Building and Environment 226 (2022) 109689

Contents lists available at ScienceDirect

Building and Environment

journal homepage: www.elsevier.com/locate/buildenv

Occupancy prediction using deep learning approaches across multiple space types: A minimum sensing strategy

Zeynep Duygu Tekler, Adrian Chong [*]

Department of the Built Environment, National University of Singapore, 4 Architecture Drive, 117566, Singapore

ARTICLE INFO

Keywords:
Occupancy prediction
Deep learning
Sensor fusion
Feature selection
Building science

ABSTRACT

The proliferation of sensing technologies has allowed the collection of occupancy-related data to support various building applications, including adaptive HVAC and lighting controls, maintenance operations, and space utilisation. However, past occupancy prediction studies often considered different combinations of sensor data and investigated a limited number of space types. This study performs occupancy prediction based on a minimum sensing strategy by using a comprehensive set of sensor data (i.e., indoor environmental and outdoor weather conditions, Wi-Fi connected devices, energy consumption data, HVAC operations, and time-related information) to identify the most crucial features through a proposed feature selection algorithm. Occupancy predictions were subsequently performed using different deep learning architectures, including Deep Neural Network (DNN), Long Short-Term Memory (LSTM), Bi-directional LSTM (Bi-LSTM), Gated Recurrent Unit (GRU), and Bi-directional GRU (Bi-GRU) in an office, library, and lecture room. Our findings highlighted that the proposed feature selection algorithm outperformed a popular feature selection algorithm to achieve a higher model performance with lower sensing requirements. Furthermore, empirical results showed that indoor CO_2 levels and Wi-Fi connected devices were crucial features for predicting occupancy across all space types. The best model performances were achieved using Bi-GRU for office, GRU for library, and Bi-GRU for lecture room.

1. Introduction

The adoption of various sensing technologies has enabled building managers and researchers to collect vast amounts of data on the building's operation and its occupants to facilitate effective building management while maintaining a comfortable indoor environment. The collection of occupancy and occupant-related information, in particular, has been useful in many building applications, including adaptive HVAC, lighting, and plug load controls [1–3], maintenance operations [4], space utilisation analysis [5], point-of-interest identification [6], and performative building design [7]. The occupancy information can be collected at different resolutions at the spatial scale (e.g., building, floor, zone, room), temporal scale (e.g., hours, minutes, seconds), and occupancy types (e.g., presence, count, identity, activity) [8] based on the sensing technology adopted. The complexity involved during the data collection process and the sensing requirements also increase proportionally with the resolution of the occupancy data obtained.

Occupancy sensing approaches can be categorised into two groups: terminal-based approaches and non-terminal-based approaches [9]. Terminal-based approaches use various active sensing technologies such as wearable sensors and the occupants' smartphone devices to collect high-resolution and accurate occupancy information within the building. The advancements in Internet-of-Things and communication technologies have also facilitated the adoption of different wireless technologies in occupancy sensing, such as Radio Frequency Identification (RFID) [10], Wi-Fi [11], and Bluetooth Low Energy (BLE) [12] to further improve the range, latency, accuracy, resolution, and energy performance of existing sensing approaches. However, despite their advantages, terminal-based approaches are limited due to the need to deploy dedicated sensors and perform third-party software installations (i.e., BLE beacons, Wi-Fi access points, mobile applications), which increases the implementation cost, intrudes upon the occupants' regular routines, and raises privacy concerns. Non-terminal-based approaches, on the other hand, relies on passive sensing technologies, such as CO_2 sensors [13], Passive Infrared (PIR) sensors [14], ultrasonic detection sensors [15], sound detection sensors [16], camera systems [17] and smart power meters [18], to indirectly collect the occupancy information for a particular zone within the building where the sensors are deployed. While these sensing technologies are arguably less intrusive than those used in terminal-based approaches, the resulting occupancy

[*] Corresponding author.
E-mail addresses: zdtekler@nus.edu.sg (Z.D. Tekler), adrian.chong@nus.edu.sg (A. Chong).

https://doi.org/10.1016/j.buildenv.2022.109689
Received 18 May 2022; Received in revised form 26 July 2022; Accepted 7 October 2022
Available online 19 October 2022

sensing systems are often limited in their detection accuracy and resolution.

Aside from the advancements in sensing technologies, the recent application of machine learning for occupancy sensing has resulted in notable improvements in detection accuracy over traditional approaches [19]. These improvements can be attributed to the machine learning algorithms' ability to learn from the vast amount of sensor data collected from the building and identify its correlation with the building's occupancy. Advancements in machine learning have also led to the development of the field of deep learning, where dense neural networks can be used to capture the hidden relationship between the building's sensor data and occupancy information by iteratively updating the networks' parameters based on the error gradient. Deep learning architectures are also able to provide more flexibility than traditional machine learning algorithms due to their ability to automatically extract high-level representations for model inference [20]. This is known as feature engineering and is a useful step previously performed by human domain expertise to produce handcrafted features for improving model performance [21]. Several specialised deep learning architectures (i.e., Long-Short Term Memory (LSTM) and Gated Recurrent Units (GRU)) have also been proposed and applied in fields dealing with time-series data (i.e., vehicle activity recognition, weather forecasting, financial market forecasting) due to the networks' designed ability to retain the sequential correlation between the data collected within the same period [22–24]. Given the time-series nature of the building's sensor data and the temporal correlation in its occupancy, the application and comparison of different deep learning architectures for occupancy prediction require a deeper investigation.

Currently, researchers and building managers face two main challenges. Firstly, with the plethora of different sensor data (e.g., indoor environmental data, Wi-Fi data, and energy consumption data) that has been used by past studies to perform occupancy prediction, it is challenging to perform a fair comparison between different studies and identify the most crucial sensors to deploy within the building when implementing a new occupancy prediction model. Past studies were also conducted in different space types and using different modelling approaches without clear benchmarks, which further reduces the generalisability of their results to other space types and model architectures not evaluated. Secondly, developing such data-driven approaches is a costly effort as it often requires setting up the necessary infrastructure and communication network to collect and store the vast amounts of sensor data collected from the building. Apart from the initial installation, there is also a need to regularly monitor and maintain the deployed sensors to ensure that the occupancy models remain operational. Therefore, a real-world occupancy prediction system implementation requires striking an optimal balance between high model performance and low sensing requirements.

1.1. *Objective and contributions*

The objective of this paper is to perform occupant count prediction using different deep learning architectures within multiple space types based on a minimum sensing strategy.

The contributions of this work are listed as follows:

- Proposed a novel feature selection algorithm and applied it to a comprehensive dataset containing a wide range of sensor data (i.e., indoor environmental and outdoor weather data, Wi-Fi connected devices, energy consumption data, HVAC operations data, and time-related information) to achieve a minimum sensing strategy.
- Identified the most crucial and optimal number of features for occupant count predictions in three different space types (i.e., office space, library, and lecture room).
- Evaluated the performance of several deep learning architectures (i.e., Deep Neural Networks, LSTM, Bidirectional LSTM (Bi-LSTM), GRU, and Bidirectional GRU (Bi-GRU)) and compared their performance in different space types.

2. Related work

This section provides a comprehensive review of past occupancy prediction studies using different sensor data and machine learning approaches to perform occupancy prediction in different space types.

Some of the most common sensor data used in occupancy prediction comprise of indoor environmental data, which encompasses a wide variety of information, including indoor CO_2 levels, indoor air temperature, pressure, relative humidity, illuminance, sound pressure level, and $PM_{2.5}$ levels. For instance, a study conducted by Lam et al. [25] predicted the number of occupants in a smart office testbed by using a Hidden Markov model (HMM) to analyse the indoor environmental data collected from the testbed. The indoor environmental data was captured using a system of sensor networks to obtain various parameters such as CO_2 levels, carbon monoxide levels, volatile organic compounds, $PM_{2.5}$, acoustics, illumination, motion, temperature, and humidity. Another study conducted by Vela et al. [26] attempted to perform occupancy prediction in a fitness gym and a residential living room based on the indoor environmental data (e.g., relative humidity, temperature, atmospheric pressure, altitude) collected from both spaces. The authors evaluated the performance of three machine learning algorithms, including support vector machine (SVM), k-nearest neighbour (KNN), and decision trees (DT). An IoT framework was also proposed by Hitimana et al. [27] to capture the real-time indoor environmental data in an office space for occupancy prediction. The proposed occupancy prediction model follows a LSTM neural network architecture and is compared against other machine learning algorithms, including SVM, Naive Bayes network, and multilayer perception feed-forward network. Chen et al. [28] proposed deep learning model based on a convolutional deep bi-LSTM architecture to automatically learn the local sequential features from the raw environmental sensor data collected from the study area and encode the temporal dependencies of these local features. The study was conducted in a university research lab where several indoor environmental sensor data (e.g., CO_2 levels, air temperature, air pressure, and humidity) were collected for building occupancy prediction.

Some studies have also attempted to combine the energy consumption data of various building systems (e.g., HVAC, lighting, and plug load) with indoor environmental data to enhance the performance of the occupancy prediction model. The energy data were either collected at the building level through smart meters or at the individual appliance level through the deployment of smart plugs [29]. An example is a study conducted by Razavi et al. [30], which analysed the electricity consumption data of more than 5000 residential homes and evaluated the performance of a wide array of machine learning models (e.g., SVM, KNN, Random Forest (RF), Gradient Boosting (GB), and neural networks) to predict the present and future occupancy status. Another study conducted by Park et al. [31] proposed an occupancy detection model based on an LSTM architecture by using the energy consumption data collected from smart plugs to identify occupant presence in residential homes. Ryu and Moon [32] also proposed a machine learning-based occupancy prediction model, which combines indoor environmental sensor data (e.g., air temperature, humidity, CO_2 levels, illuminance, and motion data) with lighting and appliance energy consumption data to predict the occupant count in a university testbed. The proposed model is developed using a two-step approach, which first detects the current occupancy level based on the present indoor environmental conditions and energy consumption data before predicting the future occupancy using an HMM.

Other less common sensor data used for occupancy prediction include pyroelectric infrared sensors, which have been used by Liu et al. [33] to infer the real-time occupancy presence information of a real-world office space. The study proposes a low-cost, battery-power, wireless occupancy detector with a pre-trained HMM capable of generating the occupancy status of an area of interest. Huckuk et al. [34] also attempted to use the thermostat data from 100 randomly selected residential homes to develop a machine learning model for predicting future occupancy state. Different models were evaluated during the study, including simple baseline models, classification models including Logistic Regression (LR) and RF, and sequential models such as HMM and Recurrent Neural Networks (RNN). The study concluded that the RF algorithm could outperform all other candidate models, which raises several questions regarding the generalisability of the study's findings to other space types and sensor data. Lastly, Yuan et al. used the location-based services supported by the users' mobile devices to obtain their presence information within a building. Based on the occupancy data obtained from 16 different buildings, the authors developed an integrated approach using temporal-sequential analysis and an ANN model to perform occupancy forecasting [35].

Apart from the studies reviewed above, some have also attempted to include an additional feature selection step during the model development stage to identify the most crucial features for occupancy prediction. For instance, Zimmermann et al. [36] proposed a correlation-based feature selection algorithm to identify useful subsets of environmental features for occupancy prediction before evaluating the resulting model performance based on different machine learning algorithms (e.g., Naive Bayes, DT, LR, KNN, and RF). Another study conducted by Chen et al. proposed a data fusion framework that consists of an extreme learning machine-based (ELM) wrapper method to identify the most important subset of environmental sensors for occupancy prediction before evaluating the model performance against different machine learning models. The different machine learning models evaluated include ELM, SVM, Deep Neural Network (DNN), KNN, linear discrimination analysis (LDA), and DT. Wang et al. [37] also proposed an adaptive lasso approach that evaluates the correlation between different environmental and Wi-Fi-based features (e.g., temperature, relative humidity, CO_2 level, media access control (MAC) address, and received signal strength indicator (RSSI) value of Wi-Fi-enabled devices) to identify the most critical features for occupancy detection. A DNN model was also developed to predict the number of occupants and occupancy level in a graduate student office based on the important features identified by the proposed adaptive lasso method to result in a computationally efficient model. Lastly, a study conducted by Masood et al. [38] introduced two feature selection methods to identify the most crucial indoor environmental features for occupancy prediction. The first method is a wrapper-based method known as WRANK-ELM, which selects from an ordered list of features using an ELM classifier, while the second approach uses a filter-wrapper hybrid method (i.e., RIG-ELM) that uses the relative information gain criterion to rank each feature before applying an ELM to perform an incremental search.

However, despite the amount of past research conducted on this topic, many studies have only considered utilising a limited number of features for occupancy prediction, with other building-related data such as HVAC operations data, outdoor weather data, and Wi-Fi related data being underutilised. Furthermore, past studies that attempted to identify the most crucial features for occupancy prediction are often lacking as the feature selection analysis is often conducted on a small pool of sensor data collected from a specific space type. These factors reduce the generalisability of the studies' findings, especially when dealing with buildings with different space types and containing other sensor data that was not considered in the original study. Therefore, this study addresses these research gaps by applying a feature selection algorithm to a comprehensive set of building sensor data to identify the most crucial features for occupancy prediction and strike a good balance between model performance and minimum sensing requirements. To further enhance the generalisability of our findings, an identical analysis was also conducted over three different space types (i.e., office space, library, and lecture room) while evaluating five different deep learning architectures. Table 1 provides a summary of all of the studies reviewed in this section and the scope of this study by highlighting the studies' occupancy resolution, sensor data used, machine learning models evaluated, and space types investigated.

3. Methodology

This section provides an overview of the methodology proposed to perform occupant count prediction based on a minimum sensing strategy. We begin by collecting a comprehensive set of sensor data from three different space types in our study building and their corresponding occupant count information to serve as ground truth. Following this, several data processing steps are performed to address any erroneous data and extract the time-related features to improve the model's predictive performance. The updated list of features is subsequently passed into a 2-step feature selection algorithm to identify the most crucial features for occupancy prediction and evaluated using five different deep learning architectures along with various analysis. The detailed description of each step is elaborated in the following subsections and depicted in Fig. 1.

3.1. Space type and sensor data description

The dataset [40] used in this study was collected from the School of Design and Environment 4 (SDE4) building at the National University of Singapore. SDE4 is a six-story net-zero energy building in Singapore with a total gross floor area of around 8588 square meters. Three different space types were analysed in this study, including an office space for researchers, a library space accessible to all students, and a lecture room, located on different levels of the study building. A detailed description of each space type, such as the floor level, floor area, floor-to-ceiling height, room volume, seating capacity, maximum occupancy density, and HVAC system deployed, has been provided in Table 2. It should also be highlighted that the office space and library are conditioned by a common air handling unit (AHU) serving multiple rooms in the building, while the lecture room is conditioned by fan coil units (FCU).

By developing a data fusion pipeline to extract the sensor data collected from each space type, we obtained a total of 123 days of data collected during the weekdays at a sampling resolution of 5-minute intervals. The data categories included in this dataset consist of indoor and outdoor environmental conditions, number of Wi-Fi connected devices at each access point, energy consumption data of different end uses (i.e., HVAC, ceiling fan, plug loads, and lighting), and HVAC operations collected through various sensors deployed at the room and building level. Furthermore, the dataset is supplemented with the ground truth occupant presence and count information for each space type by manually reviewing the surveillance camera footage deployed within and at the entrances of each space type. Each data entry is also appended with the timestamp information, indicating the date and time when each entry is captured. The detailed information of each sensor data, including its respective unit, data category, and availability in each space type, is provided in Table 3.

3.2. Data processing

Based on the raw sensor data obtained through the data fusion pipeline, multiple data processing steps were performed before the data is passed into the proposed feature selection algorithm.

The first step involves removing missing or erroneous data as a result of sensor failure or sensor fault. Missing sensor data occurs when the sensor fails to collect any readings or the readings were lost in transmission during a particular time interval, while erroneous data occurs when the sensor successfully records a reading but it was not successfully stored in the database due to hardware glitches. The erroneous sensor data is removed from the dataset by replacing it with a missing value before addressing it with the other missing data in the dataset using a data imputation algorithm known as MissForest [41]. The random forest-based imputation algorithm begins by imputing the candidate column with the least number of missing values and replacing the missing values in the remaining columns with their

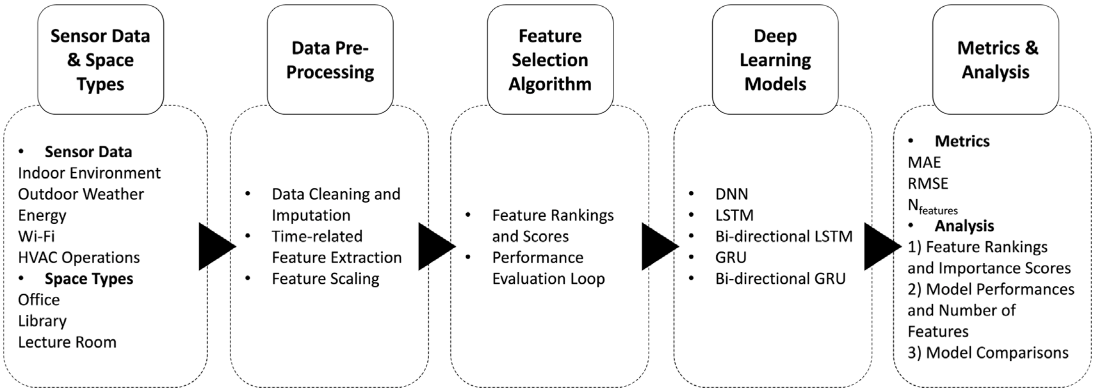

Fig. 1. Overview of the steps taken to perform occupant count prediction and to achieve a minimum sensing strategy.

Table 1
Reviewed studies based occupancy resolution, sensor data, machine learning models and space types.

Study	Occupancy resolution	Sensor data	Models	Space types
[33]	Presence	Motion	HMM	Office
[36]	Presence, Count	CO_2, VOC, air temperature, relative humidity	NB, DT, LR, kNN, RF	Residential (Student Apartment)
[34]	Presence	Thermostat	LR, RF, HMM, RNN	Residential
[39]	Presence	CO_2, pressure, air temperature, relative humidity	ELM, SVM, ANN, kNN, LDA, CART	Office
[30]	Presence	Electricity consumption	RF, SVM, kNN, ANN, GB	Residential
[31]	Presence	Plug load energy consumption	LSTM	Office
[28]	Count(low, medium, high)	CO_2, air temperature, pressure, relative humidity	Convolutional Deep Bi-directional Long Short-Term Memory (CDBLSTM)	Office (Graduate Student)
[32]	Count	CO_2, air temperature, relative humidity, illuminance, motion, energy consumption of lighting, plug loads, HVAC	DT, HMM	Office (Test-bed)
[11]	Count	CO_2, air temperature, relative humidity, MAC address	kNN, ANN, SVM	Office (Graduate Student)
[27]	Count	CO_2, air temperature, relative humidity, illuminance, motion	SVM, NB, MLP, LSTM	Office
[37]	Count(low, medium, high)	CO_2, air temperature, relative humidity, MAC address	ANN, Feature Selection Algorithm (Adaptive lasso filtering)	Office (Graduate Student)
[26]	Count(low, medium, high)	Relative humidity, air temperature, pressure, altitude	kNN, SVM, DT	Fitness Gym and Residential (Living Room)
[25]	Count	CO_2, VOC, outside temperature, dew point, $PM_{2.5}$, lighting, indoor temperature, indoor relative humidity, motion, indoor acoustics	HMM	Office
[38]	Count	CO2, relative humidity, air temperature, pressure	ELM, Feature Selection Algorithm (Wrapper-based ranking using ELM and Filter-wrapper hybrid algorithm based on RIG)	Office
[35]	Count	Smartphone devices	Temporal-sequential analysis with ANN	Railway Station, Airport, Commercial, Hospital
Tekler & Chong	Count	All sensor data in Table 3	DNN, LSTM, Bi-LSTM, GRU, Bi-GRU, Proposed Feature Selection Algorithm	Office, Library, Lecture Room

Table 2
Detailed description of each space type measured directly from the study building.

Space type	Level	Floor area [m^2]	Floor-to-ceiling height [m]	Volume [m^3]	Seating capacity [person]	Max. occupancy density [m^2/person]	HVAC type
Office	3	141.9	4.1	581.7	25	5.6	AHU
Library	2	182.8	7.5	1363.3	36	5	AHU
Lecture Room	4	118.6	4.1	486.2	40	3	FCU

Table 3
Detailed description of each sensor data used in this study, including its respective unit, data category, and availability in each space type.

Data category	Sensor data	Data unit	Avail. in office	Avail. in library	Avail. in lecture room
Indoor Environmental Quality	VOC	ppb	X	X	X
	Sound pressure level	dB(A)	X	X	X
	Relative humidity	%RH	X	X	X
	Air temperature	°C	X	X	X
	Illuminance	lux	X	X	X
	$PM_{2.5}$	$\mu g/m^3$	X	X	X
	Indoor CO_2	ppm	X	X	X
Wi-Fi	Wi-Fi connected devices	Number	X	X	X
Energy Consumption	Ceiling fan energy	kWh	X	X	X
	Lighting energy	kWh	X	X	X
	Plug load energy	kWh	X	X	X
	Chilled water energy	kWh	X	X	X
	AHU/FCU fan energy	kWh	X	X	X
HVAC Operations	Supply air flow	CMH	X	X	–
	Damper position	%	X	X	–
	Temperature setpoint	°C	X	X	X
	Cooling coil valve position	°C.	X	X	–
	Cooling coil valve command	°C	X	X	–
	AHU/FCU fan speed	Hz	X	X	X
	Offcoil air temperature	°C	X	X	–
	Offcoil temperature setpoint	°C	X	X	–
	Supply air humidity	%RH	X	X	–
	Pressure across filter	Pa	X	X	–
	Supply air static pressure	Pa	X	X	X
	Supply air temperature	°C	X	X	X
Outdoor Weather	Baromatic pressure	hPa	X	X	X
	Dry bulb temperature	°C	X	X	X
	Global solar radiation	W/m^2	X	X	X
	Wind direction	°(Degree)	X	X	X
	Wind speed	m/s	X	X	X
	Outdoor CO_2	ppm	X	X	X
	Rainfall	mm	X	X	X
	Relative humidity	%RH	X	X	X
Time-related	Hour (12AM-12PM)	Hour	X	X	X
Occupancy	Occupant Count	Number	X	X	X

respective means. Following this, a RF model is trained by setting the candidate column as the output variable and the remaining columns as the model's input for those rows that do not contain missing values in the candidate column. After the model has been trained, it is used to impute the missing values in the candidate column before moving on to the next candidate column with the second smallest number of missing values. This process is repeated for each column over multiple iterations until there is minimal difference between the imputed dataset in the previous round and the newly imputed dataset.

After addressing the missing and erroneous data, the second data processing step involves extracting the time-related features from the dataset's timestamp to provide the model with more temporal information about the predicted occupancy level in each space type. This feature is particularly useful for space types whose occupancy levels are highly correlated with specific periods of the day. Some examples include the operating hours of the library space, specific time slots where lectures are regularly scheduled, and the working hours of the office workers. The time-related features are obtained by first extracting the hour information from each data point's timestamp before performing one-hot-encoding to obtain a one-hot vector of dimension 24 (i.e., one for each hour in the day).

The last data processing step involves feature scaling, where each numerical feature is transformed to follow a standard normal distribution with a mean of 0 and a standard deviation of 1. This step is performed to standardise each feature's magnitude and reduce the convergence time during the training process. An inverted transformation step is also performed during the model prediction phase to revert the model's output to its original scale before calculating its error scores.

3.3. Feature selection algorithm

Feature selection refers to a process of reducing the number of input features used within a predictive model by evaluating and identifying the most important features that significantly contribute to the model's performance. This step is crucial to achieving a minimum sensing strategy as the resulting occupancy prediction model is more robust to issues caused by sensor failure and lowers the overall cost related to deployment and maintenance. The reduction in the number of model parameters, in general, also helps to simplify the model's complexity, reduce the likelihood of overfitting, and decrease training time.

Based on the features obtained from each space type's sensor data and the time-related features extracted from the timestamp information, the next step in our proposed methodology involves the application of a 2-step feature selection algorithm for occupant count prediction, as depicted in Fig. 2.

3.3.1. Step 1: Feature rankings and scores

In the first step of the proposed algorithm, an extreme gradient boosting-based recursive feature elimination (XGB-RFECV) approach was used to generate the feature rankings for each feature in the dataset. At the same time, an extreme gradient boosting (XGB) model is used to generate the feature importance scores for the same feature set. The XGB model was chosen as the base model in this case due to its superior performance and efficient learning algorithm, which is crucial given the recursive nature of our proposed algorithm.

Recursive Feature Elimination (RFE) is a topdown elimination method that begins with the complete set of features and recursively eliminates the least relevant features that do not significantly contribute to the model's predictive performance. The relevancy of each feature is determined by fitting a machine learning algorithm (i.e., a XGB model) that ranks the features based on their importance scores. The least important features are discarded during each iteration, and the remaining features are refitted to the model to generate a new set of importance scores. The top-ranked features will be removed in the last few iterations, while those removed in the initial iterations are assigned a lower rank. RFE with cross validation (RFECV) extends upon RFE by including an additional cross-validation step to increase the approach's robustness. This is achieved by splitting the dataset into k folds and applying the RFE algorithm on each fold to generate each feature's ranking. The rankings from each fold are subsequently combined via averaging to obtain the final feature rankings [42].

Given that the feature rank assigned to each feature takes on an integer value with the possibility of multiple features being assigned the same rank, the features are once again fitted to an XGB model to generate their feature importance scores. The feature importance scores for decision tree-based models are calculated by weighting the decrease in impurity achieved at each attribute split point with the number of samples affected by the split. In the case of ensemble-based models, such as XGB models, the feature importance score is calculated by averaging the importance scores of each decision tree used within the ensemble [43]. This step helps identify the importance order between features within the same rank determined by RFECV.

Once the feature rankings and feature importance scores are calculated, the entire feature list is sorted first based on their feature rankings, followed by their feature importance scores. The sorted feature set is subsequently stored before moving on to the next step of the proposed feature selection algorithm.

3.3.2. Step 2: Performance evaluation loop

The second step involves the performance evaluation loop, where the top N features are selected from the sorted feature list from step 1 to train and evaluate the occupancy prediction model using a 5-fold rolling-basis cross-validation approach. The evaluation loop adopts a bottom-up approach by starting with the most important feature (i.e., N=1) and storing the resulting model's predictive performance before re-evaluating the model's performance with the top 2 most important features (N=2). By comparing the predictive performance of both models, the loop's exit condition is achieved when we observe a drop in the model performance after the inclusion of the Nth feature. Otherwise, N is incremented by 1, and the evaluation loop continues with the next iteration by fitting a new model based on the updated feature list.

A detailed flowchart of the proposed 2-step feature selection algorithm is provided in Fig. 2, together with its corresponding pseudocode (refer to Algorithm 1).

3.4. Model development

This section provides an overview of the five deep learning architectures investigated in this study to predict occupant count in multiple space types. The five deep learning architectures include DNN and sequential-based models such as LSTM, Bi-LSTM, GRU, and Bi-GRU, due to their ability to retain the temporal information in time-series data. The detailed description of each model architecture is described in the following subsections.

3.4.1. Deep Neural Network (DNN)

The DNN architecture consists of a layered arrangement of neurons, where each neuron passes a signal to the other neurons in the adjacent layer based on the received input from the previous layers. Depending on the model's complexity, each layer can consist of one or more neurons, where each neuron will apply an activation function on an incoming signal before passing it on to the neurons in the subsequent layers. By applying a weight to the connection between two neurons, the weight's magnitude can help to transform the input signal into a desired final output. During the training phase of the DNN, the weight parameters between two neurons are iteratively updated via a process known as backpropagation to train the model to generate the desired output. Furthermore, by increasing the number of hidden layers in the neural network and training the model on a sufficiently large amount of training data, DNNs can learn complex non-linear patterns to improve their prediction performance.

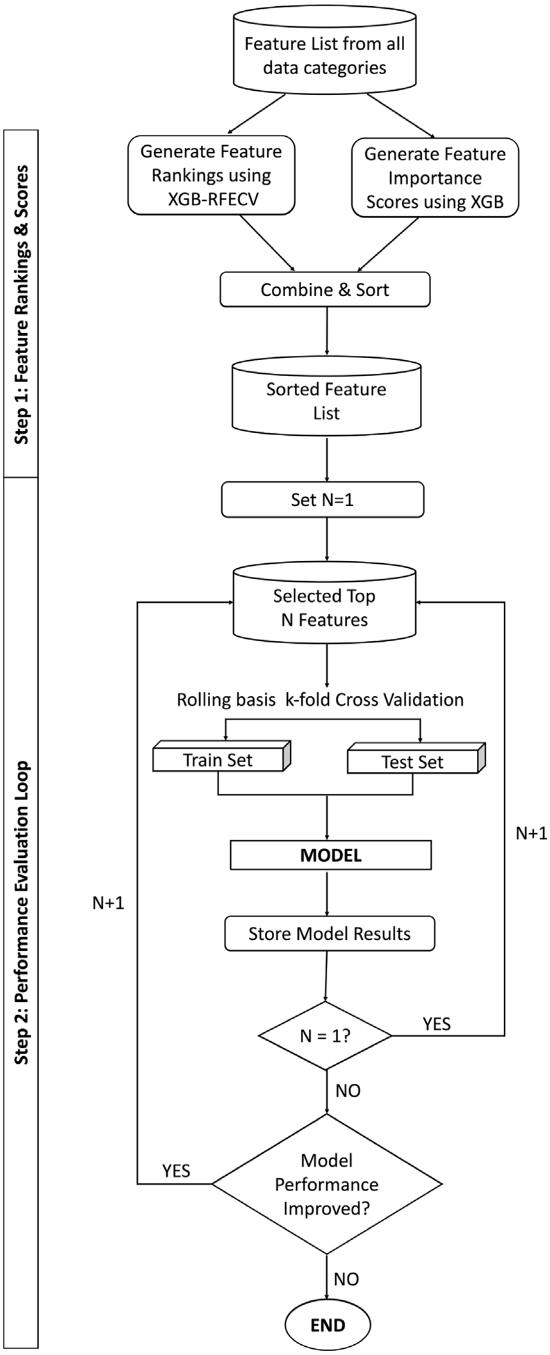
Step 1: Feature Rankings & Scores
Step 2: Performance Evaluation Loop
Feature List from all data categories
Generate Feature Rankings using XGB-RFECV
Generate Feature Importance Scores using XGB
Combine & Sort
Sorted Feature List
Set N=1
Selected Top N Features
Rolling basis k-fold Cross Validation
Train Set
Test Set
MODEL
Store Model Results
N = 1?
YES
NO
N+1
N+1
Model Performance Improved?
YES
NO
END

Fig. 2. Detailed flowchart of the proposed 2-step feature selection algorithm.

3.4.2. Long Short Term Memory (LSTM Neural Network)

LSTM is a type of sequential-based model that can be seen as an improvement over the DNN architecture due to its ability to retain historical information to inform its predictions [44]. As a result of this feature, this model architecture has been adopted in many applications

```
Algorithm 1: Proposed Feature Selection Algorithm
Result: Select top n features for occupant count prediction
$X$ represents the complete set of features evaluated using the
  proposed feature selection algorithm.
$y$ represents the predicted occupant count (i.e., ground truth).
Step 1: Feature Ranking and Scores
$feature\_rankings \leftarrow XGB - RFECV(X, y)$
$feature\_importance \leftarrow XGB.feature\_importance(X, y)$
sort $X$ in descending order based on $feature\_rankings$ followed by
  $feature\_importance$
Step 2: Performance Evaluation Loop
$n = 1$
while $n \leq \|X\|$ do
    $X_{selected} = X[: n]$;
    for $i \leftarrow 0$ to $k$ do
        $model \leftarrow train(X_{selected,train}, y_{train})$ on rolling basis
        $result \leftarrow evaluate(model.predict(X_{selected,test}), y_{test})$
    end
    $performance_{current} \leftarrow mean(result)$ if
     $performance_{prev} > performance_{current}$ then
        $X_{final} = X[: n-1]$;
        $exit$;
    else
        $n++$;
        $performance_{prev} \leftarrow performance_{current}$;
    end
end
```

involving sequential data, including image classification [45], stop activity recognition [22], and speech recognition [46]. The ability to retain past information is achieved through the LSTM unit, which consists of a cell state and three interacting gate layers (i.e., input gate, forget gate, and output) that regulate the flow of information into and out of the cell. Through these interacting layers, the LSTM architecture is said to resolve the exploding and vanishing gradient problems faced historically by the recurrent neural network architecture, allowing it to solve complex time series problems.

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

The Bi-LSTM is an extension of the regular LSTM architecture by combining two independent LSTMs. The first LSTM is provided information about the sequence in a forward fashion, while the second LSTM is provided information about the sequence in a backward fashion. This architecture allows information about the sequence's past and future states to be accounted for simultaneously when generating the model's output [47].

3.4.4. Gated Recurrent Unit (GRU)

The GRU architecture can be viewed as a relatively newer architecture based on recurrent neural networks but has a more straightforward implementation than LSTMs. Instead of having a cell state and three gates for regulating information flow, GRUs only consists of an update gate and a reset gate to regulate information flow. Due to a fewer number of tensor operations conducted, the GRU architecture requires a shorter training time and can produce a comparable performance to LSTMs [48].

3.4.5. Bidirectional Gated Recurrent Unit (Bi-GRU)

Lastly, the Bi-GRU architecture is an extension of the standard GRU architecture by putting two independent GRUs together. Each GRU is provided with the forward and backward information about the sequence for every time step, respectively, thereby allowing the model to account for the sequence's past and future states simultaneously.

4. Results and discussions

4.1. Model implementation

The occupant count prediction models are implemented using the Python language and the Tensorflow Keras API library to evaluate the overall model performance and ensure computational efficiency.

The models developed in this study were also trained and evaluated on the dataset based on a train-test ratio of 80%-20%. Furthermore, in order to retain the time-series nature of the dataset and to avoid information leakage that arises due to random sampling, the models were specifically trained on data that was initially collected (i.e., the first 80%) before they were evaluated on data that was collected towards the end of the data collection period (i.e., the final 20%). Both the training and test datasets are transformed by grouping the sequential data points based on a moving window of size 5 as the model's input. At the same time, the ground truth is set as the occupant count in the next time step (i.e., prediction horizon of 5 min). In this case, we have determined that a window size of 5 is appropriate as a shorter window size will provide limited information about the historical occupancy trends for forecasting, while a larger window size can reduce the model's predictive performance due to the introduction of excessive noise. Similarly, the prediction horizon is also set as 5 min as it is suitable for applications related to real-time predictive control, with a longer prediction window resulting in larger prediction errors.

All models are implemented using the same set of hyperparameters without hyperparameter tuning to allow a fair comparison between different deep learning architectures. These hyperparameters include:

- 3 hidden layers with 32 neurons/cells each
- ReLU activation function for neurons in the hidden layer
- An output layer with 1 neuron
- Adam optimiser
- Batch size of 32

4.2. Model evaluation metrics

This study uses two evaluation metrics to assess the models' predictive performance due to their frequent application in past occupancy prediction studies: Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

MAE is a measure of the absolute difference between the predicted occupant count O_p and the observed number of occupants O_{ob} in the selected space type, as defined in Eq. (1).

$$MAE(O_p) = \frac{1}{N}\sum_{i=1}^{N}|O_{obi} - O_{pi}| \tag{1}$$

where N is the size of the dataset

RMSE is another frequently used metric that measures an average of the squared differences between the predicted occupant count and the observed number of occupants in the selected space type, as represented in Eq. (2). This metric, in particular, exerts a more significant penalty on larger errors made by the occupancy prediction model.

$$RMSE(O_p) = \sqrt{\sum_{n=1}^{N}(O_{ob} - O_p)^2/N} \tag{2}$$

4.3. Feature rankings and importance scores in multiple space types

The results of the feature selection rankings and feature importance scores generated according to the first step of the proposed feature selection algorithm for occupant count prediction. Fig. 3 depicts the feature rankings and importance scores for the top 15 features for each space type (i.e., office, library, and lecture room) arranged in descending order. The feature ranking for each feature is labelled on the right-hand side of each bar, and the bar length corresponds

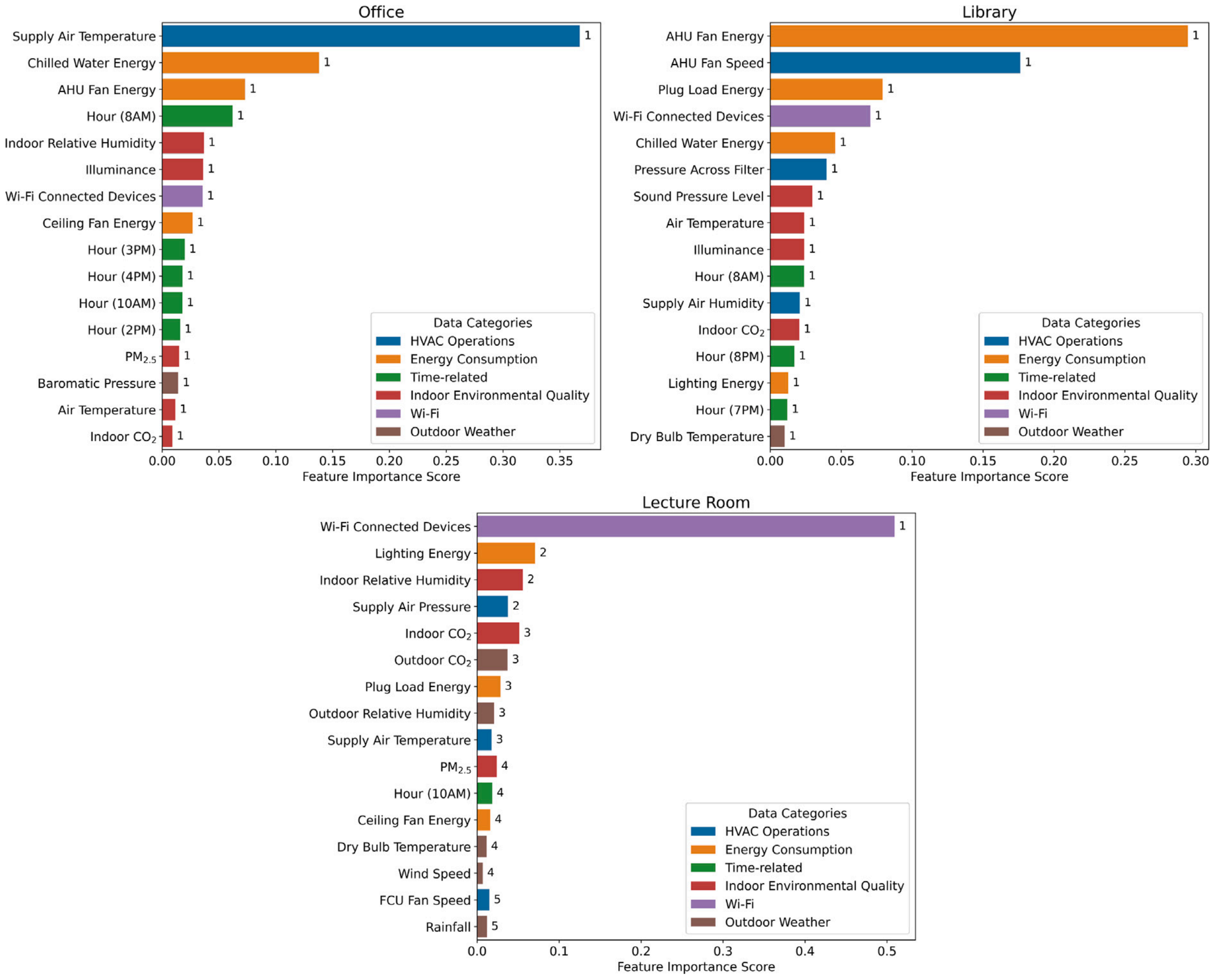

Fig. 3. Top 15 features of each space type (i.e., office, library, and lecture room) sorted in descending order according to their feature rankings and feature importance scores.

to the feature's importance score. On top of that, each feature is also colour-coded based on its respective data category (i.e., HVAC Operations, Energy Consumption, Time-related, Indoor Environmental Quality, Wi-Fi, and Outdoor Weather).

By analysing the results from Fig. 3, it can be observed that the most crucial features for occupancy prediction differ significantly between different space types.

For the office space, while all top 15 features were assigned a feature rank of 1, the supply air temperature was observed to be the most important feature overall for occupancy prediction, with an importance score of 0.35 out of 1. This feature is also observed to be the only important feature among the other HVAC operation-related features considered in the dataset. Apart from the top feature, the other important features for performing occupancy prediction in the office space include the energy consumption contributed by the AHU fan unit, chilled water energy, and ceiling fan. All these features are related to different HVAC systems deployed within the office, further highlighting the value of using HVAC-related features for occupant count prediction within office spaces. Apart from HVAC-related features, other features such as the indoor environmental conditions of the room (i.e., indoor relative humidity, illuminance, air temperature, indoor CO_2, and $PM_{2.5}$), the number of Wi-Fi connected devices, and certain hours of the day (i.e., 8 AM, 10 AM, 2 PM, 3 PM, and 4 PM) are also found to be useful for occupant count prediction.

For the library space, the findings appear to be similar to those highlighted in the office space, where HVAC-related features such as AHU fan energy, AHU fan speed, and chilled water energy are among the top 5 features for occupant count prediction. Apart from these features, other HVAC operations-related features such as fan speed, pressure across filter, supply air humidity, and the number of Wi-Fi connected devices are also found to be useful features for occupancy prediction. One of the interesting findings from Fig. 3 is the importance of plug load energy in predicting occupant count, as it is common for the students to charge their laptops or smartphone devices while they are studying in the library. On top of that, given that the library's operating hours, it is also not surprising to see time-related features being deemed important for occupant count prediction as they indicate a future increase or decrease in the library space's occupant count.

Lastly, for the lecture room, it can be observed from Fig. 3 that the number of Wi-Fi connected devices is found to be the most useful feature for predicting occupant count. This feature is also the only feature with a feature rank of 1 and is assigned a high feature importance score of 0.5 out of 1. Given that the lecture rooms are used to conduct regular lectures during the day and serve as a self-study space for students in the evening, the number of occupants in the lecture room can be accurately predicted based solely on the number of Wi-Fi devices connected to the room's access points. Furthermore, the usefulness of this feature is particularly evident compared to other space types such as the office or library space as the lecture room often does not come equipped with many Wi-Fi devices except for the electronic devices carried by the occupants (e.g., smartphone devices and laptops).

Apart from the differences among the different space types when identifying the most useful features for occupant count prediction, there are also several features found to be useful among all three space types. These features include the number of Wi-Fi devices in the room and the indoor CO_2 levels. When comparing different space type

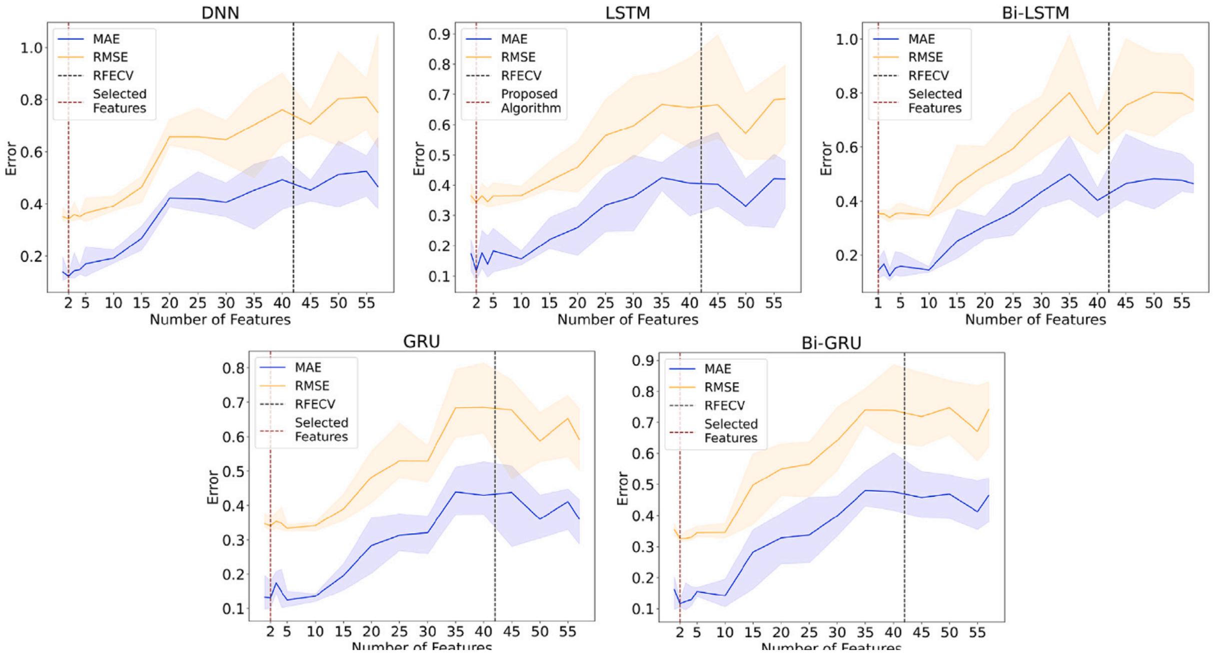

Fig. 4. Model performance for different deep learning architectures (i.e., DNN, LSTM, Bi-LSTM, GRU, Bi-GRU) when performing occupant count prediction in the office using different number of features.

pairs, we observed that chilled water energy, AHU fan energy, number of Wi-Fi connected devices, illuminance, indoor CO_2, air temperature level, and time-related features are important features for both office and library spaces. The number of Wi-Fi connected devices, indoor CO_2, lighting energy consumption, plug load energy consumption, fan speed, and dry bulb temperature are also important features for both library and lecture room. Finally, when comparing the office space and lecture room, the common features include the number of Wi-Fi connected devices, indoor CO_2, ceiling fan energy, indoor relative humidity, $PM_{2.5}$, supply air temperature, and time-related features.

4.4. Optimal number of features based on the proposed feature selection algorithm and RFECV

This section provides an in-depth comparison of the model performance of different occupancy prediction models developed using five different deep learning architectures in three different space types. To facilitate a comparison between the proposed feature selection algorithm and the RFECV algorithm, each model's predictive performance is evaluated using different number of features and sorted based on their importance score, before plotting out their results for each space type in Figs. 4–6. The prediction performance for each model is represented as a band to indicate the maximum, minimum, and average errors obtained through the 5-fold cross-validation step performed in the Performance Evaluation Loop in Section 3.3.

It can be observed from Fig. 4 that all five model architectures showed a similar pattern, where the model's prediction errors generally follow an upward trend when more features are included for occupancy prediction in the office space. Furthermore, when comparing the proposed feature selection algorithm with the RFECV algorithm, the proposed feature selection algorithm consistently outperformed the latter, for all model architectures, by requiring significantly fewer features while achieving a lower MAE and RMSE score. More specifically, the proposed feature selection algorithm determined that the optimal number of features for the DNN, LSTM, GRU, and Bi-GRU models to be set at 2 (i.e., supply air temperature and chilled water energy as listed in Fig. 3), while the Bi-LSTM model only required the use of supply air temperature for predicting occupant count. This result contrasts with the RFECV algorithm, which selected the top 42 features for occupant count prediction, which resulted in a higher MAE and RMSE score and a larger sensing requirement compared to our proposed algorithm.

A similar pattern can be observed in Fig. 5 where the proposed feature selection algorithm consistently outperformed the RFECV algorithm in the library space by selecting a significantly fewer number of features while still achieving lower MAE and RMSE scores. More specifically, the proposed feature selection algorithm evaluated that the DNN model required the use of the top 3 features for performing occupant count prediction (i.e., AHU fan energy, AHU fan speed, and plug load energy), while the LSTM, Bi-LSTM, GRU, and Bi-GRU models only required the use of the top feature (i.e., AHU fan energy) for occupant count prediction. This result differs from the RFECV algorithm's conclusion, which selected up to 22 features for occupant count prediction. Unfortunately, this resulted in higher MAE and RMSE error scores, as well as a more significant sensing requirement overall.

Lastly, based on Fig. 6, it can be observed that the RFECV algorithm and the proposed feature selection algorithm had both recommended a similar number of features for occupant count prediction in the lecture room. More specifically, the RFECV algorithm recommended the use of a single feature for occupancy prediction (i.e., number of Wi-Fi connected devices), while the proposed feature selection algorithm had proposed the use of the top two features (i.e., number of Wi-Fi connected devices and lighting energy) for the DNN, LSTM, Bi-LSTM, and Bi-GRU models. It can also be observed from Fig. 6 that with the inclusion of the second feature (i.e., lighting energy), all four models were able to further reduce their prediction errors (i.e., MAE and RMSE), outperforming the model generated by the RFECV algorithm.

4.5. Model comparisons between different deep learning models and ground truth

This section highlights the best deep learning models for each space type by comparing (1) the models' predictive performance based on their MAE and RMSE scores and (2) the number of optimal features selected based on the proposed feature selection algorithm. A minimum sensing strategy can be achieved by developing an ideal model with high predictive performance while having low sensing requirements.

By plotting the predictive performance of each model obtained as a result of the proposed feature selection algorithm and comparing

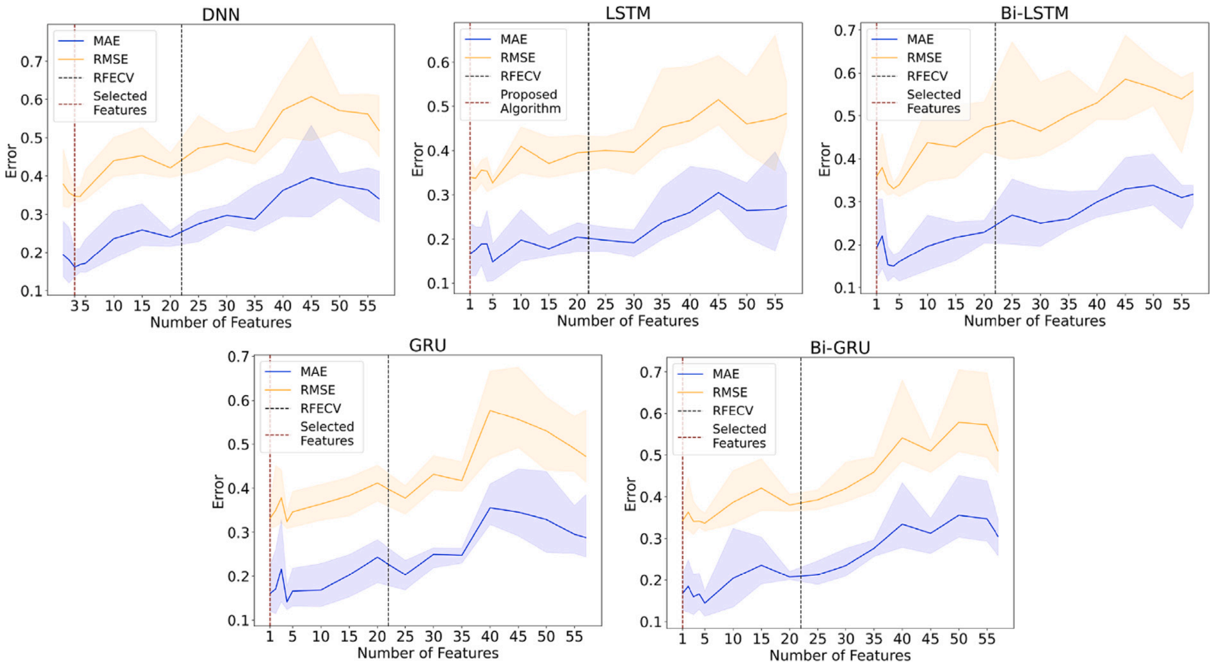

Fig. 5. Model performance for different deep learning architectures (i.e., DNN, LSTM, Bi-LSTM, GRU, Bi-GRU) when performing occupant count prediction in the library using different number of features.

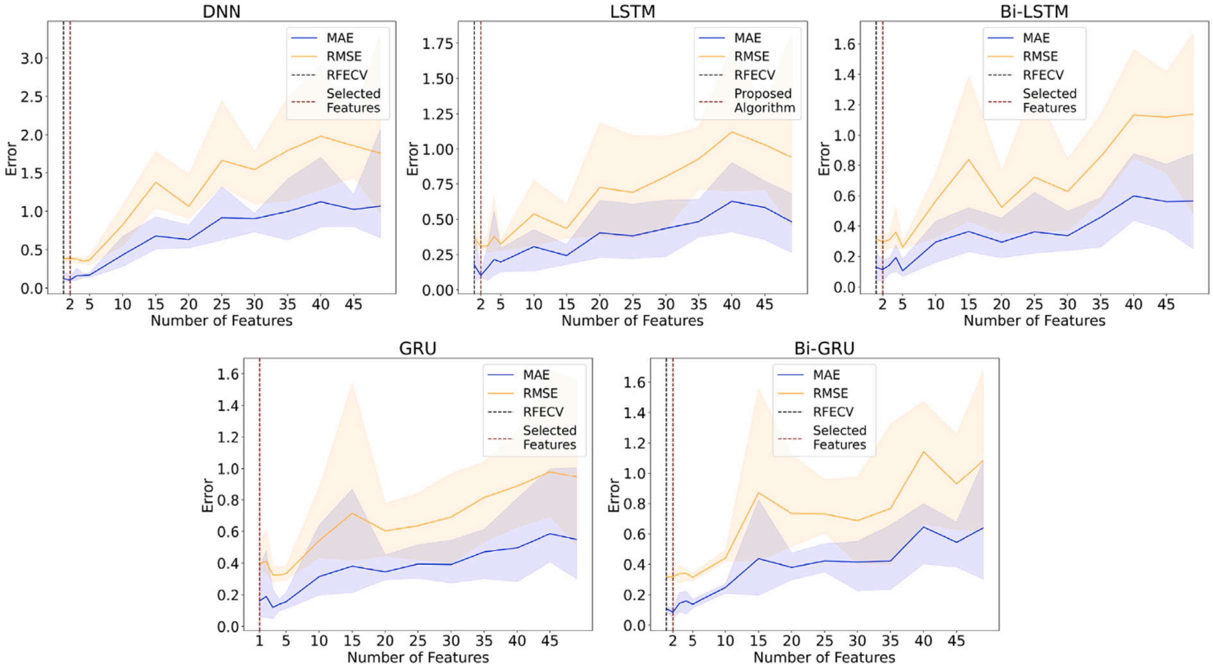

Fig. 6. Model performance for different deep learning architectures (i.e., DNN, LSTM, Bi-LSTM, GRU, Bi-GRU) when performing occupant count prediction in the lecture room using different number of features.

its performance against the other deep learning architectures (refer to Fig. 7), the best performing models for each space type are:

- Office: Bi-GRU, which uses two features (i.e., supply air temperature and chilled water energy) to achieve an MAE and RMSE score of 0.116 and 0.326, respectively.
- Library: GRU, which uses one feature (i.e., AHU Fan Energy) to achieve an MAE and RMSE score of 0.160 and 0.331, respectively.
- Lecture room: Bi-GRU, which uses two features (i.e., number of Wi-Fi connected devices and lighting energy) to achieve an MAE and RMSE score of 0.085 and 0.318, respectively.

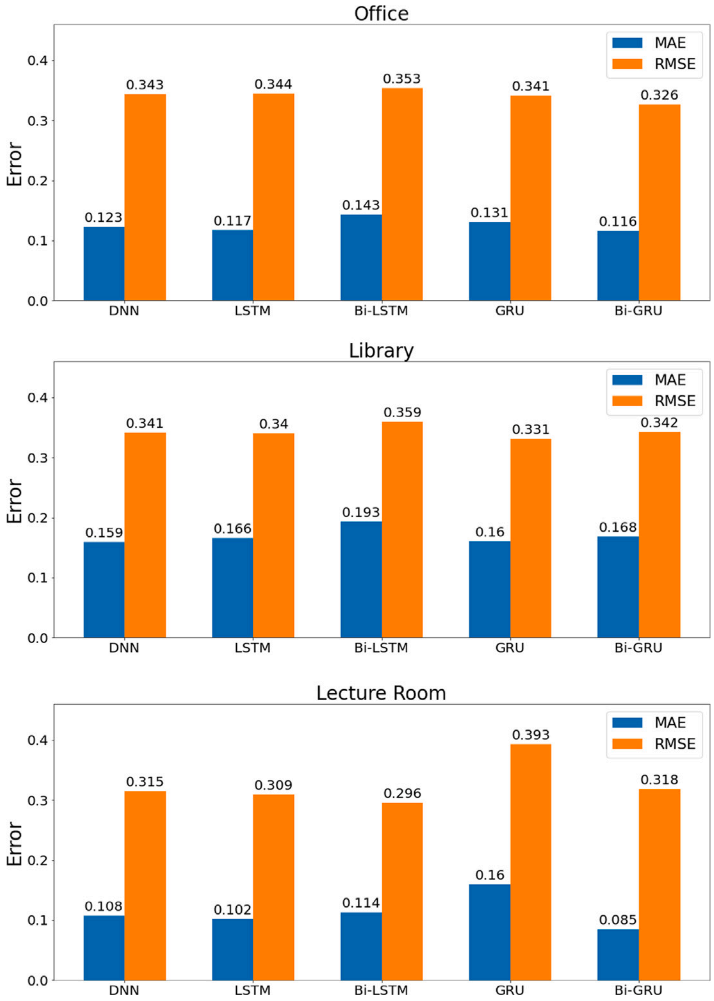

Fig. 7. Model performances for different deep learning architectures for the office, library and lecture room.

Based on the best-performing models identified in Fig. 7, the average predictions made by the selected models for each space type are compared against the averaged ground truth information in the test dataset and plotted in Fig. 8. It can be observed from Fig. 8 that the selected models were able to predict the occupancy levels in their respective space types accurately. Furthermore, several interesting observations can be made based on the occupancy profiles for each space type. For instance, the occupants in the office space tend to follow a regular working schedule, which involves arriving at the office around 10 AM, leaving for their lunch break between 12 PM and 1 PM, and ending their work before 7 PM. On the other hand, the occupancy profile for the library follows a different pattern where there is a sudden spike at 9 AM and a considerable drop at 6 PM, indicating the start and end of the library's operating hours. The library's average occupancy level also tends to increase after 3 PM, which could be explained by the students moving to the library to continue their revision after attending their morning and early afternoon classes. Lastly, the occupancy profile of the lecture room shows several peaks throughout the day to represent regular lectures that are usually scheduled during the morning, early afternoon and evening periods. It can also be observed from Fig. 8 that the occupancy level of the lecture room continues to fluctuate after the dinner period as the space switches into a study area for the students when there are no lectures scheduled.

5. Conclusion

In this study, we performed occupant count predictions by applying a novel feature selection algorithm on a comprehensive sensor dataset containing indoor environmental and outdoor weather condition data, number of Wi-Fi connected devices, energy consumption data, HVAC operations, and time-related information from three space types (i.e., office, library, and lecture room). Several popular deep learning architectures were also implemented and evaluated in this study, including DNN, LSTM, Bi-LSTM, GRU, and Bi-GRU. Finally, a comprehensive analysis was conducted to (1) identify the most crucial features for occupant count prediction in an office, library, and lecture room, (2) identify the optimal number of features for each space type using the proposed feature selection algorithm compared to RFECV, and (3) identify the best model architecture for each space type based on model performance and sensing requirements.

Our empirical results highlighted that the proposed feature selection algorithm outperformed the baseline feature selection algorithm (i.e., RFECV) consistently by achieving a higher model performance while maintaining a significantly lower sensing requirement. Furthermore, empirical results showed that the indoor CO_2 levels and the number of Wi-Fi connected devices were always among the top 15 most crucial features for occupancy prediction across all space types, with the best model performances achieved by Bi-GRU for the office, GRU for the library, and Bi-GRU for the lecture room. Lastly, it should be highlighted that the proposed feature selection algorithm is highly applicable as it can be directly applied to other structured datasets from other domains with minimal changes to the algorithm.

Through the insights gained in this study, building managers and researchers can better identify the most crucial features for occupant count predictions to reduce the need for expensive sensors and minimise deployment costs. Future directions of this work can also strive

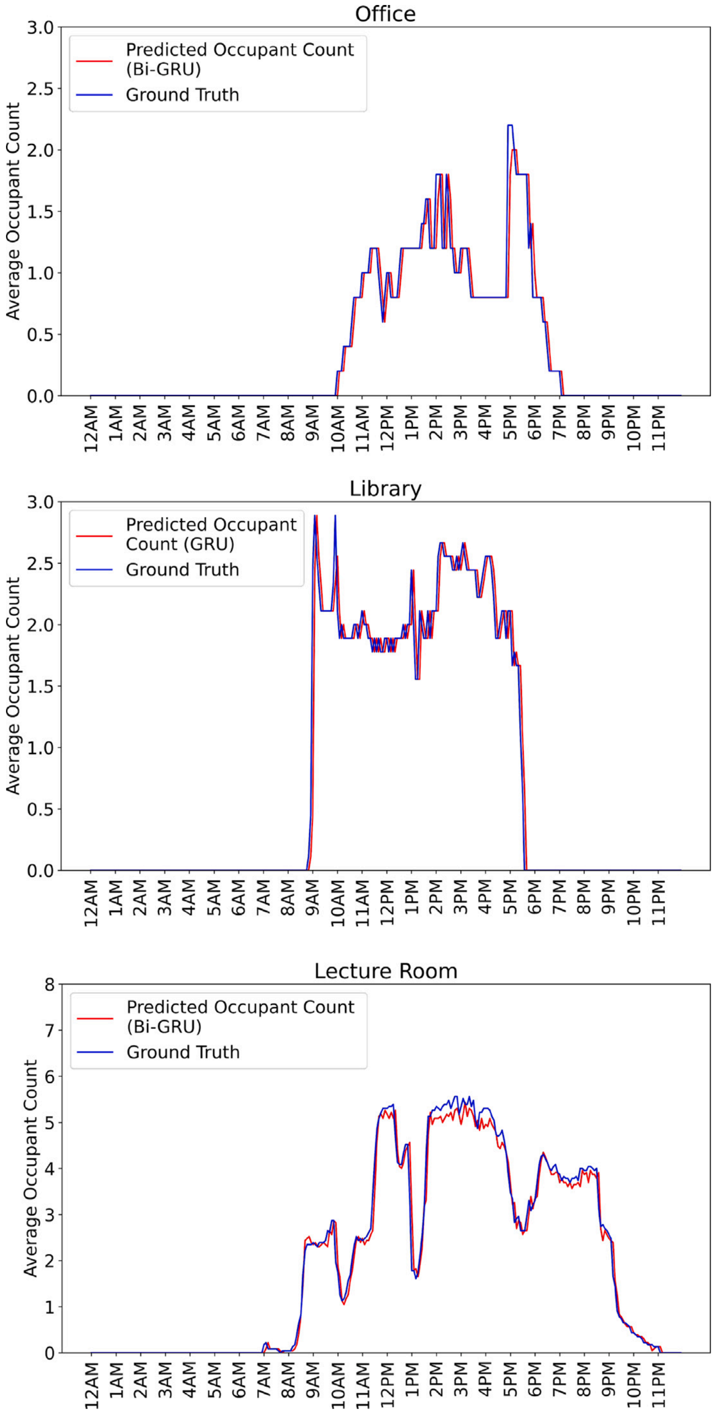

Fig. 8. Comparison of the average predictions made by the best performing models selected from each space type (i.e., office, library, lecture room) against the averaged ground truth information.

towards further improving the generalisability of the study's findings by analysing other space types typically found in a building to support building-wide implementations of such occupancy prediction systems.

CRediT authorship contribution statement

Zeynep Duygu Tekler: Conceptualization, Methodology, Investigation, Data curation, Visualization, Writing – original draft, Writing – review & editing. **Adrian Chong:** Writing – review & editing, Supervision, Funding acquisition, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This research project is supported by the National Research Foundation, Singapore, and Ministry of National Development, Singapore under its Cities of Tomorrow R&D Programme (CoT Award COT-V4-2020-5). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Ministry of National Development, Singapore.

References

[1] C. de Bakker, M. Aries, H. Kort, A. Rosemann, Occupancy-based lighting control in open-plan office spaces: A state-of-the-art review, Build. Environ. 112 (2017) 308–321.

[2] M. Esrafilian-Najafabadi, F. Haghighat, Occupancy-based HVAC control systems in buildings: A state-of-the-art review, Build. Environ. 197 (2021) 107810.

[3] Z.D. Tekler, R. Low, L. Blessing, User perceptions on the adoption of smart energy management systems in the workplace: Design and policy implications, Energy Res. Soc. Sci. 88 (2022) 102505.

[4] S. Azimi, W. O'Brien, Fit-for-purpose: Measuring occupancy to support commercial building operations: A review, Build. Environ. (2022) 108767.

[5] C. Tagliaro, Y. Zhou, Y. Hua, A change in granularity: Measure space utilization through smart technologies, Facilities (2020).

[6] R. Low, Z.D. Tekler, L. Cheah, An end-to-end point of interest (POI) conflation framework, ISPRS Int. J. Geo-Inf. 10 (11) (2021) 779.

[7] C. Ataman, B. Tuncer, Urban interventions and participation tools in urban design processes: A systematic review and thematic analysis (1995–2021), Sustainable Cities Soc. 76 (2022) 103462.

[8] R. Melfi, B. Rosenblum, B. Nordman, K. Christensen, Measuring building occupancy using existing network infrastructure, in: 2011 International Green Computing Conference and Workshops, IEEE, 2011, pp. 1–8.

[9] Z.D. Tekler, R. Low, B. Gunay, R.K. Andersen, L. Blessing, A scalable bluetooth low energy approach to identify occupancy patterns and profiles in office spaces, Build. Environ. 171 (2020) 106681.

[10] N. Li, B. Becerik-Gerber, Performance-based evaluation of RFID-based indoor location sensing solutions for the built environment, Adv. Eng. Inform. 25 (3) (2011) 535–546.

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

[12] Z.D. Tekler, R. Low, L. Blessing, An alternative approach to monitor occupancy using bluetooth low energy technology in an office environment, J. Phys.: Conf. Ser. 1343 (1) (2019) 012116.

[13] N. Nassif, A robust CO2-based demand-controlled ventilation control strategy for multi-zone HVAC systems, Energy Build. 45 (2012) 72–81.

[14] Y.P. Raykov, E. Ozer, G. Dasika, A. Boukouvalas, M.A. Little, Predicting room occupancy with a single passive infrared (PIR) sensor through behavior extraction, in: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, 2016, pp. 1016–1027.

[15] O. Shih, A. Rowe, Occupancy estimation using ultrasonic chirps, in: Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems, 2015, pp. 149–158.

[16] S. Uziel, T. Elste, W. Kattanek, D. Hollosi, S. Gerlach, S. Goetze, Networked embedded acoustic processing system for smart building applications, in: 2013 Conference on Design and Architectures for Signal and Image Processing, IEEE, 2013, pp. 349–350.

[17] D. Liu, X. Guan, Y. Du, Q. Zhao, Measuring indoor occupancy in intelligent buildings using the fusion of vision sensors, Meas. Sci. Technol. 24 (7) (2013) 074023.

[18] Z.D. Tekler, R. Low, C. Yuen, L. Blessing, Plug-mate: An IoT-based occupancy-driven plug load management system in smart buildings, Build. Environ. 223 (2022) 109472.

[19] A. Arora, M. Amayri, V. Badarla, S. Ploix, S. Bandyopadhyay, Occupancy estimation using non intrusive sensors in energy efficient buildings, in: 14th Conference of International Building Performance Simulation Association, Hyderabad, India, Dec. 7-9, 2015, 2015.

[20] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[21] Z.D. Tekler, R. Low, Y. Zhou, C. Yuen, L. Blessing, C. Spanos, Near-real-time plug load identification using low-frequency power data in office spaces: Experiments and applications, Appl. Energy 275 (2020) 115391.

[22] R. Low, L. Cheah, L. You, Commercial vehicle activity prediction with imbalanced class distribution using a hybrid sampling and gradient boosting approach, IEEE Trans. Intell. Transp. Syst. 22 (3) (2020) 1401–1410.

[23] J. Cao, Z. Li, J. Li, Financial time series forecasting model based on CEEMDAN and LSTM, Physica A 519 (2019) 127–139.

[24] Z. Karevan, J.A. Suykens, Transductive LSTM for time-series prediction: An application to weather forecasting, Neural Netw. 125 (2020) 1–9.

[25] K.P. Lam, M. Höynck, B. Dong, B. Andrews, Y.-S. Chiou, R. Zhang, D. Benitez, J. Choi, et al., Occupancy detection through an extensive environmental sensor network in an open-plan office building, IBPSA Build. Simul. 145 (2009) 1452–1459.

[26] A. Vela, J. Alvarado-Uribe, M. Davila, N. Hernandez-Gress, H.G. Ceballos, Estimating occupancy levels in enclosed spaces using environmental variables: A fitness gym and living room as evaluation scenarios, Sensors 20 (22) (2020) 6579.

[27] E. Hitimana, G. Bajpai, R. Musabe, L. Sibomana, J. Kayalvizhi, Implementation of IoT framework with data analysis using deep learning methods for occupancy prediction in a building, Future Internet 13 (3) (2021) 67.

[28] Z. Chen, C. Jiang, M.K. Masood, Y.C. Soh, M. Wu, X. Li, Deep learning for building occupancy estimation using environmental sensors, in: Deep Learning: Algorithms and Applications, Springer, 2020, pp. 335–357.

[29] Z. Tekler, R. Low, L. Blessing, Using smart technologies to identify occupancy and plug-in appliance interaction patterns in an office environment, IOP Conf. Ser.: Mater. Sci. Eng. 609 (6) (2019) 062010.

[30] R. Razavi, A. Gharipour, M. Fleury, I.J. Akpan, Occupancy detection of residential buildings using smart meter data: A large-scale study, Energy Build. 183 (2019) 195–208.

[31] S. Park, K. Kwon, E. Lee, S. Kim, Y. Kim, LSTM-based office occupancy detection using smart plug data, in: 2021 International Conference on Information and Communication Technology Convergence, ICTC, IEEE, 2021, pp. 1707–1709.

[32] S.H. Ryu, H.J. Moon, Development of an occupancy prediction model using indoor environmental data based on machine learning techniques, Build. Environ. 107 (2016) 1–9.

[33] P. Liu, S.-K. Nguang, A. Partridge, Occupancy inference using pyroelectric infrared sensors through hidden Markov models, IEEE Sens. J. 16 (4) (2015) 1062–1068.

[34] B. Huchuk, S. Sanner, W. O'Brien, Comparison of machine learning models for occupancy prediction in residential buildings using connected thermostat data, Build. Environ. 160 (2019) 106177.

[35] Y. Jin, D. Yan, X. Kang, A. Chong, S. Zhan, et al., Forecasting building occupancy: A temporal-sequential analysis and machine learning integrated approach, Energy Build. 252 (2021) 111362.

[36] L. Zimmermann, R. Weigel, G. Fischer, Fusion of nonintrusive environmental sensors for occupancy detection in smart homes, IEEE Internet Things J. 5 (4) (2017) 2343–2352.

[37] W. Wang, T. Hong, N. Xu, X. Xu, J. Chen, X. Shan, Cross-source sensing data fusion for building occupancy prediction with adaptive Lasso feature filtering, Build. Environ. 162 (2019) 106280.

[38] M.K. Masood, Y.C. Soh, C. Jiang, Occupancy estimation from environmental parameters using wrapper and hybrid feature selection, Appl. Soft Comput. 60 (2017) 482–494.

[39] Z. Chen, M.K. Masood, Y.C. Soh, A fusion framework for occupancy estimation in office buildings based on environmental sensor data, Energy Build. 133 (2016) 790–798.

[40] Z.D. Tekler, E. Ono, Y. Peng, S. Zhan, B. Lasternas, A. Chong, ROBOD, room-level occupancy and building operation dataset, in: Building Simulation, Springer, 2022, pp. 1–11.

[41] D.J. Stekhoven, P. Bühlmann, MissForest—Non-parametric missing value imputation for mixed-type data, Bioinformatics 28 (1) (2012) 112–118.

[42] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (1) (2002) 389–422.

[43] R. Low, Z.D. Tekler, L. Cheah, Predicting commercial vehicle parking duration using generative adversarial multiple imputation networks, Transp. Res. Rec. 2674 (9) (2020) 820–831.

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

[45] H.S. Gill, B.S. Khehra, An integrated approach using CNN-RNN-LSTM for classification of fruit images, Mater. Today: Proc. 51 (2022) 591–595.

[46] O. Atila, A. Şengür, Attention guided 3D CNN-LSTM model for accurate speech based emotion recognition, Appl. Acoust. 182 (2021) 108260.

[47] Z. Huang, W. Xu, K. Yu, Bidirectional LSTM-CRF models for sequence tagging, 2015, arXiv preprint arXiv:1508.01991.

[48] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014, arXiv preprint arXiv:1412.3555.