

Inferring occupant counts from Wi-Fi data in buildings through machine learning

Zhe Wang, Tianzhen Hong*, Mary Ann Piette, Marco Pritoni
Building Technology and Urban Systems Division
Lawrence Berkeley National Laboratory

*Corresponding author: T. Hong, thong@lbl.gov, 1(510)4867082

Abstract

An important approach to curtail building energy consumption is to optimize building control based on occupancy information. Various studies proposed to estimate occupant counts through different approaches and sensors. However, high cost and privacy concerns remain as major barriers, restricting the practice of occupant count detection. In this study, we propose a novel method utilizing data from widely deployed Wi-Fi infrastructure to infer occupant counts through machine learning. Compared with the current indirect measurement methods, our method improves the performance of estimating people count: (1) we avoid privacy concerns by anonymizing and reshuffling the MAC addresses on a daily basis; (2) we adopted a heuristic feature engineer approach to cluster connected devices into different types based on their daily connection duration. We tested the method in an office building located in California. In an area with an average occupancy of 22-27 people and a peak occupancy of 48-74 people, the root square mean error on the test set is less than four people. The error is within two people counts for more than 70% of estimations, and less than six counts for more than 90% of estimations, indicating a relatively high accuracy. The major contribution of this study is proposing a novel and accurate approach to detect occupant counts in a non-intrusive way, i.e., utilizing existing Wi-Fi infrastructure in buildings without requiring the installation of extra hardware or sensors. The method we proposed is generic and could be applied to other commercial buildings to infer occupant counts for energy efficient building control.

Key words

Occupancy estimation; Occupant count; Wi-Fi data; Random Forest; Machine learning; Building control

1. Introduction

Buildings consume more than 40% of primary energy in the United States, United Kingdom, France, Germany; more than 30% in Japan; and more than 20% in China and India [1]. Reducing building energy usage is important to curtail fossil fuel consumption, reduce building operational costs, and enable affordability.

Energy in commercial and residential buildings is consumed to deliver services occupants need. However, because of the lack of occupancy information in current building control, buildings consume more energy than they need. For example, Studies on commercial buildings in the United States [2] and South Africa [3] found that more than half of the building energy was consumed during non-working hours. Occupancy information could not only be used to avoid energy waste but also to improve building energy efficiency [4], [5]. Typical applications using occupancy information to improve building energy efficiency include Demand Controlled Ventilation (DCV) [6], and Model Predictive Control (MPC) [7], [8]. In DCV, the fresh air supply volume was set based on indoor occupant counts. As a large amount of building energy is consumed to filter and condition the fresh outdoor air [9], DCV is effective to reduce building energy consumption. In MPC, the occupant counts could be used to predict internal heat gains, and accordingly to optimize HVAC control.

Melfi et al. defined four different resolution levels for occupancy information [10], as summarized in Table 1. Different resolution levels can serve different applications. The occupancy information could be used to reset the lighting and HVAC schedule, for example, lights can be turned off in unoccupied spaces, HVAC system or zone terminal equipment can be turned off or thermostat can be reset in unoccupied spaces. The occupant count information could be used for HVAC control, as in DCV [6] or model predictive control (MPC) [7], [8], since the equipment schedule and internal heat gains are correlated to the number of occupants. Additionally, occupant count is useful as the normalizing denominator in energy benchmarking, Measurement & Verification (M&V), and Fault Detection and Diagnosis (FDD) [11]. The identity and activity level information might be used to address the individual difference in thermal comfort preference [12] and to develop personalized thermal environment management [13]. By identifying who the occupants are and how they behave (e.g., clothing, activities) [14], [15], appropriate thermal environment (indoor temperature setpoint) could be provided to meet the diverse needs. Due to the wide application of occupant count for HVAC control and retrospective analysis, this study focuses on the resolution level of occupant count.

Table 1: Four resolution levels of occupancy information

Resolution level	Definition	Application
Occupancy status	Whether space is occupied or not	Lighting, HVAC schedule optimization
Occupant count	How many people are in a space	HVAC control optimization: DCV, MPC; Energy benchmarking, M&V and FDD
Identity	Who they are	Personalized thermal environment management
Activity	What they are doing	Personalized thermal environment management

Because of the substantial energy saving potential of utilizing occupancy information to optimize building control, several methods using a variety of sensors have been proposed to detect occupant counts in buildings. CO₂ concentration-based method leverages the law of mass conservation to infer indoor occupant count [16], [17], [18], but was challenged to be unable to timely reflect rapid changes of occupants [19]. Another widely used approach to detect occupant count is Radio Frequency (RF) based sensors, which are typically consisted of an antenna, a transceiver and a transponder. RF-based sensors could detect occupant count and location by sensing the electromagnetic signal reflected (so-called passive mode [20]) or emitted (so-called active mode [21]) from occupants. The third mainstream approach to detect occupant count is camera-based [22] sensors, which typically requires applying an image recognition algorithm to detect occupants from other objects. To protect privacy, infrared-based sensors [23] could also be used, which detect long-wave radiation rather than visible light emitted from occupants. To increase the field of view and sensitivity to occupant detection, Mikkilineni et al. (2019) proposed to use long-wave infra-red focal-plane arrays, which could be coupled with radio frequency and ultrasonic-based radar to improve accuracy [24]. The last but not least occupancy detection method uses smart meter data [25], [26], [27], leveraging the relations between occupant presence with building power consumption.

A common practice to improve the estimation accuracy is to ensemble the result from different estimators: either ensemble the estimators developed by different input variables, a.k.a data fusion [28] (using CO₂, temperature), [25] (using CO₂, sound level, power use), [26] (using CO₂, power use), [29] (using CO₂, humidity, temperature) or ensemble the estimators developed by different algorithms or with different hyper-parameters [30].

However, all the aforementioned occupant count detection approaches require installing additional sensors or hardware equipment, which leads to extra cost and labor [31]. With the wide deployment in almost every building nowadays, Wi-Fi infrastructure provides internet connections and thus offers a unique opportunity for virtual sensing of occupant count [32]. Multiple researchers have proposed methods to leverage Wi-Fi infrastructure to infer occupant count [33], [34], [35], [36], [37], [38]. Despite the rapid technology development and promising application potential, the reported methods using Wi-Fi data to infer occupant count have two limitations: (1) some technologies require installing extra apps on the Access Point or end-use devices [33], [34], [37], [38]; and (2) the other require recording the MAC addresses of connecting devices [35], [36], which would raise privacy concerns. For instance, Wang et al. applied location filter and MAC address filter to enhance detection accuracy, which needs to record the calibrated Received Signal Strength and MAC address [39]. Therefore, there still exists a research gap, demanding an accurate and non-intrusive approach to detect occupant count, i.e., using the existing information infrastructure in buildings and not requiring the installation of extra hardware or software packages [40].

In this study, we proposed a novel method to infer occupant count using the Wi-Fi connection counts through machine learning. By anonymizing and reshuffling the MAC addresses every day,

we avoid privacy concerns. By clustering connected devices into different types based on their daily connection duration, we improve the estimation accuracy. Wi-Fi data from an office building located in Berkeley California was used to test our approach, and the accuracy was compared with existing studies, demonstrating the reliability of this method.

2. Method to infer occupant counts

Figure 1 presents the major steps of the workflow to infer occupant counts, which will be described in detail in this section.

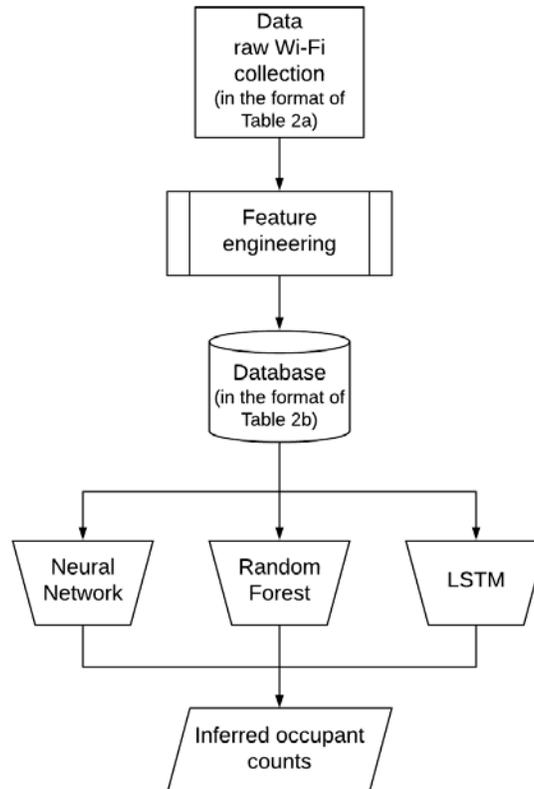


Figure 1: Work flow of this study

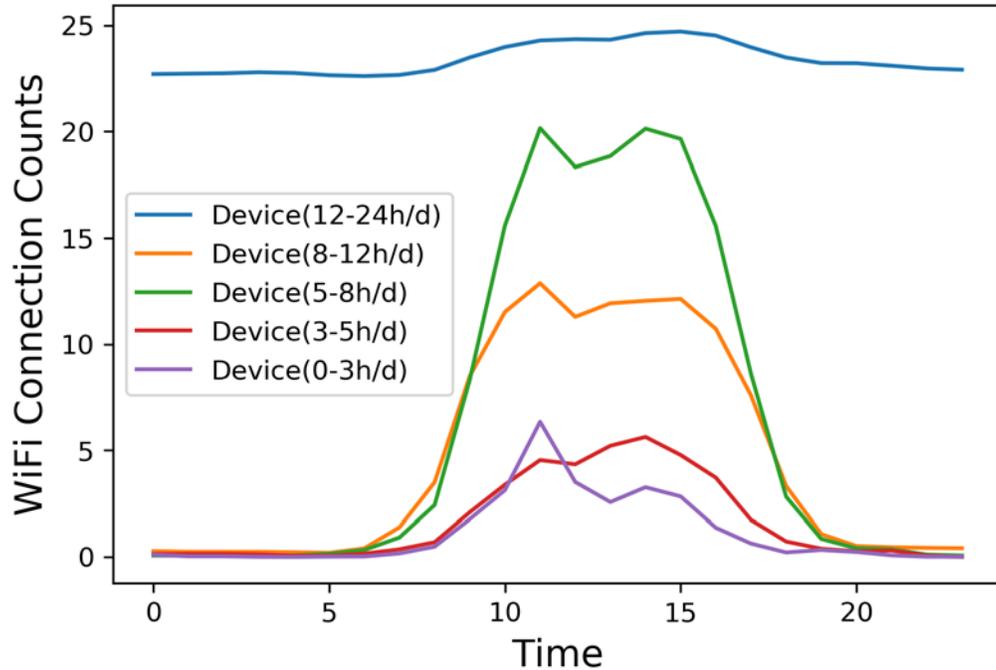
2.1 Feature engineering

In this section, we created and selected features first to improve the estimation accuracy. A major reason using Wi-Fi connection counts alone could not accurately infer occupant counts is the mapping relations between the number of connected devices and the number of occupants are not consistent and might change temporally and spatially. As shown in Figure 2(a), there are different types of Wi-Fi connection devices, which belong to different types of owners, subject to different mapping rules of Wi-Fi connection counts and occupant counts. There are some devices connected to Wi-Fi almost throughout the whole day. Those devices are more likely office equipment or appliances such as printers or 24-hours-on computers or servers, or belong to occupants who never turn off the devices even though they are not present. In either case, those long-term connected devices might not be very informative to infer the occupancy variability. The second type of devices is connected with Wi-Fi for a relatively long period of time, which probably belong to long-term inhabitants like office workers, who averagely has

two devices connected while present - one is the cellphone and the other is the computer. The third type of devices is short-term connected with Wi-Fi, probably between one and three hours per day. Those devices might belong to short-term visitors, showing up for conferences or meetings, who usually only have one device (cellphone) connected. The last type of devices only connect to the Wi-Fi AP for a very limited period of time (less than one hour), which are highly likely to belong to occupants passing by the target area.

Type of devices	Type of owners	Mapping rules of Wi-Fi connection counts and occupant counts
 <i>Always connected</i>	 <i>Office appliances</i>	<i>Could not be used to infer occupant counts</i>
 <i>Long-term connected</i>	 <i>Inhabitants</i>	<i>Each occupant averagely has two devices (cellphone and computer) connected with Wi-Fi</i>
 <i>Short-term connected</i>	 <i>Visitors</i>	<i>Each occupant averagely each has one device (cellphone) connected with Wi-Fi</i>
 <i>Occasionally connected</i>	 <i>Passerby</i>	<i>Does not locate in the target area, should not be counted</i>

(a) mapping relation of devices, owners and Wi-Fi usage behaviors



(b) hourly variation of different type of devices in the target area
 Figure 2: Different types of devices with Wi-Fi connections

Since devices with different connection periods have different mapping relations between the Wi-Fi connection counts and occupant counts (for instance, two devices per occupant vs. one device per occupant), it is reasonable to assume the estimation accuracy could be improved if we could differentiate various types of devices based on their daily connection duration, and use that information into the machine learning algorithm. Therefore, a major innovation of this study is: rather than input one variable of the total number of connected devices into the algorithm, we input multiple variables into the algorithm to enhance accuracy, representing the number of different types of connected devices, from short-term connected to long-term connected.

As the input variables for occupant count inference only include the number of Wi-Fi connected devices for each device type (long-term connected or short-term connected), this approach does not require to record the MAC address of connected devices as [35], [36] did, which could help protect users' privacy.

2.2 Machine Learning Algorithms

As an exploratory study, we applied and compared three different machine learning algorithms to infer occupant counts with Wi-Fi data.

2.2.1 Random forest

Random forest is an ensemble learning method constituted of multiple decision trees. Random forest is a widely used machine learning algorithm for three major merits. First, the overfitting problem could be avoided by randomly selecting a subset of features to constitute the individual tree in the random forest [41]. Second, random forest is easy to use, without time-

consuming hyper-parameter tuning process. Third, random forest is a flexible algorithm that could be used for both regression and classification tasks.

2.2.2 Deep learning neural network

The artificial neural network (ANN) is a biologically-inspired machine learning algorithm that mimics how human brains function. The neural network is constituted of three layers of neurons: the input layer, hidden layer and output layer. The deep learning neural network advances ANN by adding multiple hidden layers to extract different features and to learn complicated non-linear relations.

2.2.3 Long term short term memory networks (LSTMs)

A key characteristic of time series data is the existence of time-dependence, for instance, what happened at the timestamp (t-k) might influence the value at timestamp t. To capture this time-dependence, the recurrent neural network has been proposed which takes the input of value at timestamp (t-1, t-2 ... t-n) to predict the value at timestamp t. However, with the increasing of n, more memory space and computation capability is demanded. What makes things worse, the vanishing gradient problem would be triggered, i.e., the sensitivity decays exponentially over time when n is large. To solve this problem, LSTMs, as a special form of deep learning, was proposed, and proved to be very useful for inferring and predicting time-series data [42]. By inputting the data from the current (t) and previous (t-1, t-2 ... t-n) time-stamps into the estimator, long-term time-dependencies of time-series data could be captured. LSTMs is widely used in speech recognition and other time-series data analysis.

It could be observed that the algorithm complexity increases from the Random Forest to LSTMs. In the next section, we will compare not only the estimation accuracy but also the computation complexity by looking at the CPU running time of the three algorithms.

2.3 Assessment metrics

Two assessment metrics are used in this study to compare the estimation accuracy of different methods of inferring occupant counts with Wi-Fi connection counts.

2.3.1 Root Mean Square Error (RMSE)

RMSE is defined in Equation 1 where n is the sample size, y_n is the measured value, and \bar{y}_n is the predicted value. As a two-norm error, which has the same unit as the measured value, RMSE is widely used in accuracy comparison.

$$RMSE = \sqrt{\frac{\sum_1^n (\bar{y}_n - y_n)^2}{n}} \quad \text{Equation 1}$$

2.3.2 X-tolerance accuracy

Considering the fact that in practical building control and operation, an error of one or two occupants would not lead to real difference especially in a space with dozens of occupants, Jiang et al. proposed the metrics of X-tolerance accuracy, defined in Equation 2 as the percentage of the estimations whose errors are less than X [18].

$$Acc(x) = \frac{\sum_1^M 1(|\bar{y}_n - y_n| \leq x)}{M} \quad \text{Equation 2}$$

3. Testbed and data collection

3.1 Testbed

3.1.1 The Case Building

The third and fourth floor of a four-story office building located in Berkeley, California was selected as the testbed for this study. We focused on the south end of the two floors, which have private and cubicle offices, with a floor area of around 800 m² on each floor. There are seven and nine Wi-Fi Access Points (AP) installed in the south end of the third and fourth floor, respectively. As can be seen from Figure 3, some Wi-Fi APs locate very close to the border of the target area, especially on the fourth floor. Therefore, it is possible that someone outside the target area connecting their devices to those Wi-Fi APs, which would unavoidably result in estimation errors.

The data collection period of this study is from late May to early July of 2018. The occupant counts and Wi-Fi connection counts data were collected every one minute and every ten minutes, respectively. Considering the time-step of HVAC control, both the occupant counts and Wi-Fi connection counts are down-sampled and averaged for every 30 minutes.

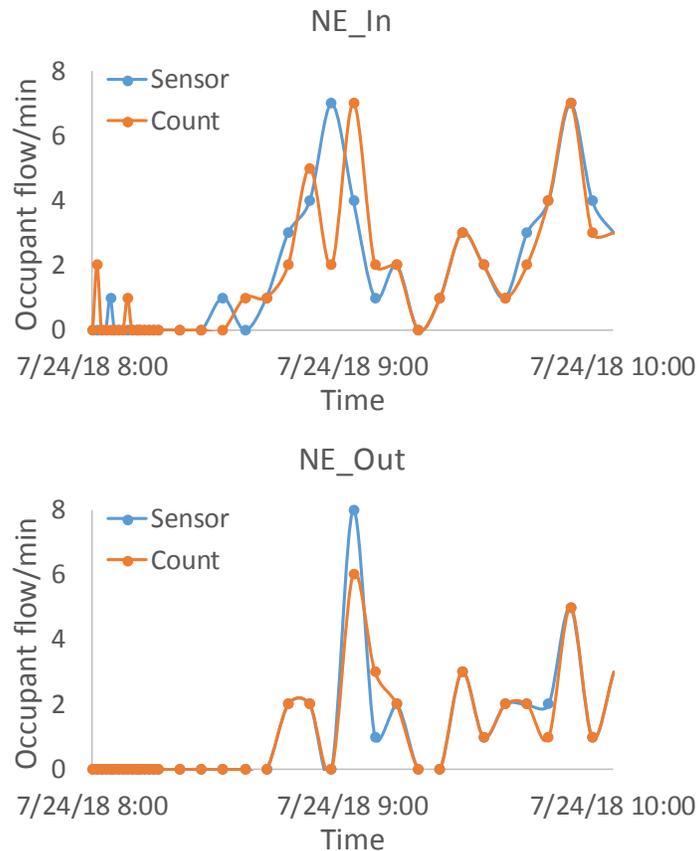


Figure 3: Floor plan and sensor locations of the case building

3.1.2 Ground truth occupancy data

To collect the ground truth data, three camera-based occupant count sensors manufactured by the TRAF-SYS company¹ were installed in the three entrances of the target zone of each floor. The camera-based sensor could detect the number of people entering and leaving the space. Integrating the net flow of people entering the border could inform the number of occupants in the target area.

The occupant counts measured by the camera-based sensors would serve as the ground truth data. To validate the measurement accuracy of occupant sensors, we sent a crew of researchers to the three entrances of the third floor between 8 and 10 AM, a typical period of people arriving in office, to manually count the net number of people through each entrance. We compared the people count measured by the occupant sensors and manually counted by the researchers. Figure 4(a) plotted the occupants flow from the three entrances we observed and the integral of the occupant flow, which is the accumulated indoor occupant count. Integrating the net occupant flow could get the indoor occupant counts, which was presented in Figure 4(b). It was confirmed that the measurement error of the camera-based occupant sensors is 8%², and the cumulative error is 9%³.

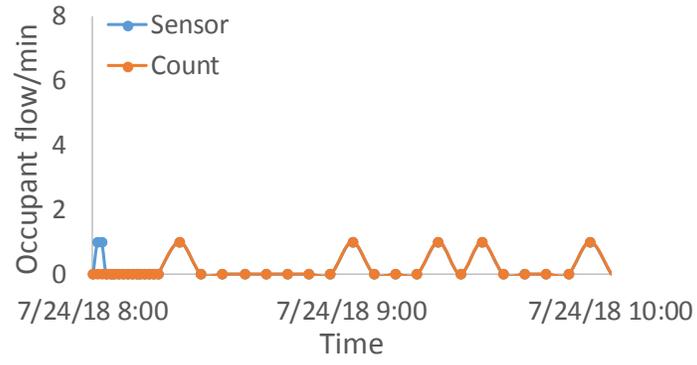


¹ <https://www.trafsys.com/>

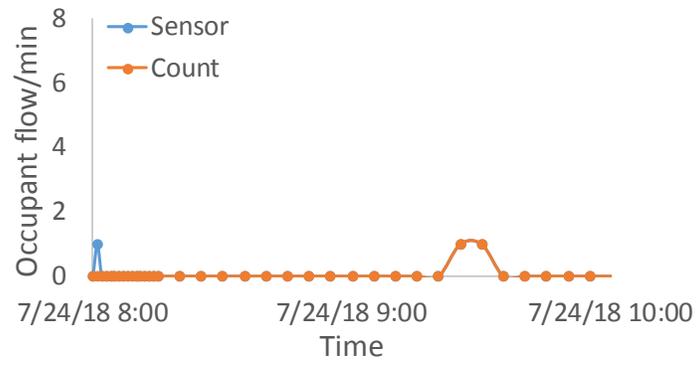
² Defined as: $\sum_t (\#(\text{sensed occupant flow})_t - \#(\text{actual occupant flow})_t) / \sum_t \#(\text{actual occupant flow})_t$

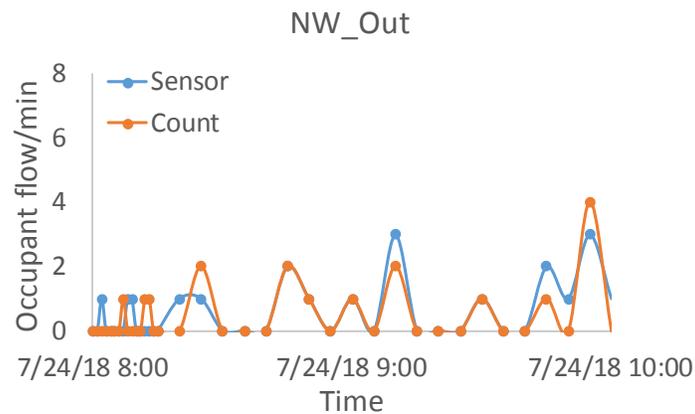
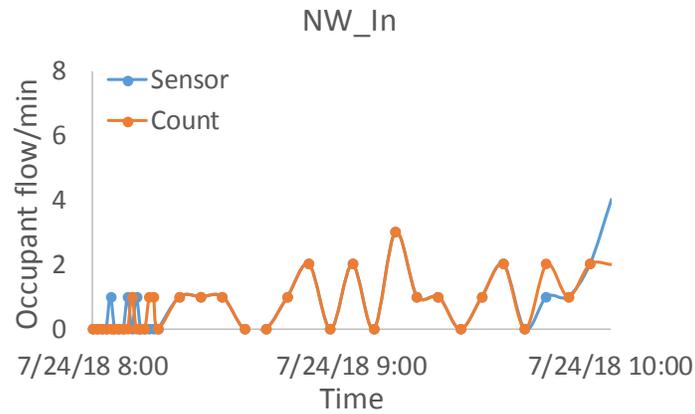
³ Defined as: $\sum_t (\#(\text{sensed occupant})_t - \#(\text{actual occupant})_t) / \sum_t \#(\text{actual occupant})_t$

SE_In

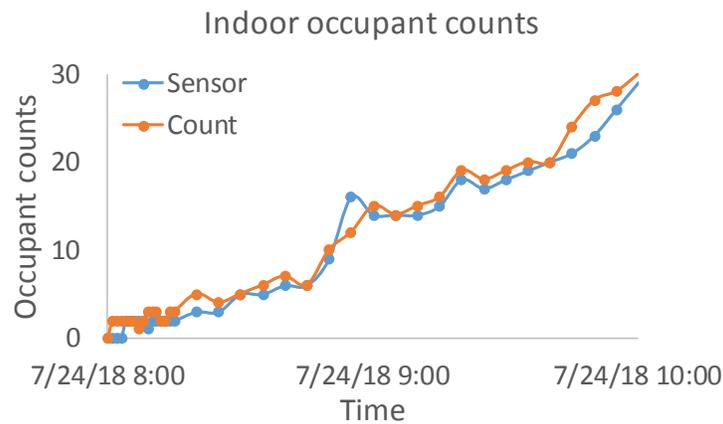


SE_Out





(a) Measurement error



(b) Cumulative error

Figure 4: Camera-based sensor calibration: orange line for the sensor measured values, the blue line for the manual count values

Due to the sensor errors, the total number of people entering the space during one day might not equal to the total number of people leaving the space on the same day. If this error were not corrected, the accumulated errors might be substantial after a period of time. Furthermore, the sensor measurement error might lead to a negative number of people in the space. To deal

with these two types of errors, we processed the data using a script summarized by the pseudo code in Appendix A to clean and calibrate the ground truth data on a daily basis.

3.1.3 Wi-Fi data

Table 2(a) presents a snapshot of the collected Wi-Fi data in this study, which include three columns: the timestamp of the recording, the ID of connected devices, the ID of Wi-Fi AP to which the device is connected. To protect privacy, the device IDs were randomly shuffled on a daily basis.

As we discussed in Section 2.1, theoretically, the method we proposed does not need MAC address. What needs to be input into the algorithm is just the number of connected devices for each device type, as shown in Table 2(b). The device type could be easily determined by looking at the duration that a device is connected to the Wi-Fi on the previous day of the same type (working or non-working). For instance, if today is Monday, the device type could be determined by checking last Friday’s connection duration of the same device. In this study, the data-preprocessing from the raw data (in the form of Table 2a) to the data needed by the algorithm (in the form of Table 2b) was done by researchers, to reduce the IT staff’s workload. In practice, the process of converting data from Table 2a to Table 2b could be done automatically by writing a script. So, the output information is only the device count, without MAC address, to protect users’ privacy.

Table 2: A snapshot of Wi-Fi data
(a) the data used in this study

Time	Device_ID	AP_ID
...		
20180521_0000	dfd6bafb68c1cd1f1e2d9190ca9d55f0	ap135-4206w
20180521_0000	e6c1fe930c6d2c2f2e2d9d69fc0abeda	ap135-3103
...		
20180521_0000	dd464552ecc1208e94a955bffee1f749	ap135-4110
20180521_0010	dfd6bafb68c1cd1f1e2d9190ca9d55f0	ap135-4206w
20180521_0010	e6c1fe930c6d2c2f2e2d9d69fc0abeda	ap135-3103
...		

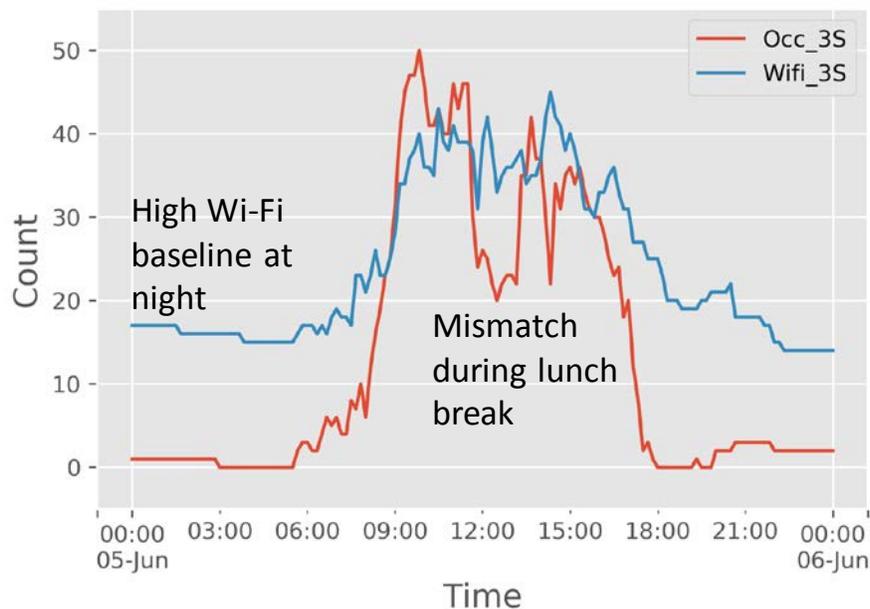
(b) the data input to the ML algorithm

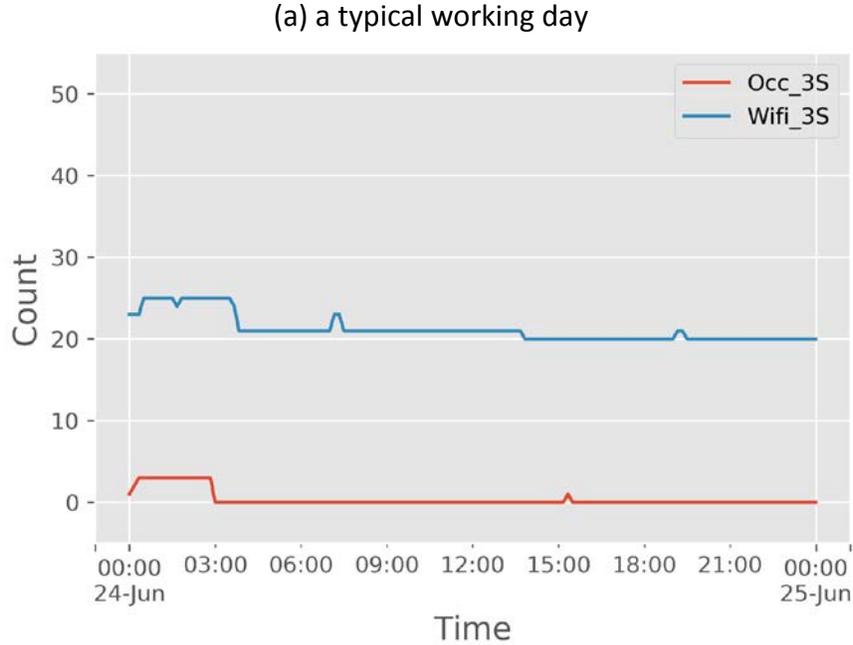
Time	Target zone	Device_type	Device_count
...			
20180521_0000	Zone 1	Short term (less than 1h per day)	0
...			
20180521_0000	Zone 1	Long term (more than 12h per day)	20
20180521_0000	Zone 2	Short term (less than 1h per day)	0
...			
20180521_0000	Zone 2	Long term (more than 12h per day)	15
20180521_0010	Zone 1	Short term (less than 1h per day)	0

3.2 Data collection and exploration

3.2.1 Typical working and non-working days

With the sensing infrastructure described in the previous section, we collected the number of occupant counts and the number of Wi-Fi connection counts. Figure 5 presented the measurement of a typical working and non-working day. Generally speaking, the occupant counts and Wi-Fi connection counts follow similar trends, starting to rise at around 8:00 AM, dropping around the mid-day for lunch break, and starting to decrease at 16:00 (4:00 PM). However, the variation of Wi-Fi connection counts is not as significant as that of occupant counts. This might be due to people leave their devices connected with the Wi-Fi for short-term leaves. For instance, people might not turn off their computer during lunch break. Therefore the decrease in the number of connected devices might not be as marked as the decrease in occupant counts. During non-working hours, there are around 20 devices connected with Wi-Fi, which might be standby office equipment (e.g., printers, computers). The relatively high proportion of office equipment standby during non-working hours (20 out of the peak of 45) indicates an opportunity to conserve energy by encouraging people to turn off the office equipment before leaving the office for a day.





(b) a typical non-working day

Figure 5: the number of occupant counts (red line) and the number of Wi-Fi connection counts (blue line) on the 3rd floor for a typical working (left) and non-working (right) day

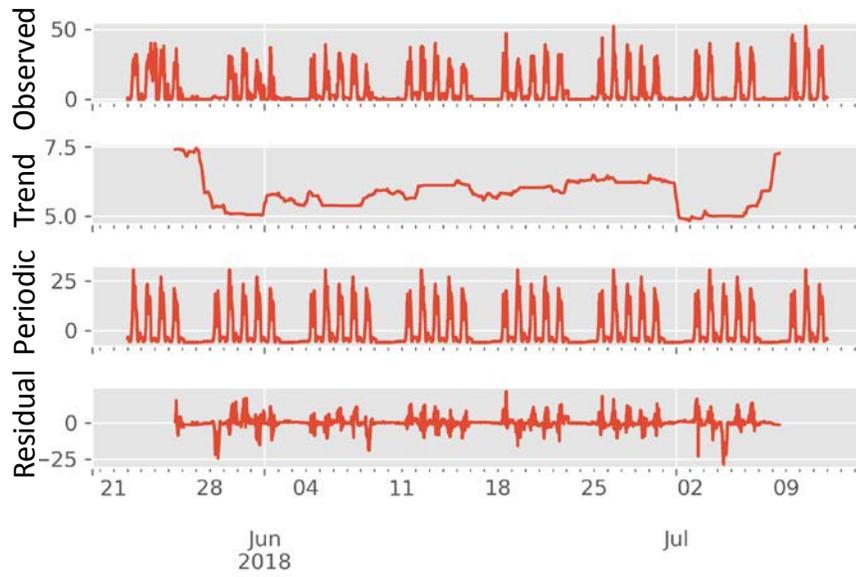
3.2.2 Time-series decomposition

A widely used approach to study time-series data is to decompose it into the trend component, the periodic component, and the residual component, as shown in Equation 3 [43]. Where y_t is the observed value at time t . T_t is the trend component at time t , reflecting the long-term progression of the series. T_t is the moving average calculated from Equation 4, where k represents the length of half a period. In this case, the length of a period is one week⁴. P_t is the periodic component at time t , reflecting the periodic fluctuation. P_t is calculated by averaging the detrended time-series value at the same time of each week. R_t is the residual component at time t , reflecting random, irregular changes. R_t is calculated by subtracting the estimated trend and periodical components from the raw data.

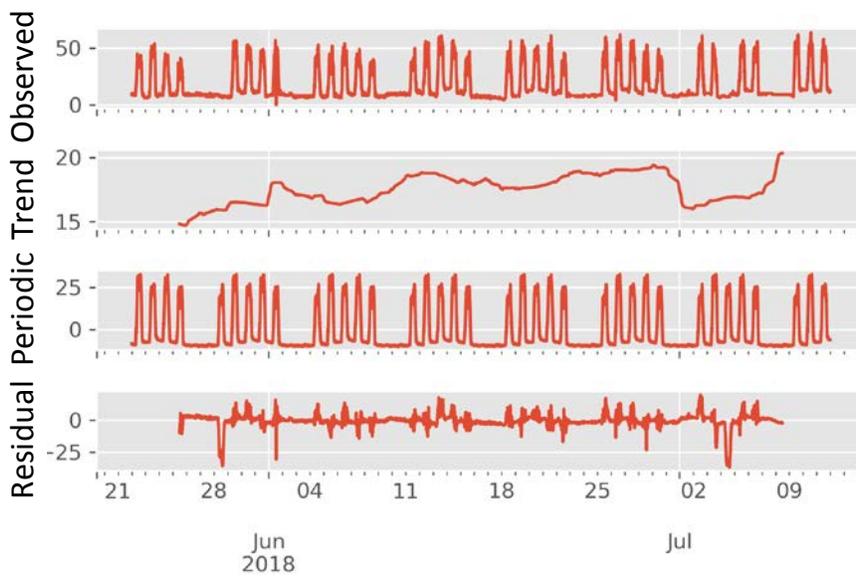
$$y_t = T_t + P_t + R_t \quad \text{Equation 3}$$

$$T_t = \frac{1}{2k+1} \sum_{j=-k}^k y_{t+j} \quad \text{Equation 4}$$

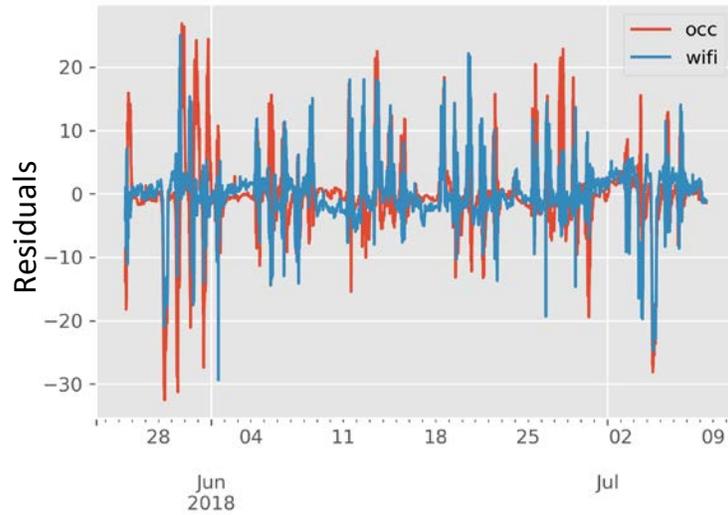
⁴ Other periods could be chosen like daily or monthly. In this study, we studied weekly patterns considering the temporal length of our data



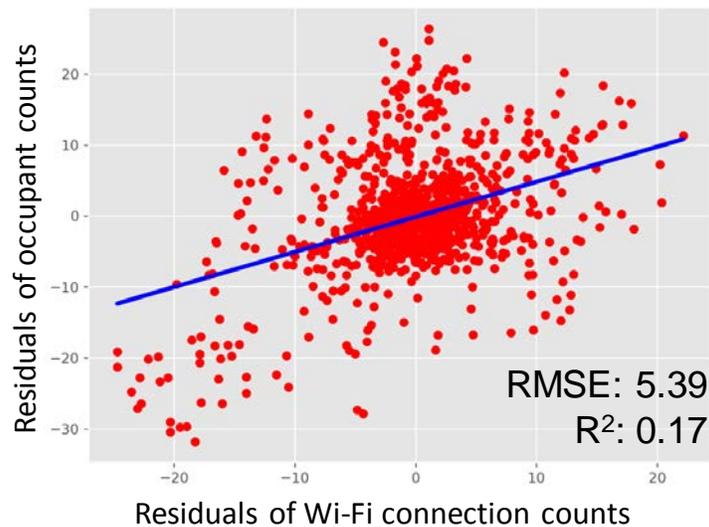
(a) Decomposition of occupant counts



(b) Decomposition of Wi-Fi connection counts



(c) Time-series of the residual component



(d) Linear regression of the residual component

Figure 6: Decomposition analysis of the data from the third floor

As observed from Figure 6(a) and Figure 6(b), both the occupant counts and Wi-Fi connection counts are highly randomly fluctuated. The irregular fluctuations (reflected by the residual) of occupant and Wi-Fi connection counts are at the scale of 25, which is comparable to the magnitude of regular component (reflected by the trend and periodic), which increases the difficulty of using Wi-Fi data to infer occupant counts. What makes things worse is the irregular fluctuations of occupant counts could not be predicted by the irregular fluctuations of the Wi-Fi connection counts. As shown in Figure 6(c), the residual component of the occupant counts and Wi-Fi connection counts are well aligned during some period of time, for instance, 4th July, when both the observed occupant and Wi-Fi connection counts are below average as it is a national holiday. However, during other period of time, the residual components of occupant and Wi-Fi connection counts are not at the same pace, for instance, a positive occupant count residual and a negative Wi-Fi connection count residual (3rd July). Because of this mismatch, the

R-squared value (coefficient of determination) of occupant and Wi-Fi connection counts is as low as 0.17 between the occupant counts and Wi-Fi connection counts. Because of this complex behavior, as Yang et al. pointed out [31], simply using Wi-Fi connection count could not achieve an accurate occupant count estimation. Feature engineering is needed for a highly accurate occupant count estimator.

4. Results and Discussion

As we introduced in Section 2, three algorithms - Neural Network, Random Forest, and LSTM - were applied to infer occupant counts from Wi-Fi connection data. In addition to the algorithm, the selection of hyper-parameters also influences the inference performance. After hyper-parameter tuning, the following set of hyper-parameters were chosen in this study, as shown in Table 3. In this study, we implemented Random Forest with the open-source Python library scikit-learn v0.20.3⁵, implemented Deep Neuron Network and LSTM with the open-source Python-based machine learning programming platform Keras⁶. The default values of hyper-parameters, listed in the library documentation, were used if not specified in Table 3.

Table 3 Hyper-parameter settings

	Parameter	Definition	Value
Random Forest	n	The number of trees in the forest	1000
	$loss$	Loss function	Mean Square Error
Deep Neural Network	l	Number of hidden layers	3
	$n^{[1]}$	Number of units in the first hidden layer	200
	$n^{[2]}$	Number of units in the second hidden layer	100
	$n^{[3]}$	Number of units in the third hidden layer	50
	$g()$	Activation function	relu
LSTM	n	Number of neurons in the hidden unit	50
	k	Number of epochs	300
	t	Time windows for input variables	24 hours
	$loss$	Loss function	Mean Square Error
	opt	Optimization method	Adam

In this study, two types of parameters, i.e., time-related, Wifi connection count, were fed into the algorithm to infer the occupant counts, as illustrated in Table 4.

Table 4: Input variables to machine learning algorithms

	Sub-category	Examples
Time-related	Day type	<i>Holiday</i> (Boolean): yes or not
	Day of week	<i>Mon., Tues., ...</i>
	Hour of day	<i>0AM, 1AM, ...</i>
Wi-Fi connection		<i>Wifi_1h</i> : number of devices connected to Wifi less than 1 hour a day;

⁵ <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>

⁶ <https://keras.io>

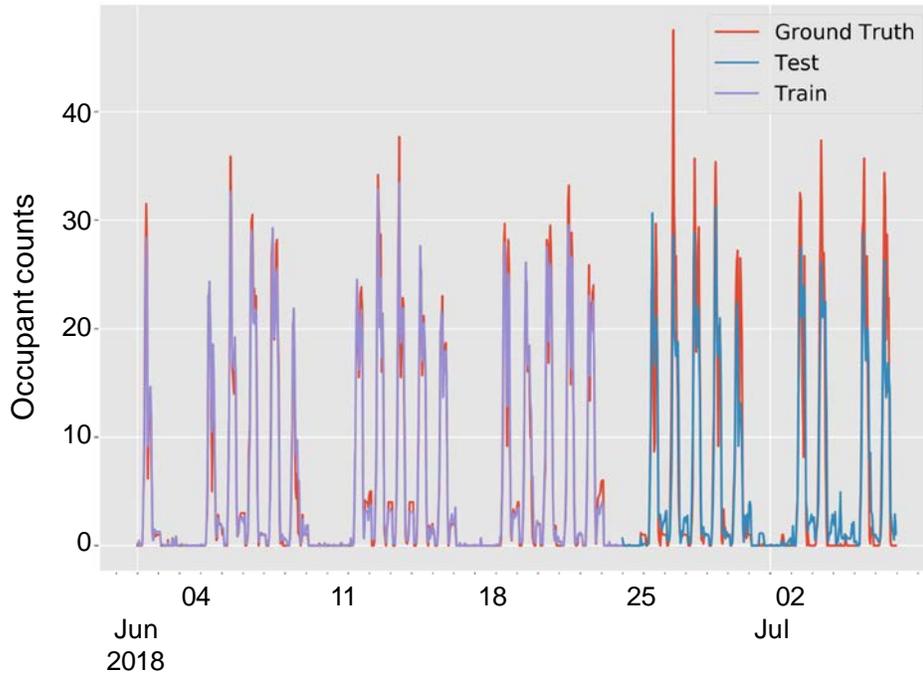
count	<i>Wifi_1-2h</i> : number of devices connected to Wifi between 1 hour and 2 hours a day; ...
-------	---

4.1 Estimation accuracy

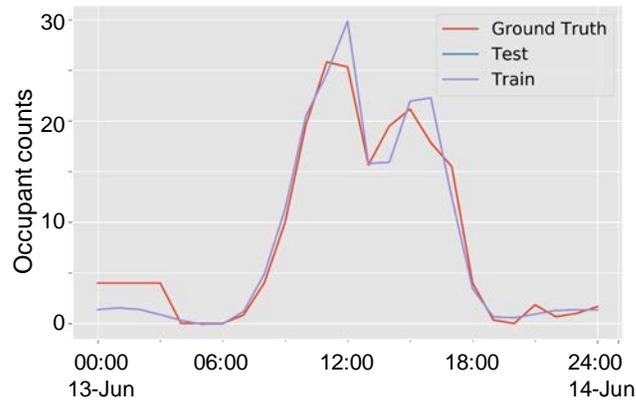
The whole dataset was split into the training and testing sets: the first three weeks serve as the training set (purple) and the last two weeks as the test set (blue). Figure 7 plotted and compared the estimated values with the ground truth data (red). To avoid redundancies, the time-series plots of a typical day and X-tolerance accuracy plots were only provided for the random forest algorithm, but the comparison between the three methods will be illustrated in Table 5.

The red line in Figure 7 represents the actual occupant counts on this specific floor. The random variation of occupant counts is more significant than we expected. For instance, the weekly peak occupants happened on Tuesday in the Week starting from 4th June, on Wednesday in the Week starting from 11th June, while on Thursday in the week starting from 18th June. The time-series decomposition result presented in Figure 6 (a) showed that the random variation (excluding the trend and periodic component) is between -25 to +20, almost at the same scale with the periodic variations. The reasons behind the larger random variation might include irregular special events such as seminars, the increasing popularity of working from home, etc. Therefore, we could not solely rely on the pre-determined occupancy schedule to estimate occupant counts. Instead, we need to input other features (Wi-Fi in this case) and to leverage machine learning algorithms to infer occupant counts.

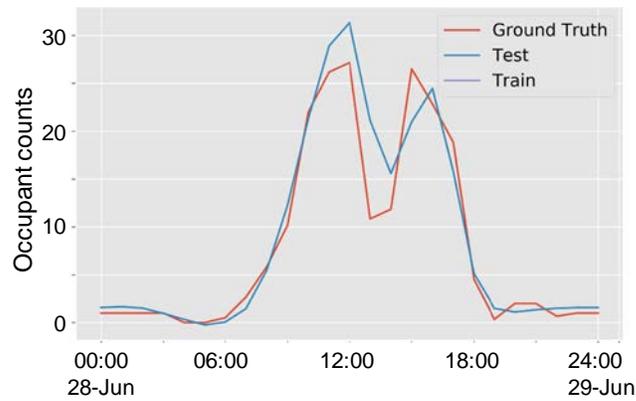
The general trend could be captured in all three estimators. The estimation error is within two occupant counts for more than 70% of the estimations, and within six occupant counts for more than 90% of the estimations. Considering the average occupant counts during working hours is 27 on the third floor and 22 on the fourth floor, with a peak value of 74 and 48, respectively, this estimation error is acceptable for HVAC control.



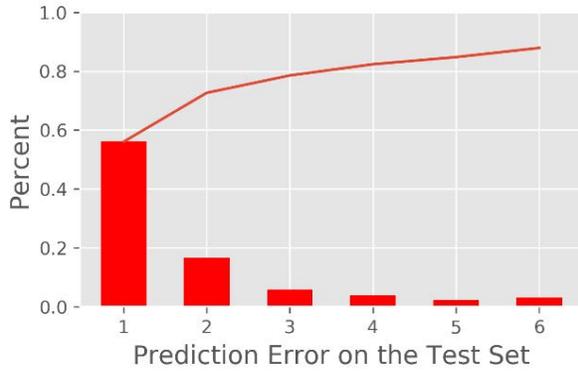
(a1) Random Forest: the whole data collection period on the 4th Floor



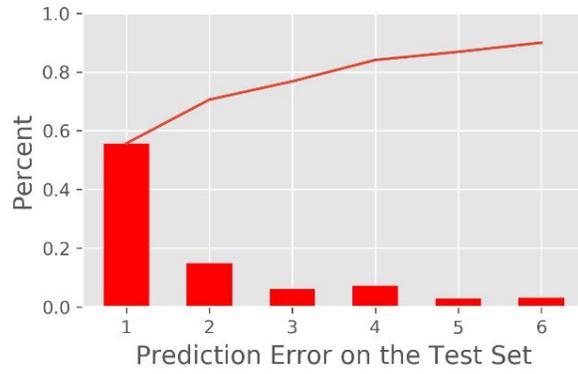
(a2) RF: a typical train day on the 4th Floor



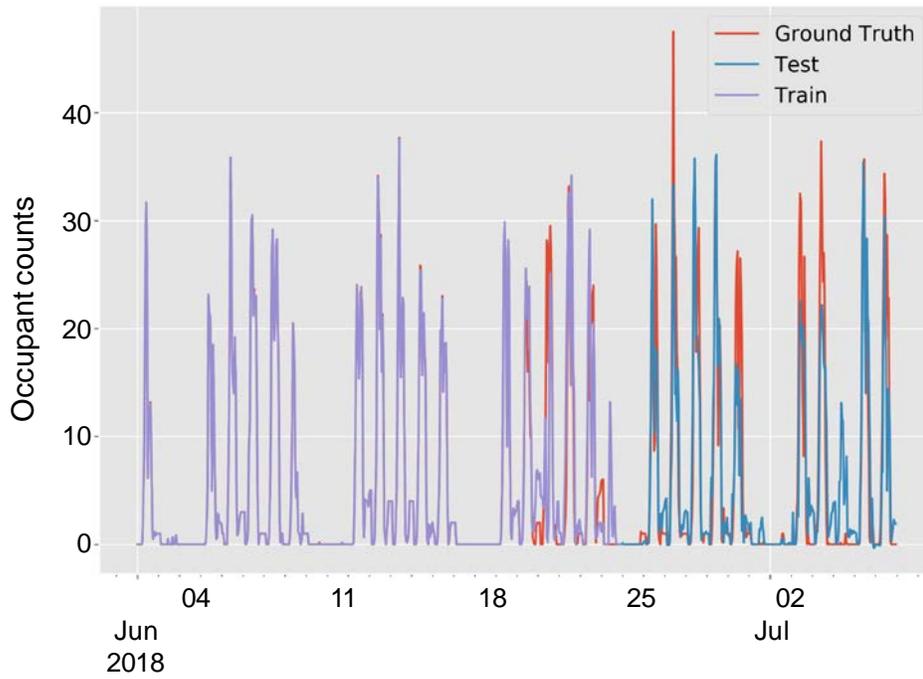
(a3) RF: a typical test day on the 4th Floor



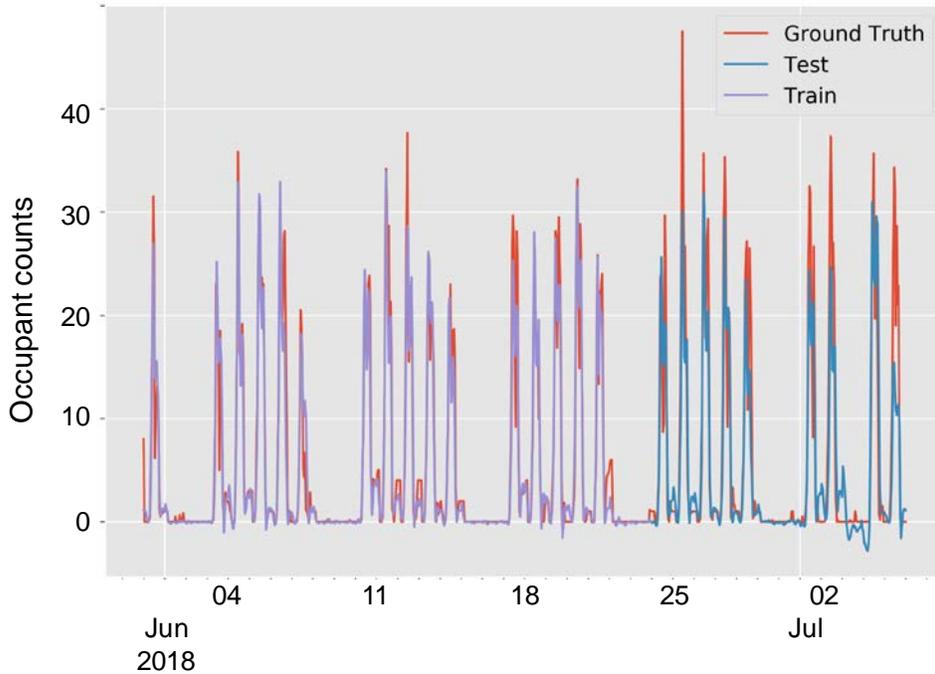
(a4) RF: X-tolerance accuracy on the 3rd Floor



(a5) RF: X-tolerance accuracy on the 4th Floor



(b) Deep neural network: the whole data collection period on the 4th Floor



(c) LSTMs: the whole data collection period on the 4th Floor

Figure 7: Results of occupant count estimation using the three machine learning algorithms

Another observation from Figure 7 is the peak occupancy could not be accurately estimated by either of the three algorithms. By revisiting the data, it was found when the peak event occurred, the Wi-Fi connection counts did not increase as markedly as the occupant counts increased, leading to an under-estimation of the peak occupancy. A possible explanation for this phenomenon is the peak occupancy occurred when a seminar was held and lots of people from other parts of the building or even other buildings came to the target area. A substantial proportion of seminar attendees might not use or connect their Wi-Fi devices during the seminar to stay focused. One possible solution to this problem is to introduce and use new event-related features to reflect the occurrence of seminars or conferences.

Table 5 and Figure 8 compared the RMSE and computation time of the three algorithms from two dimensions, the inference error and the computation time. The algorithm is considered to be better if it has a smaller error and consumes less time, as indicated by the green arrow in Figure 8. It could be observed that *the Random Forest* provides accurate occupant counts estimation from Wi-Fi data with the least computation time, indicating that more complicated algorithms might not necessarily outperform simple ones for this study.

Table 5: Comparison of three algorithms

	Random Forest (RF)	Neural Network (NN)	LSTM
RMSE on the training set	1.20	2.63	2.21
RMSE on the testing set	3.95	4.62	4.52

Computation time ⁷	2.38s	24.86s	65.61s
-------------------------------	-------	--------	--------



Figure 8: Comparison of three algorithms⁸

Table 6 compares the 1- and 2-tolerance accuracy of the method proposed in this study with previous studies using different office buildings (i.e., different dataset). The peak and average occupant counts were also presented in the table since an estimation error of two in a space with 25 people is more acceptable compared with the same estimation error in another space with only 15 people.

It can be seen that the estimations from our method delivered a higher accuracy prediction than methods proposed by previous research, except for the estimator applying Feature Scaled Extreme Learning Machine on smoothed CO₂ concentration data documented in Jiang et al. (2016) [18]. However, as the authors pointed out, smoothing algorithms require the data to be measured. Either locally or globally smoothed CO₂ concentration could not be obtained in a real-time manner. Therefore, this method could only be used in retrospective analysis but not to estimate real-time occupant counts for building control purpose. Additionally, as discussed in the Introduction Section, Jiang et al. (2016)'s method requires installing CO₂ sensors in the target areas. CO₂ sensors require to be calibrated periodically. Installing additional sensors would lead to extra economic and labor costs.

It is acknowledged that different office spaces might not be comparable to each other. For example, offices with more visitors and seminars would be more challenging for using Wi-Fi connection counts to infer occupant counts. Therefore, the comparison shown in Table 6 does not necessarily mean our method is superior to others, but rather serves as a proof that the method proposed in this study could be used to infer occupant counts.

Table 6: Accuracy comparison of Random Forest with previous studies⁹

⁷ On a Dell desktop with 4-Core Intel Xeon CPU E5-1630 v4 @ 3.70GHz

⁸ To calculate relative inference error, the Root Mean Square Error (RMSE) is normalized by the peak occupant counts, which is 48 in this case.

		Data source	Method	1-tolerance accuracy	2-tolerance accuracy	Peak occupant counts	Average occupant counts
This study	Floor 3 training	Wi-Fi	Random Forest	72%	85%	74	27
	Floor 3 testing	Wi-Fi	Random Forest	57%	72%	74	27
	Floor 4 training	Wi-Fi	Random Forest	70%	84%	48	22
	Floor 4 testing	Wi-Fi	Random Forest	56%	70%	48	22
Jiang et al. (2016) [18]	Measured CO ₂	Standard Extreme Learning Machine	45%	54%	26	15	
	Measured CO ₂	Feature Scaled Extreme Learning Machine	55%	68%	26	15	
	Globally smoothed CO ₂	Feature Scaled Extreme Learning Machine	71%	86%	26	15	
	Locally smoothed CO ₂	Feature Scaled Extreme Learning Machine	68%	80%	26	15	
Wang et al. (2017) [35]	Wi-Fi	Linear Regression	30%	50%	25	15	
	Wi-Fi	Support Vector Machine	25%	45%	25	15	
	Wi-Fi	Auto-Regressive Moving Average	30%	55%	25	15	
	Wi-Fi	Dynamic Markov Time-Window Inference	38%	60%	25	15	
Wang et al. (2018) [36]	CO ₂	Mass conservation	35%	50%	19	11	
	Wi-Fi	Markov	40%	60%	19	11	
	Wi-Fi	Markov-based recurrent neural network	55%	70%	19	11	

4.2 Feature importance

The random forest provides us a chance to revisit the topic of feature engineering we discussed in the previous section. There are multiple ways to define feature importance, and no strict consensus has been reached so far. In this study, we leverage scikit-learn, the Python-based

⁹ The value on the reference [18], [35], [36] in this table was estimated from figures and is accordingly approximate numbers rather than accurate numbers

machine learning library [44], to calculate the feature importance. In scikit-learn, the feature importance is defined by the Mean Decrease Impurity (MDI) [45]. MDI is the weighted average of the total decrease in node impurity of each feature over all trees of that ensemble. If a feature is important, then the node impurity¹⁰ would be markedly reduced by passing the splits that include that feature.

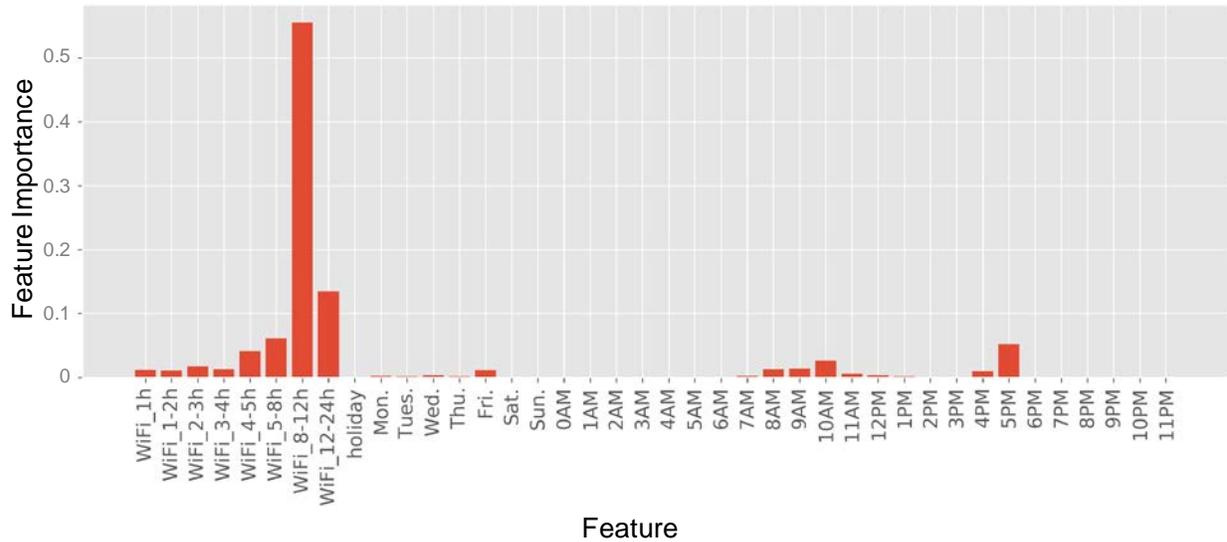


Figure 9: Feature importance for occupant counts estimation

As we expected, the number of long-term connected devices is a better feature for occupant count estimation than the number of short-term connected devices. Because Figure 9 illustrates that the number of devices connected to Wi-Fi for 8-12 hours per day is the single most important feature, with higher feature importance than other features. Those devices are highly likely to be personal computers that do not shut off during lunch break. It is a bit surprising that devices connected to Wi-Fi for more than 12 hours per day are also very important features to infer occupant counts. This might be because those devices are good indicators of working and non-working days. Devices connected for 5-8 hours per day and 4-5 hours per day are very likely to be office workers' cellphones, ranking 3rd and 5th in the feature importance lists. Those devices are not as important as we thought because cellphones might enter the idle mode from time to time and lose Wi-Fi connection even though the occupants keep staying indoors. Accordingly, this information is noisy in inferring occupant counts.

In addition to the number of connected devices, we use the features of the time in the random forest algorithm to capture the periodic behavior of occupant counts. Generally speaking, features of the time are not very important to infer occupant counts, because the information the time features could bring have already been reflected by the Wi-Fi connection counts, as Wi-Fi connection counts demonstrate a similar periodic variation. Features of time would be

¹⁰ The node impurity is a measure of the homogeneity of the labels at the node. In a regression problem, as in this case, the node impurity could be calculated as the variance of observations in that specific node

important only when they could capture some behaviors which have not been reflected by Wi-Fi connection counts. For instance, at 5 PM, the Wi-Fi connection counts are still pretty high while the people start to leave the office, due to some time lag effect. Because of this, 5 PM is the single most important feature compared with other hours of the day.

4.3 Limitations and future work

In this study, we rely on tracking the connection time to cluster each device into long-term or short-term connected devices. However, we realized that cellphone manufacturers are developing new privacy protection functions, such as automatic randomization of MAC address when the device is searching for a Wi-Fi network. The automatic address randomization technology would make the device tracking and clustering based on daily connection time more difficult, and has negative impacts on the inference accuracy. However, we believe this influence would be minimal. Because we found the counts of long-term connected devices (more than 8 hours per day) are more important features for occupant count inference. The number of connected cellphones is actually not a very important feature, as shown in Figure 9, because cellphones might enter the idle mode from time to time and lose Wi-Fi connection even though the occupants stay indoors. In this regard, the newly developed privacy protection functions might restrict the application of this occupant count inference approach, but in a limited way.

Another limitation of this study lies in whether the model we learned from one building could be applied to another. Transfer learning is associated with the question whether the knowledge learned from one task could be transferred to another. To be more specific, whether the mapping relation between the Wi-Fi connection counts and occupant counts we learned from one building could be applied to another building. This is critical since collecting the ground truth data – in this case, the real occupant counts – is expensive in the real world. An occupant count estimator would be valuable only if the trained estimator could be transferred to other buildings without retraining, as it is expensive and impractical to collect the ground truth data (occupant count) for every building. Actually, transferring and generalizing the knowledge learned from one building to another is a major constraint to be solved in occupant behavior studies as occupant behaviors in different buildings varied significantly [46].

Theoretically, whether an estimator could be transferred to other buildings or not depends on whether the mapping relation between the features (Wi-Fi connection counts, and time) and outputs (occupant counts) is stable and could be generalized to other buildings. The essence behind this mapping relation is the distribution of how many connected devices each person has, and whether this distribution would change from building to building. It is reasonable to argue that this distribution would change by different building types (e.g., office vs. retail), but stays stable and predictable in buildings with similar functions and occupants' Wi-Fi connection behaviors. For example, occupant Wi-Fi behaviors might be different in restaurants and offices, since occupants are more likely to have one device connected with Wi-Fi in restaurants (only their cellphones), but two in office buildings (cellphones and laptops). Therefore, clustering the buildings first and then develop estimators for each category of buildings with similar characteristics might be necessary to guarantee the scalability of the occupancy estimator. Das

et al. proposed two methods to cluster buildings given the ground truth data are unknown: by building functions, and by input data patterns [47]. However, more in-depth discussion is still in need.

As for the next step, we plan to collect data from different buildings to test if the estimator trained in one office building could be transferred and applied to another building without the ground truth data of occupant counts. It would also be helpful if researchers in this field could open-source their data and establish a shared database for the testing and comparison of new methods and algorithms.

5. Conclusions

Inferring occupant counts has wide applications in energy efficient building control. Though multiple methods have been proposed to estimate occupant counts, there is still a demand to detect occupant counts in an accurate and non-intrusive way using existing information infrastructure in buildings.

We employed the Random Forest method to infer occupant counts using the Wi-Fi connection counts data. The method was tested in a real office building and demonstrated better accuracy than the existing methods in the literature. In an office area with an average occupancy of 22-27 people and a peak occupancy of 48-74 people, the root square mean error is four people on the test set. For more than 70% of estimations, the errors are within two people counts, and for more than 90% of estimations, the errors are within six people counts.

The major contribution of this paper is proposing a novel and accurate approach to detect occupant counts in a non-intrusive way, utilizing the existing Wi-Fi infrastructure in buildings without requiring the installation of extra hardware or sensors. As an infrastructure deployed in almost every modern building, Wi-Fi data provide a unique opportunity to infer occupant counts with minimum additional cost. Our proposed method utilizes anonymized Wi-Fi data which can be adopted by other buildings to infer occupant counts for energy efficient building control. . Future research will explore transfer learning so the machine trained estimator of occupant counts can be applied to other buildings of similar types but without the ground truth data of occupant counts.

Acknowledgment

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 Wannu Zhang, as well as data collection and related support from Michael Smitasin, Baptiste Ravache, Bruce Nordman, Han Li, and Sang Hoon Lee.

Appendix

A. Pseudo code for the daily calibration of occupant counts data

Pseudocode

Set $occupant(t)$ = the number of occupants at time t , $occupantFlow(t)$ = the net number of occupants entering the space, T_start = the start time of detection, T_end = the end time of detection

For t in (T_start, T_end) :

 If $t == 3\text{ am}$:

$occupant(t) = 0$ # reset the occupant counts to 0 at the begin of day

 Else if $occupant < 0$:

$occupant(t) = 0$ # reset the occupant counts to 0 if occ falls below 0

 Else:

$occupant(t) = occupant(t-1) + occupantFlow(t)$

References

- [1] International Energy Agency, "IEA Statistics," 2016. [Online]. Available: <https://www.iea.org/>. [Accessed: 18-Dec-2018].
- [2] C. A. Webber, J. A. Roberson, M. C. McWhinney, R. E. Brown, M. J. Pinckard, and J. F. Busch, "After-hours power status of office equipment in the USA," *Energy*, vol. 31, no. 14, pp. 2823–2838, 2006.
- [3] O. T. Masoso and L. J. Grobler, "The dark side of occupants' behaviour on building energy use," *Energy Build.*, vol. 42, no. 2, pp. 173–177, Feb. 2010.
- [4] Y. Agarwal, B. Balaji, S. Dutta, R. K. Gupta, and T. Weng, "Duty-cycling buildings aggressively: The next frontier in HVAC control," in *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*, 2011, pp. 246–257.
- [5] V. L. Erickson, S. Achleitner, and A. E. Cerpa, "POEM: Power-efficient Occupancy-based Energy Management System," in *Proceedings of the 12th International Conference on Information Processing in Sensor Networks*, New York, NY, USA, 2013, pp. 203–216.
- [6] W. J. Fisk and A. T. de Almeida, "Sensor-based demand-controlled ventilation: a review," 1998.
- [7] A. Mirakhorli and B. Dong, "Occupancy behavior based model predictive control for building indoor climate—A critical review," *Energy Build.*, vol. 129, pp. 499–513, Oct. 2016.

- [8] S. C. Benghea, A. D. Kelman, F. Borrelli, R. Taylor, and S. Narayanan, "Implementation of model predictive control for an HVAC system in a mid-size commercial building," *HVACR Res.*, vol. 20, no. 1, pp. 121–135, Jan. 2014.
- [9] 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.
- [10] R. Melfi, B. Rosenblum, B. Nordman, and K. Christensen, "Measuring building occupancy using existing network infrastructure," in *2011 International Green Computing Conference and Workshops*, 2011, pp. 1–8.
- [11] P. Price *et al.*, "Automated Measurement and Verification and Innovative Occupancy Detection Technologies," LBNL-1007182, 2015.
- [12] Z. Wang *et al.*, "Individual difference in thermal comfort: A literature review," *Build. Environ.*, vol. 138, pp. 181–193, Jun. 2018.
- [13] J. Kim, S. Schiavon, and G. Brager, "Personal comfort models – A new paradigm in thermal comfort for occupant-centric environmental control," *Build. Environ.*, vol. 132, pp. 114–124, Mar. 2018.
- [14] P. X. Gao and S. Keshav, "Optimal Personal Comfort Management Using SPOT+," in *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, New York, NY, USA, 2013, pp. 22:1–22:8.
- [15] A. Rabbani and S. Keshav, "The Spot* System for Flexible Personal Heating and Cooling," in *Proceedings of the 2015 ACM Sixth International Conference on Future Energy Systems*, New York, NY, USA, 2015, pp. 209–210.
- [16] K. Shan, Y. Sun, S. Wang, and C. Yan, "Development and In-situ validation of a multi-zone demand-controlled ventilation strategy using a limited number of sensors," *Build. Environ.*, vol. 57, pp. 28–37, Nov. 2012.
- [17] S. Wang, J. Burnett, and H. Chong, "Experimental Validation of CO₂-Based Occupancy Detection for Demand-Controlled Ventilation:," *Indoor Built Environ.*, Jul. 2016.
- [18] 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.
- [19] W. Fisk, D. Faulkner, and D. Sullivan, "Accuracy of CO₂ Sensors in Commercial Buildings: A Pilot Study," 2006.
- [20] R. Tesoriero, R. Tebar, J. A. Gallud, M. D. Lozano, and V. M. R. Penichet, "Improving location awareness in indoor spaces using RFID technology," *Expert Syst. Appl.*, vol. 37, no. 1, pp. 894–898, Jan. 2010.
- [21] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The Active Badge Location System," *ACM Trans Inf Syst*, vol. 10, no. 1, pp. 91–102, Jan. 1992.
- [22] J. A. Davis and D. W. Nutter, "Occupancy diversity factors for common university building types," *Energy Build.*, vol. 42, no. 9, pp. 1543–1551, Sep. 2010.
- [23] M. S. Gul and S. Patidar, "Understanding the energy consumption and occupancy of a multi-purpose academic building," *Energy Build.*, vol. 87, pp. 155–165, Jan. 2015.
- [24] A. K. Mikkilineni, J. Dong, T. Kuruganti, and D. Fugate, "A novel occupancy detection solution using low-power IR-FPA based wireless occupancy sensor," *Energy Build.*, vol. 192, pp. 63–74, Jun. 2019.

- [25] E. Hailemariam, R. Goldstein, R. Attar, and A. Khan, "Real-time Occupancy Detection Using Decision Trees with Multiple Sensor Types," in *Proceedings of the 2011 Symposium on Simulation for Architecture and Urban Design*, San Diego, CA, USA, 2011, pp. 141–148.
- [26] J. A. Díaz and M. J. Jiménez, "Experimental assessment of room occupancy patterns in an office building. Comparison of different approaches based on CO₂ concentrations and computer power consumption," *Appl. Energy*, vol. 199, pp. 121–141, Aug. 2017.
- [27] R. Razavi, A. Gharipour, M. Fleury, and I. J. Akpan, "Occupancy detection of residential buildings using smart meter data: A large-scale study," *Energy Build.*, vol. 183, pp. 195–208, Jan. 2019.
- [28] 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)*, 2013, pp. 114–119.
- [29] S. Datta and S. Chatterjee, "An Efficient Indoor Occupancy Detection System Using Artificial Neural Network," in *Proceedings of International Ethical Hacking Conference 2018*, 2019, pp. 317–329.
- [30] W. Wang, T. Hong, N. Li, R. Q. Wang, and J. Chen, "Linking energy-cyber-physical systems with occupancy prediction and interpretation through WiFi probe-based ensemble classification," *Appl. Energy*, vol. 236, pp. 55–69, Feb. 2019.
- [31] 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.
- [32] M. Pritoni, M. Piette, and B. Nordman, "Accessing Wi-Fi Data for Occupancy Sensing," LBNL-2001053, 2017.
- [33] B. S. Çiftler, S. Dikmese, İ. Güvenç, K. Akkaya, and A. Kadri, "Occupancy Counting With Burst and Intermittent Signals in Smart Buildings," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 724–735, Apr. 2018.
- [34] Li Xuan, Liu Xuesong, and Qian Zhen, "Towards an Occupancy-Enhanced Building HVAC Control Strategy Using Wi-Fi Probe Request Information," *Comput. Civ. Eng.* 2017.
- [35] W. Wang, J. Chen, and X. Song, "Modeling and predicting occupancy profile in office space with a Wi-Fi probe-based Dynamic Markov Time-Window Inference approach," *Build. Environ.*, vol. 124, pp. 130–142, Nov. 2017.
- [36] 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.
- [37] 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.
- [38] I. Bisio, F. Lavagetto, M. Marchese, and A. Sciarrone, "Smart probabilistic fingerprinting for WiFi-based indoor positioning with mobile devices," *Pervasive Mob. Comput.*, vol. 31, pp. 107–123, Sep. 2016.
- [39] J. Wang, N. C. F. Tse, and J. Y. C. Chan, "Wi-Fi based occupancy detection in a complex indoor space under discontinuous wireless communication: A robust filtering based on event-triggered updating," *Build. Environ.*, vol. 151, pp. 228–239, Mar. 2019.

- [40] 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.
- [41] A. Liaw and M. Wiener, "Classification and regression by randomForest," *R News*, vol. 2(3), pp. 18–22, 2002.
- [42] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
- [43] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 2nd Edition. OTexts: Melbourne, Australia, 2018.
- [44] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, p. 2825–2830, Oct. 2011.
- [45] L. Breiman, *Classification and Regression Trees*. Routledge, 2017.
- [46] Z. Wang, T. Hong, and R. Jia, "Buildings.Occupants: a Modelica package for modelling occupant behaviour in buildings," *J. Build. Perform. Simul.*, vol. 0, no. 0, pp. 1–12, Nov. 2018.
- [47] A. K. Das, P. H. Pathak, J. Jee, C.-N. Chuah, and P. Mohapatra, "Non-Intrusive Multi-Modal Estimation of Building Occupancy," in *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*, New York, NY, USA, 2017, pp. 14:1–14:14.