presents an overview of all the attributes in the dataset.

Later, we want to release the full dataset, which can serve for benchmarking and foster data-driven research in occupant centric MEL prediction. This dataset used in this paper would help to extract the inter-relationships of different devices influencing occupant behavior and utilization patterns.

Table 2: The attributes included in the dataset (which is collected for a period 8 months)

Attributes	Description
Time	Time-stamp when the entries were collected.
Time-Interval	every 15 minutes (up-sampled).
Day ID	Randomly assigned number which holds the ID for the specific day in the dataset.
Week ID	Randomly assigned number which holds the ID for the specific week in the dataset.
Workday	Boolean, telling if the entries were collected on a workday.
Weekend	Boolean, telling if the entries were collected on a weekend.
Holiday	Boolean, telling if the entries were collected on a national holiday.
Occupant ID	Each head-count is assigned a unique Occupant-ID, in number.
Presence Status	The occupancy status in a boolean format (0/1) from PIR sensors.
No. of occupants	8 occupants observed (in-situ)
Consumption Data	MEL Devices: Miscellaneous, Dock, Monitor 1, Monitor2, Laptop, Lamp

6. Models Used

Initially, the time-series dataset gets split into training, test, and validation set. The prediction models have been developed using the training sets, and the predictions are made on the test set, which is unseen by the model. The prediction model used for comparison are - Long Short-Term Memory (LSTM) [baseline model], Bidirectional Long Short-Term Memory (Bi-LSTM), and Gated Recurrent Unit (GRU).

The next step is transforming the time-series data into a supervised learning problem, i.e., the data gets organized into input and output patterns where the observation at the previous time step is fed to the network as an input to predict the observation at the current time step. Another step would be that the observations get transformed to have a specific scale, i.e., re-scale the data values to lie between -1 and 1. The re-scaling gets done to meet the default hyperbolic tangent activation function of the model. These transforms are inverted on predictions to revert them into their original scale before calculating an error score.

LSTM was proposed by Hochreiter et al. in 1997 [25], and the major motivation behind building the model using LSTM for MEL predictions on plug load devices for multiple occupants is that the model can account for energy savings and efficient device utilization within the test zone. Also, to comprehend the significant factors influencing decision-making for different device choices and individual consumption patterns in the multi-utility test zone. This could further help to perform intelligent building operations and curtail energy wastage. Bidirectional LSTMs (Bi-LSTMs) proposed by Graves et al. in 2005 [33] are an augmented version of conventional LSTMs that can boost up the performance of the model on prediction problems. Cho et al. in 2014 proposed GRU and it got successfully implemented for sequence prediction tasks [34].

For implementing the prediction models, we have used Python language to evaluate and process the time-series data. For the data analysis, we developed prototypes using Scikit-Learn, Scipy, Pandas, Numpy, Seaborn, Matplotlib libraries. We also used sequential models in Keras (compatible with python 3.6 version) under Tensor-flow as back-end. We used a workstation with an Intel i7-8850H with a base frequency of 2.6 GHz and a turbo boost to a decent 4.3 GHz. Our algorithm utilizes Tensor-flow as the back-end to evaluate the overall model performance and computational efficiency. The explicit description of how each model gets implemented follows in the next subsections:

6.1. Long Short-Term Memory (LSTM) [Baseline Model]

The advantage of using LSTM as a baseline model over others is that LSTM's can randomize the order dependencies, possess memory blocks across observations in the input, which Multi-Layer Perceptron (MLP's) lack. The gates involved in implementing the LSTM and the information-flow is described below:

1) Forget Gate(Ft) decides which information to delete that is not important from the previous time-stamp. The unnecessary parts of the previous cell state are forgotten. To decide which information to be omitted to form the cell in that particular time step, it is decided by the sigmoid function σ. It looks at the previous state $h(t-1)$ and the memory of the previous unit represented as m_{t-1} and the current input X_t then compares the function. The weights get represented by w. The weights in the LSTM module is updated using Back-Propagation through Time (BPTT). This enables stability in the model. This can be given by the equation 1:

$$Ft = \sigma(wX_{t} + wh_{t-1} + wm_{t-1} + bias(Ft)) \tag{1}$$

2) The second layer comprises 2 parts i.e., input gate (It) and cell state (St). There are two activation functions, one is the sigmoid function, and the other is tanh. The sigmoid function then decides which values to let through (0 or 1). The tanh function gives the weightage to the values which are passed, deciding their level of importance (-1 to 1). Cell state (St) can selectively update cell state values and decide what part of the current cell state makes it to the output and gets defined by the equation 2 and 3:

$$It = \sigma(wX_{t} + wh_{t-1} + wm_{t-1} + bias(It)) \tag{2}$$

$$St = \sigma(wX_{t} + wh_{t-1} + wm_{t-1} + bias(St)) \tag{3}$$

3) The third step is to decide what will be the output. It gets governed by the output gate (Ot), and Ht captures the entire process, defined in equation 4 and 5:

$$Ot = \sigma(wX_{t} + wh_{t-1} + wm_{t} + bias(Ot)) \tag{4}$$

$$Ht = (Ot) * tanh(St) \tag{5}$$

6.2. Bidirectional Long Short-Term Memory (Bi-LSTM)

Bi-LSTMs train on two rather than one LSTM on the input time-series data. Bi-LSTMs runs the inputs in two approaches, one from past to future and one from future to past, i.e., the first on the input sequence as it is, and the second is a reversed copy of the input sequence. However, what varies this approach from others is that in the LSTM that runs backward, the information from the future gets retained. Using the hidden states in combination enables the preservation of information at any point in time from both past and future. The implementation steps are defined by equation 6, 7 and 8.

$$h_t^f = f_h(wX_{\mathrm{t}}, h_{t-1}^f)...Forward \tag{6}$$

$$h_t^b = f_h(wX_{\mathrm{t}}, h_{t+1}^b)...Backward \tag{7}$$

$$h_t = f_o(h_t^f, h_t^b)...OutputLayer \tag{8}$$

Bi-LSTMs can provide additional context to the neural network and result in faster convergence and even fuller learning on the prediction problem, although it depends on the task. The structure of Bi-LSTM allows having both backward and forward information about the sequence of consumption patterns at every time step. By using the information from the future, it becomes easier for the network to predict the consumption patterns efficiently.

6.3. Gated Recurrent Unit (GRU)

In this section, we describe the execution of the prediction model, using Gated Recurrent Units (GRU). GRU performs better since it has a less complicated structure and is computationally less expensive. Also, the GRU training phase is faster than RNN or LSTMs on limited training data.

Another advantage of GRU is that it solves the problem of vanishing gradient, which generally occurs with vanilla RNN. Vanishing Gradient problem occurs when the gradient shrinks as it back-propagates through time. If the gradient value becomes too small, it doesn't contribute much in the learning phase.

To solve the vanishing gradient problem and short-term memory, the gates in GRU help to regulate the flow of information and handle which data in the sequence is essential to retain and others to throw away. By continuing this process, the relevant information is passed along the sequence-chain and makes accurate predictions.

The GRU discloses the memory content at each time-stamp between the previous and the upcoming next memory content with the help of an update gate. The update gate governs how much of the previous memory state should get forgotten and how much new content should pass into the future. The update gate is analogous to the forget and input gates of an LSTM model. GRU has fewer parameters and train a bit faster and also need fewer data to generalize. GRU's got rid of the cell state value and directly use the hidden layers to transfer the information ahead. The update gate z_t at time-stamp t gets defined in the equation 9:

$$z_t = \sigma(w^z X_t + U^z h_{t\text{-}1}) \tag{9}$$

X_t is the input that gets multiplied with its weight w^z. Similarly, $h_{t\text{-}1}$, which holds the information regarding the previous time-stamp t-1, is multiplied by its own weight U^z. The results added together, and a sigmoid activation function gets implemented to squish the results between 0 and 1. The reset gate decides how much of the past information to forget. It can be defined as given in equation 10:

$$r_t = \sigma(w^r X_t + U^r h_{t\text{-}1}) \tag{10}$$

The current memory content h_t, which uses the reset gate to store the necessary information from the previous time-stamp, is given in equation 11. An element-wise product between the reset gate r_t and $Uh_{t\text{-}1}$ is performed. This step handles what information is needed to be removed from the previous time-stamps and apply the tanh activation function. The tanh activation function squishes the values between -1 and 1, thus regulating the output while performing the parameter tuning through successive runs.

$$h_t = tanh(wX_t + r_t \odot Uh_{t\text{-}1}) \quad (11)$$

6.4. Model Evaluation Metrics Used

For the evaluation, a rolling-forecast gets implemented, also known as walk-forward model validation. Every time step of the test set gets escorted one at a time. The prediction models are used to make a forecast on the MEL consumption for the time step, and then the actual expected value from the test dataset gets captured and made accessible to the model for the prediction on the next time step. This walk-forward model validation mimics a sophisticated real-world setting where training data would be accessible, and test data gets used in forecasting (in a one-shot method), i.e., the energy consumption of the occupants. All forecasts on the test dataset get accumulated; thus, an error score gets calculated to encapsulate the skill of the prediction model.

For the evaluation of the models, the first metric used is Root Mean Squared Error (RMSE), and it quantifies the amount by which the estimator deviates from the targeted output. The RMSE gets used as the evaluation metric as it penalizes significant errors, and the outcome is a score that is in the same units as the forecasted data, i.e., occupants plug load energy consumption.

Another metric used is Mean Absolute Error (MAE), which is the average of the absolute values of the differences between predicted and the corresponding observation in the plug load consumption data. The evaluation results are elucidated in Section 9.

7. Hyper-parameter tuning and optimization strategies

The hyper-parameter for a model is a configuration that is external to the model and whose value cannot get approximated from the data. The hyper-parameters are often practiced to help estimate model parameters and are usually specified heuristically. The hyper-parameters get tuned for a given prediction problem. The best value for a parameter tuning on a given problem is not known; however, we often use different rules of thumb or explore for the best value by trial and error. The hyper-parameters for the prediction model gets discussed below:

The *number of training epochs* is the number of times that the entire training dataset gets demonstrated to the network during training. The *batch size* of a model can be referred to as iterative gradient descent in the patterns exhibited and defining what patterns to read at one go and restore in-memory and discard others before the weights get updated in the model [35]. *Dropout layer* was added to prevent overfitting. *Mean squared error (MSE)* is used as the loss function as it helps to evaluate the accuracy of the model in predicting the test data.

Table 3: Hyper-parameters selected for the three different deep learning models

Hyper-parameters	**LSTM**	**Bi-LSTM**	**GRU**
1) No. of Training Epochs	100	100	100
2) No. of Neurons Activated	64	64	64
3) No. of Layers	3	2	3
4) Batch-size	64	256	256
5) Dropout	0.1	0.1	0.1
6) Learning Rate	0.001	0.001	0.001
7) Activation Function	tanh	tanh	tanh
8) Optimizer	Adam	Adam	Adam
9) Loss Function	MSE	MSE	MSE
10) Evaluation Metric	MSE MAE MAPE	MSE MAE MAPE	MSE MAE MAPE

Optimization Strategies:. We implemented a grid search for hyper-parameter optimization. Grid search helps to find the optimal hyper-parameters of the predictive model, which results in better predictions. Table 3 highlights the selected hyper-parameters for the three different deep learning models. Also, we choose Adam because it can get perceived as a merged version of RMSprop and Stochastic Gradient Descent (SGD) with momentum [36]. The advantage of using Adam is that it is computationally efficient and straightforward to implement for big datasets; also, hyper-parameters have intuitive interpretation capabilities and require less parameter tuning. Adam is also appropriate for noisy datasets or a sparse gradient.

Batch Normalization (BN) is another optimization technique where the distribution of each layer in the input of the deep learning model can change quite consistently. As the input changes, the model parameters keep changing too, during the training phase. Lower learning rates can be set to deal with the dynamic parameters. However, lower learning rates slow down convergence and therefore make the learning process slower. There seems to be a trade-off between dynamic parameters and learning rates. The trade-off gets more emphasized with saturating nonlinearities across different layers in the deep learning model. It has been demonstrated how by initializing values of parameters with zero mean and unit variance, also known as normalization. By updating parameters as training progresses for every mini-batch, and then back-propagating through time (BPTT), it is viable to use higher learning rates, and to pay less attention to the initial values of the parameters. This process not only makes learning more robust but also speeds up the training process, as higher learning rates need fewer epochs to converge. Notably, batch normalization acts as a regularizer, requiring less dropout and discouraging overfitting [37], normalizing the loss, i.e., sum the loss terms along with the sequences and divide by the maximum length of the sequence. In that way, it becomes easier to reuse the hyper-parameters between multiple case scenarios.

Another strategy is early pruning of the training phase to avoid overfitting and to evade from training a neural network more than needed. Detecting when a model offsets to overfit the data is a challenge and one of the few methods to discover when the network is not learning anything new about the data comprises of investigating the validation loss, which gets calculated on the validation dataset. If the validation loss does not improve, it indicates that further training doesn't add anything new to

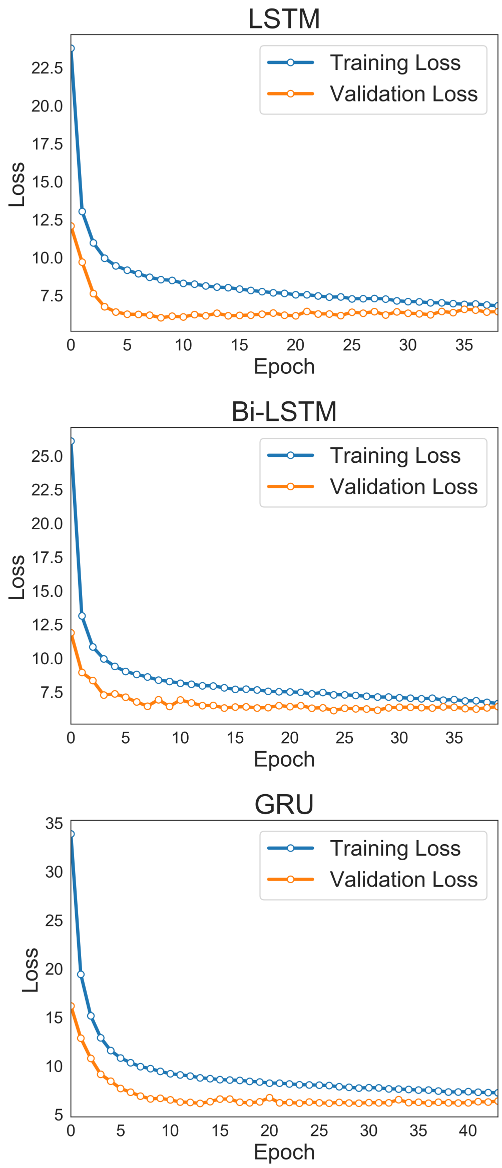

Figure 3: Training Loss vs. Validation Loss for the LSTM, Bi-LSTM and GRU Model respectively, using Mean Squared Error as the Loss Function. The unit is in kW.

the current parameters of the model. Early pruning is decided based on the patience parameter in Keras. In our case, a value of 30 epochs was used for the early pruning of the models if the validation loss doesn't decrease and remains almost stagnant for

30 epochs).

A good fit represents a case where the performance of the deep learning model is extremely well on both the training and validation sets. This can get diagnosed from a plot where the train and validation loss decrease and stabilize around the same number of epochs. We have highlighted diagnostic plots for the three deep learning models, see Figure 3 for LSTM, Bi-LSTM, and GRU, respectively. The Figure 3 shows that the models stabilize around the same point indicating a well-fitted model. The loss function used is Mean Squared Error (MSE).

8. Experimental Results

This section highlights the day-ahead and week-ahead prediction for the different plug load devices in Figure 4 and 5, respectively, for Occupant No. 6. The week chosen for the week-ahead prediction was selected based on its high occupancy. In order to compare the occupancy between different weeks, diversity factors were used. Diversity factors represent the ratio of actual occupancy to the maximum possible occupancy of an hour. These factors form the basis for generating standard occupancy schedules in offices [38] [39]. The week selected was the one with the highest diversity factor.

The six plug load devices are - Miscellaneous (no fixed device label i.e., occupants had the flexibility to plug in any device), dock charging station, monitor 1, monitor 2, laptop, and desk lamp.

From Figure 4, it is evident that the traffic of occupants starts at around 9 am. Based on the prediction result analysis, Occupant No. 6 has unstable patterns for the Miscellaneous plug load. Miscellaneous is hard to predict, and the reason is that there is less training data, and it becomes hard for the models to learn and capture the consumption patterns for it. However, the Bi-LSTM model was the best one to capture the miscellaneous plug load patterns with the least RMSE and MAE. The lower value of RMSE and MAE interprets that the predictions were closer to the groundtruth data.

As seen in Figure 4, dock consumption had quite a consistent pattern during

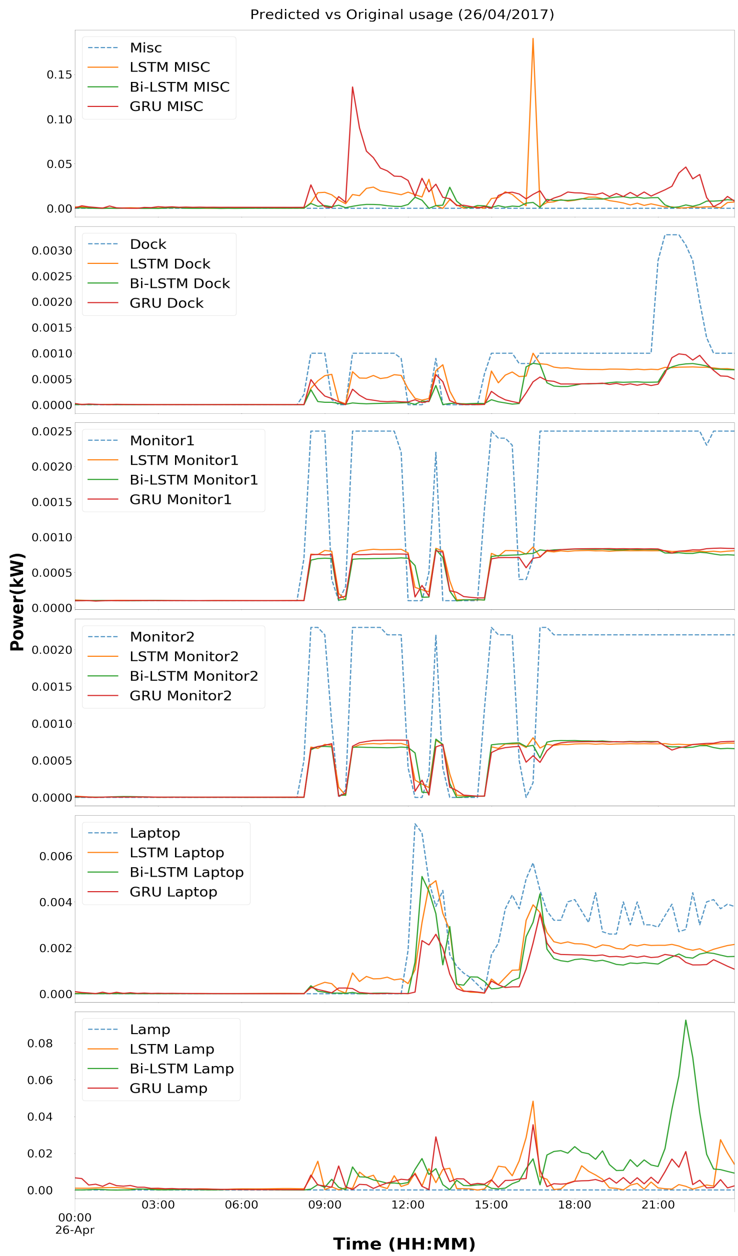

Figure 4: Day-ahead Prediction for Occupant 6 shows comparison between the prediction models (LSTM (baseline), Bi-LSTM and GRU with the actual MEL consumption (dashed) for the six plug load devices [in kW].

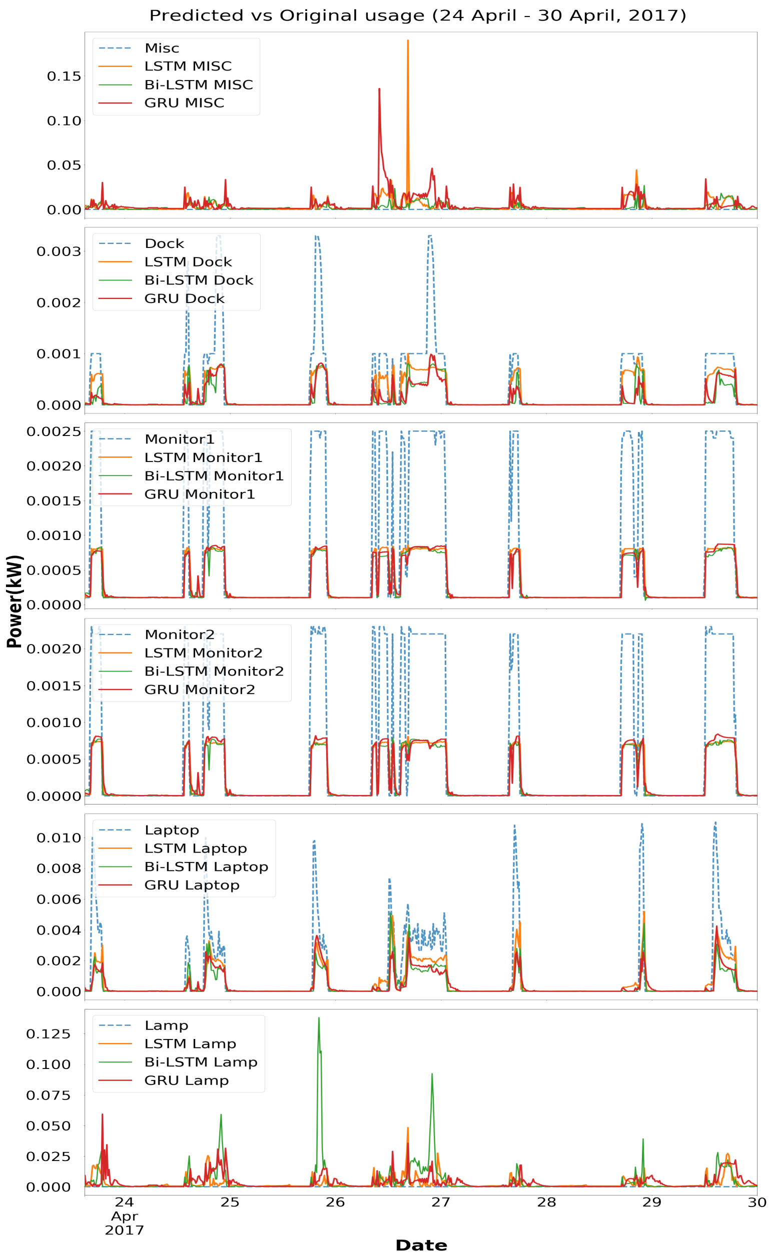

Figure 5: Week-ahead Prediction for Occupant 6 shows comparison between the prediction models (LSTM (baseline), Bi-LSTM and GRU) with the actual MEL consumption (dashed) for the six plug load devices [in kW].

office hours. However, it can be noted that there is a peak right before midnight, which coincides with the deviation in the models' prediction (LSTM, Bi-LSTM, and GRU). This peak is again repeated in the monitor and laptop consumption as well. This would constitute a scenario where the desk occupant has stayed much longer than the regular office hours. Nevertheless, the deviation of the prediction is less than two decimal points, indicating the accuracy and reliability of such a prediction. The peak consumption of the monitors starts with the regular office hours, taking a dip at noon, which represents the occupants leaving for lunch. The similarity in the consumption patterns for the monitors is corroborated by the fact that both of the monitors are simultaneously switched on and used by the occupant.

The peak load for laptop consumption occurs from 12 pm to 1 pm, indicating a lag in the initiation of the consumption compared to the monitors and the dock. In comparison to the rest of the devices, the lamp is rarely used by the occupant. The area of study was well lit by the motion-controlled lights, which were complemented with natural daylight during the day. Hence, the need to use the lamps did not often arise, which can be seen in both Figure 4 and Figure 5.

From the week-ahead predictions in Figure 5, it can be seen that the largest deviations occur in the miscellaneous plug load and lamp consumption. Both of these categories had the least amount of training data since the occupants rarely used the lamps, and miscellaneous plug load had no consistent device attached to it. Figure 6 highlights validation loss [in kW] for the different prediction models. It shows that the Bi-LSTM model is more stable compared to GRU and the baseline LSTM model.

The monitors consumed extra power, and the least power was devoured by the lamp, out of the six plug load devices. After the comparison of the different models, we can see that the LSTM predictions were better for day-ahead predictions. However, the LSTM model performance dropped significantly for the week-ahead predictions, and the comparison is illustrated in Figure 4 and Figure 5. On the contrary, GRU did not perform very well on the day-ahead predictions; however, it captured the future consumption patterns more competently in the week-ahead predictions. Furthermore, when comparisons are made between Bi-LSTM and GRU, Bi-LSTM is slightly better than GRU.

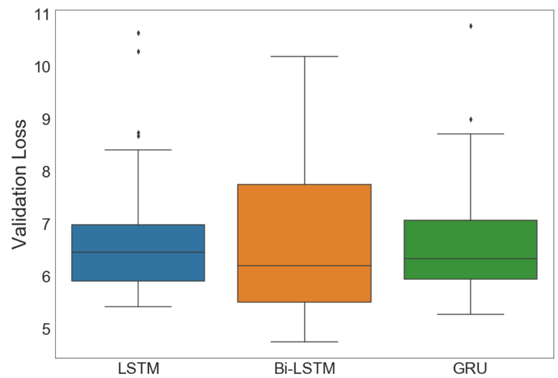

Figure 6: Validation loss [in kW] for the three different prediction models

9. Model Evaluation Results

In this section, we explicitly describe how we evaluated the model. As mentioned in section 6.4, the evaluation metrics used are RMSE and MAE. The RMSE is directly interpretable, making it a better measure of *goodness of fit* rather than using a correlation coefficient. Furthermore, the errors are squared before they are averaged, thus applying a relatively high weight to significant errors. In this case, RMSE is an appropriate metric since it provides the benefit of penalizing large errors. Table 4 highlights the average RMSE [in kW] from each plug load device for the different prediction models for the day-ahead and week-ahead predictions.

The RMSE values for the GRU model for the plug load devices were closer to the groundtruth, except miscellaneous plug load, see Figure 7 in subplot A and C, for comparison. Finally, we can interpret from the plots that the miscellaneous plug load has a vital role in the MEL prediction, and it can negatively impact the performance of the prediction models.

Another evaluation metric used is Mean Absolute Error (MAE) for evaluating the three different deep learning models. The Mean Absolute Error is the average of all absolute errors between actual and predicted values. Table 5 highlights the MAE values from each plug load device for the different prediction models for the day-ahead and week-ahead predictions. The MAE values for the GRU model mirrored the values from the RMSE evaluation, by being closer to the groundtruth except for the miscellaneous plug load. The comparison can be seen in Figure 7 between subplot B and D.

Table 4: Average RMSE Comparison [in kW] between the different prediction model on the plug- load devices [Bold indicates lowest RMSE out of the three prediction models for individual plug load devices]

Prediction	**Day- Ahead Prediction**			**Week-Ahead Prediction**		
Devices/Model	**LSTM**	**Bi-LSTM**	**GRU**	**LSTM**	**Bi-LSTM**	**GRU**
RMSE_Misc	0.7527	**0.3960**	1.3805	3.1515	**1.5172**	2.0673
RMSE_Dock	**0.0281**	0.0430	0.0413	0.1302	0.1387	**0.0874**
RMSE_Monitor1	**0.0741**	0.0760	**0.0749**	0.2388	0.2454	**0.2355**
RMSE_Monitor2	**0.0667**	0.0672	0.0674	**0.2116**	0.2189	0.2142
RMSE_Laptop	**0.0600**	0.0855	0.0969	0.3613	0.3843	**0.2860**
RMSE_Lamp	0.1020	0.0765	**0.0651**	0.2353	**0.2123**	0.3107

Table 5: Mean Absolute Error (MAE) Comparison [in kW] between the different prediction model on the plug load devices [Bold indicates lowest MAE out of the three prediction models for individual plug load devices]

Prediction	**Day- Ahead Prediction**			**Week-Ahead Prediction**		
Devices/Model	**LSTM**	**Bi-LSTM**	**GRU**	**LSTM**	**Bi-LSTM**	**GRU**
MAE_Misc	0.0078	**0.0041**	0.0144	0.3939	**0.1897**	0.2584
MAE_Dock	**0.0003**	0.0005	0.0005	0.0163	0.0173	**0.0109**
MAE_Monitor1	**0.0008**	0.0009	**0.0008**	0.0298	0.0307	**0.0294**
MAE_Monitor2	**0.0008**	**0.0008**	**0.0008**	**0.0265**	0.0274	0.0268
MAE_Laptop	**0.0008**	0.0009	0.0011	0.0452	0.0480	**0.0357**
MAE_Lamp	0.0011	0.0008	**0.0007**	0.0294	**0.0265**	0.0388

10. Discussion and Lessons Learned

A significant conclusion from this state-of-the-art overview is that user behavior has a significant impact on energy consumption and device utilization in commercial buildings, and institutional buildings reflect the same pattern.

The influence of user behavior is a challenge to quantify for methodological interpretation. The decision-making process of end-user is multi-factorial and com-

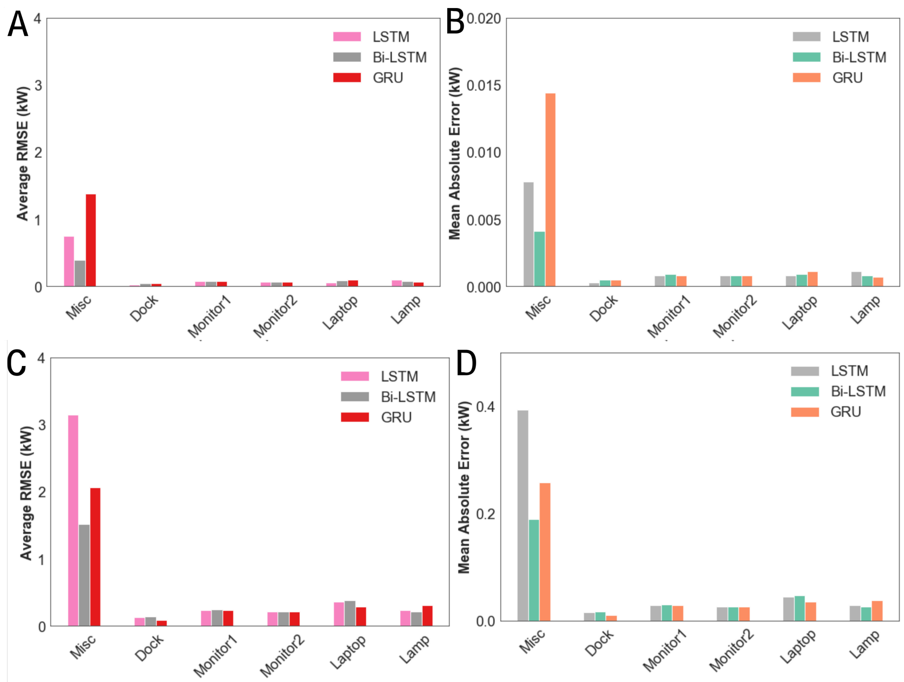

Figure 7: Average RMSE and MAE comparison [in kW] between the different prediction models: LSTM, Bi-LSTM and GRU from the six plug load devices for day-ahead predictions in subplot: A and B and week-ahead predictions in subplot: C and D.

.

plex; thus, factors influencing behavior are also numerous and varied. The dynamic nature of occupant's energy behavior is hard to comprehend, and multi-disciplinary approaches and meticulous investigations are required to contribute new insights into the energy-use domain.

To determine building occupant behaviors, a scientific study that interprets the dominant factors that are involved in energy behaviors has to be conducted with the users. Since the users do not always make rational decisions, the manner of presenting the choice itself becomes determinant in adopting energy-efficient behaviors. Also, the energy conservation measures introduced without taking into account user comfort and satisfaction can often have negative impacts and be counter-productive because users are likely to try to adapt to their environment to attain satisfactory conditions.

The prediction of energy-use complements the process of understanding and optimizing building energy consumption by providing valuable insights regarding the occupant's consumption patterns and device-utilization. These predictive models used in this research work become vital for implementing demand-response assessments, as well as providing pathways for valid pricing and tariffing for energy usage. Moreover, the benefits of such models can get amplified through the use of software tools to achieve these objectives.

11. Application and Future Research Directions

An essential application of this work is the light it sheds on the accuracy of MEL prediction using deep learning models. These estimates through software tools can be beneficial to building owners and facility managers in design and decision-making processes to recommend energy-saving control strategies and identifying energy footprints from each device. Another important application is the usage of these predictions to identify periods of peak loads for equipment to further recommend energy retrofit measures more efficiently.

Studies investigating the effect of feedback on user consumption patterns highlight that it has indeed been a successful strategy [40]. The results of this work are useful in highlighting the diversity of occupant energy-use patterns with regards to MELs, and illustrates the subsequent opportunities that arise for contributing to such potential feedback systems.

An example of inducing behavioral changes can be in the form of gamification approaches in offices/mixed-use building, wherein occupants are provided with incentives for adopting energy-saving measures. However, case studies in this regard are rare and warrant further research and evidence. Some potential applications of MEL predictions to incorporate energy-saving measures can be:

1. Developing occupant awareness programs, coupled with awards or financial incentives for reducing their energy usage.

2. Identifying the requirements of occupants through feedback systems and assess the trade-off between those requirements and actual energy usage.

3. Investigating heterogeneous sensing modalities to adopt the best metering technique for extrapolating such data.

For future studies, attention has to be paid to the interaction between the occupants' preferences and advanced building automation systems. Another aspect could be the comparison of predictions when inputs from the HVAC systems are available. There are still some open research questions and scope for further investigation about the design of such building control systems regarding occupant behaviors and preferences. One aspect is the synergy between the building control systems, with the occupant preferences regarding their comfort and energy usage. Another aspect is the flexibility or robustness of those control systems towards occupant behaviors, and how well they can adapt to those behaviors.

12. Conclusion

This work was successful in performing a case study focused on MEL prediction through deep learning methods. The data comprised of the MEL consumption of each occupant and the associated device-utilization. This data was used to develop a baseline LSTM model for MEL predictions, which was suitably efficient for predicting consumption patterns over a shorter period. Furthermore, the study compared the results from this baseline model with two state-of-the-art deep learning models (Bi-LSTM and GRU) and highlighted the evaluation results of each model. The experimental results on this new dataset collected demonstrate that considering device-utilization patterns and occupant interactions with these devices is fundamental for miscellaneous loads prediction.

In this paper, our comparative analysis results show that in terms of RMSE and MAE, all three deep learning models can achieve the best results depending on the considered devices and also the duration of the prediction required, i.e., short-term or long-term prediction. The low RMSE and MAE values indicate that the deviation of the predicted values from the measured values was quite low. Besides, Bi-LSTM proved to be the marginally more stable model out of those three, in both day-ahead and week-ahead predictions. Usually, due to GRUs less complicated structure and fewer gates, the training phase is faster and converges faster than other models. But, in our case, GRU and Bi-LSTM converge around the same number of epochs. As discussed in Section 10 and 11, the accuracy of these prediction models can be beneficial to building management systems and can enable the BMS authorities to perform intelligent energy management strategies.

References

[1] A. Kamilaris, B. Kalluri, S. Kondepudi, T. K. Wai, A literature survey on measuring energy usage for miscellaneous electric loads in offices and commercial buildings, Renewable and Sustainable Energy Reviews 34 (2014) 536–550.

[2] World energy outlook 2018 – analysis - iea, `https://www.iea.org/reports/world-energy-outlook-2018`, (Accessed on 04/03/2020).

[3] A. M. Syed, C. Hachem, Net-zero energy design and energy sharing potential of retail-greenhouse complex, Journal of Building Engineering 24 (2019) 100736.

[4] A. Afshari, C. Nikolopoulou, M. Martin, Life-cycle analysis of building retrofits at the urban scale—a case study in united arab emirates, Sustainability 6 (1) (2014) 453–473.

[5] D. Yan, T. Hong, B. Dong, A. Mahdavi, S. D'Oca, I. Gaetani, X. Feng, Iea ebc annex 66: Definition and simulation of occupant behavior in buildings, Energy and Buildings 156 (2017) 258–270.

[6] Y. Zhang, X. Bai, F. P. Mills, J. C. Pezzey, Rethinking the role of occupant behavior in building energy performance: A review, Energy and Buildings 172 (2018) 279–294.

[7] K. Roth, K. Mckenney, C. Paetsch, R. Ponoum, Us residential miscellaneous electric loads electricity consumption, in: ACEEE summer study on energy efficiency in buildings, 2008.

[8] B. E. D. Book, Us department of energy-office of energy efficiency and renewable energy (2010).

[9] G. Ghatikar, I. Cheung, S. Lanzisera, B. Wardell, M. Deshpande, J. Ugarkar, Miscellaneous and electronic loads energy efficiency opportunities for commercial buildings: a collaborative study by the united states and india, Tech. rep., Lawrence Berkeley National Lab.(LBNL), Berkeley, CA (United States) (2013).

[10] K. Raviraj, N. Gupta, H. N. Shet, Analysis of measures to improve energy performance of a commercial building by energy modeling, in: 2016 Online International Conference on Green Engineering and Technologies (IC-GET), IEEE, 2016, pp. 1–4.

[11] D. J. King, Residential dc power bus, ASHRAE Journal 52 (9) (2010) 73.

[12] J. M. Burgett, A. R. Chini, Using building and occupant characteristics to predict residential residual miscellaneous electrical loads: a comparison between an asset label and an occupant-based operational model for homes in florida, Journal of Building Performance Simulation 9 (1) (2016) 84–100.

[13] A. Mahdavi, F. Tahmasebi, Predictive models of electrical energy use in office buildings due to plug loads.

[14] M. R. Hargan, Ashrae standards committee 2001-2002.

[15] A. Mahdavi, F. Tahmasebi, M. Kayalar, Prediction of plug loads in office buildings: Simplified and probabilistic methods, Energy and Buildings 129 (2016) 322–329.

[16] Z. Wang, T. Hong, M. A. Piette, Data fusion in predicting internal heat gains for office buildings through a deep learning approach, Applied energy 240 (2019) 386–398.

[17] D. Yan, W. O'Brien, T. Hong, X. Feng, H. B. Gunay, F. Tahmasebi, A. Mahdavi, Occupant behavior modeling for building performance simulation: Current state and future challenges, Energy and Buildings 107 (2015) 264–278.

[18] A. C. Menezes, A. Cripps, R. A. Buswell, D. Bouchlaghem, Benchmarking small power energy consumption in office buildings in the united kingdom: A review of data published in cibse guide f, Building Services Engineering Research and Technology 34 (1) (2013) 73–86.

[19] A. Menezes, A. Cripps, R. A. Buswell, J. Wright, D. Bouchlaghem, Estimating the energy consumption and power demand of small power equipment in office buildings, Energy and Buildings 75 (2014) 199–209.

[20] H. B. Gunay, W. O'Brien, I. Beausoleil-Morrison, S. Gilani, Modeling plug-in equipment load patterns in private office spaces, Energy and Buildings 121 (2016) 234–249.

[21] D. L. Marino, K. Amarasinghe, M. Manic, Building energy load forecasting using deep neural networks, in: IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, IEEE, 2016, pp. 7046–7051.

[22] A. Rahman, V. Srikumar, A. D. Smith, Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks, Applied energy 212 (2018) 372–385.

[23] M. R. Arahal, A. Cepeda, E. F. Camacho, Input variable selection for forecasting models, IFAC Proceedings Volumes 35 (1) (2002) 463–468.

[24] C. Fan, F. Xiao, Y. Zhao, A short-term building cooling load prediction method using deep learning algorithms, Applied energy 195 (2017) 222–233.

[25] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[26] W. Lee, F. Yik, P. Jones, J. Burnett, Energy saving by realistic design data for commercial buildings in hong kong, Applied Energy 70 (1) (2001) 59–75.

[27] Z. Ren, G. Foliente, W.-Y. Chan, D. Chen, M. Ambrose, P. Paevere, A model for predicting household end-use energy consumption and greenhouse gas emissions in australia, International Journal of Sustainable Building Technology and Urban Development 4 (3) (2013) 210–228.

[28] L. Wang, R. Kubichek, X. Zhou, Adaptive learning based data-driven models for predicting hourly building energy use, Energy and Buildings 159 (2018) 454–461.

[29] M. Yalcintas, Energy-savings predictions for building-equipment retrofits, Energy and Buildings 40 (12) (2008) 2111–2120.

[30] M. Kottek, J. Grieser, C. Beck, B. Rudolf, F. Rubel, World map of the köppen-geiger climate classification updated, Meteorologische Zeitschrift 15 (3) (2006) 259–263.

[31] Home page - occupeye® | iot, https://www.occupeye.com/, (Accessed on 01/02/2020).

[32] Plugwise • plug, https://www.plugwise.com/en_US/products/plug?fbclid=IwAR3M8MVzhDjdzcATHEg2NijameXfjPAcLZOM-U58uOGQAAkJPs7sCKqXkvE, (Accessed on 01/03/2020).

[33] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional lstm networks, in: Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005., Vol. 4, IEEE, 2005, pp. 2047–2052.

[34] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Gated feedback recurrent neural networks, in: International Conference on Machine Learning, 2015, pp. 2067–2075.

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, The journal of machine learning research 15 (1) (2014) 1929–1958.

[36] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.

[37] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, arXiv preprint arXiv:1502.03167.

[38] B. Abushakra, A. Sreshthaputra, J. Haberl, D. Claridge, Compilation of diversity factors and schedules for energy and cooling load calculations, ashrae research project 1093-rp, final report.

[39] J. A. Davis III, D. W. Nutter, Occupancy diversity factors for common university building types, Energy and buildings 42 (9) (2010) 1543–1551.

[40] R. K. Jain, J. E. Taylor, P. J. Culligan, Investigating the impact eco-feedback information representation has on building occupant energy consumption behavior and savings, Energy and Buildings 64 (2013) 408–414.