Exploratory Data Analysis on Open Heterogeneous Building Occupancy Datasets

Georgiana Cretu
Automation and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
georgiana.cretu@upb.ro

Iulia Stamatescu
Automation and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
iulia.stamatescu@upb.ro

Grigore Stamatescu
Automation and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
grigore.stamatescu@upb.ro

***Abstract*—The development of smart buildings and advanced technologies such as Internet of Things (IoT) systems, smart sensors and data analysis methods play a crucial role in achieving a higher level of societal energy efficiency. The optimization of energy consumption by evaluating the number of occupants in buildings and assuring personalized comfort conditions requires the integration of these technologies, promoting innovation in building automation systems, through energy management and control systems. The paper focuses on a detailed analysis of open heterogenous building occupancy datasets from the perspective of a robust data science pipeline. We introduce our methodological contribution, mainly in the form of data pre-processing and standardisation, for the analysis of diverse data types utilized for occupancy estimation in buildings. The proposed algorithm is based on an examination of timely open datasets containing building sensor data, offering a well-informed reference for occupancy estimation and forecasting algorithms.**

***Index Terms*—building automation, exploratory data analysis, data science, occupancy modelling, sensors**

I. Introduction

Public statistic reports provide an overview of the important role of buildings and their occupants in addressing global environmental challenges. Overall, 36% of global final energy consumption is jointly attributed to existing buildings and building construction, as well as approximately 40% of total direct and indirect CO2 emissions [1]. In 2050, the share of space heating in the total energy consumption of buildings will represent 48%, followed by others (25%), lighting (13%), water heating (8%) and air cooling (6%) [1]. The topic of optimizing energy consumption in smart buildings through the personalized assessment of comfort conditions and the number of occupants is of significant importance from various perspectives, including from the scientific, technological and socio-economic point of view. Occupancy-based HVAC (Heating, Ventilation and Air Conditioning) Control (OBC) [2] takes into account actual and/or predicted indoor occupancy in generating optimal setpoints for the low-level control loops of the building automation system (BAS).

Analyzing the current state of knowledge related to the proposed topic, several new developments on large-scale energy monitoring inside buildings using complementary approaches have been published. Also, in recent years, the number of scientific articles studying the impact of occupant behavior and building occupancy on the energy consumption of buildings has increased, indicating a growing interest in the energy efficiency of buildings focused on the human factor. The current state of the art in this field involves the integration of advanced technologies, data analysis and customized approaches to achieve energy efficiency while ensuring optimal occupant comfort [3].

A driving factor in OBC implementations is the use of sensor networks and Internet of Things (IoT) devices to collect real-time data on various parameters such as temperature, humidity, occupancy and lighting levels, as well as information from wearables type devices such as smart watches and smart rings. These sensors allow continuous and personalised monitoring of comfort conditions and provide valuable information for energy optimization [4]. In addition, machine learning techniques are used to analyze the collected data and develop predictive models for building occupancy as well as models for energy consumption patterns. These models can identify potential energy-saving opportunities and provide customized adjustments to HVAC systems, lighting controls, and other building automation systems, enhancing comfort while minimizing energy waste [5], [6].

Our main contribution is defined with regard to the methodology for analysing various types of data used in accurately modelling and predicting the occupancy in buildings. We introduce a study of recent open datasets for building sensor data, as informed reference that can be used by occupancy estimation and forecasting algorithms. An additional argument is that, given the heterogeneous characteristics of these datasets such as, number and type of sensors, direct and indirect measurements, experimentation environment and others, robust techniques are required to bridge these in order to increase the quantity of available quality training data for subsequent data-driven machine learning algorithms. [1] The rest of the paper is structured as follows. Section II frames our contribution within the state-of-the-art with regard to estimation of building occupancy. Section III presents the conceptualised methods for data processing highlighting the main issues in selecting the types of measurements and the appropriate context along with the analysis of several public data sets available for building occupancy modeling. The availability of data sets in the challenge of detecting the degree of occupancy in buildings is very important in the creation of algorithms that use complex models, such as deep neural networks, given that, in many situations the collection, handling and pre-processing of data sets requires significant time and computational resources resources. Main findings are highlighted in Section IV through exemplification of the

[1] 979-8-3503-8215-0/24/$31.00 ©2024 IEEE

implemented methods. Section V concludes the paper with outlook on future work.

II. Related Work

Smart buildings have evolved through the use of Internet of Things (IoT) systems for environmental monitoring and automatic control of various functional subsystems. These systems collect representative data on a high temporal and geographical scale, allowing efficient operation of buildings. An important aspect in this regard is the accurate detection of occupancy in buildings, as a factor contributing to reducing the energy consumption and associated emissions produced by the building.

There are several ways to detect occupancy in buildings, and advanced technologies are used for this purpose. One of the common indirect detection methods is the use of environmental sensors, which can provide information about the presence of humans in a certain area. These sensors can detect motion, light, temperature and CO2 levels and can feed data for neural network training to learn and detect occupancy [7], [8], [9]. The use of indirect sensing is also motivated by the fact that these systems do not raise problems related to the user's confidence, reliability and privacy. In addition, to maximize the precision of detection in real time, the systems that combine environmental sensors and neural networks present the advantage of reduced execution time and more efficient use of memory.

Table I summarizes some of the studies in which data collection from environmental sensors was conducted, with the ultimate goal of predicting the number of occupants.

In previous works we have presented an analysis of deep learning model prediction performance for building occupancy estimation [12] and investigated the relative performance of machine learning (ML) techniques such as random forests compared to deep learning techniques (mostly convolutional neural networks) for the same task [13]. In order to improve the performance of such model, this current work tries to raise input data quality and availability for building such robust models.

III. Methodology

Occupancy modelling and prediction applications involve several stages, as can be seen in Figure 1. Even though model selection is typically considered as the most important stage, we argue that data (pre-)processing also requires significant attention for good robust performance. Thorough data processing leads to appropriate inputs into the predictive model, and together with a high-performing model, the prediction results will be satisfactory.

Data processing involves multiple levels: data cleaning, data reduction, data dividing, data balancing and data splitting, which are subsequently discussed in detail.

A. Data Cleaning

One of the most common issues encountered with open datasets is the absence of information at certain moments. Missing values can be caused by a variety of factors, such as faulty sensors, human error, or data corruption. These missing values need to be identified and either imputed i.e. filled in with a suitable value, or removed altogether from the dataset.

Because filling in missing values is generally quite challenging as the values around it needed to be analyzed to determine the missing value, the implemented pre-processing algorithm removes the entire measurement where information is missing from sensors. This is possible if there are few missing values, and the dataset is large enough not to be affected.

Another issue that can arise during data collection is the presence of identical measurements. Duplicate values can occur due to errors in data collection or data entry. These duplicates can introduce biases and distort the analysis or modeling process. Identifying and removing these duplicates can help improve the accuracy of the dataset. This can be achieved through various techniques such as record comparison or automated algorithms. Since this problem is time-consuming because it compares each entry, our system has not yet implemented an algorithm of this type.

Data cleaning also involves identifying and handling outliers in the dataset. Outliers are data points that are significantly different from other data points in the dataset. Outliers can occur due to measurement errors, incorrect data entry, or rare events. Identifying and handling outliers is important to avoid them having an undue influence on the analysis. One idea for identifying them is to generate plots of each of the measurements in the dataset, thus showing the extremes of each type of sensor.

Datasets can naturally contain values from environmental sensors with different scales. For this reason, value scaling is necessary for such an approach. Also, data collected from different sources can have different scales and units, which can make it difficult to compare or analyze them. Standardising the data by transforming it to a common scale or unit can help improve the consistency and comparability of the data.

The last issue discussed during the data cleaning process is the conversion of data with wrong formatting. Sometimes the data may be in the wrong format, such as numeric data recorded as text, or dates recorded in the wrong format. Converting the data to the correct format can help ensure that it is usable for analysis. Another type of conversion could be when attempting different types of datasets that have a slightly different structure. For example, in the public datasets used, there is exact information about the number of occupants, but to use the dataset in the proposed algorithm, the exact number of occupants has been converted to occupancy classes.

B. Data reduction

The algorithm used in this level is reducing the raw dataset by extracting measurements at a specific step. This idea is very straightforward, but it comes with an issue: you cannot choose which measurements to keep. To avoid losing information that is skipped during the reading step, a weighted average of the values in the skipped interval can be calculated. Instead of considering only one value at a certain step, a weighted average is calculated from the values of the passive sensors. Weighted average can be chosen experimentally, either linear or sinusoidal.

In the preprocessing algorithm constructed, linear, sinusoidal and cosinusiodal weighted average were attempted using the formulas (1):

$$LWA = \frac{\sum_{i=1}^{N} S_i \cdot W_i}{\sum_{i=1}^{N} W_i} \tag{1}$$

TABLE I
THE RELEVANT INDOOR ENVIRONMENTAL QUALITY FACTORS AND CORRESPONDING STUDIES

Study Title	IEQ Factor	Measured Parameters	Device
Indoor Environmental Quality Assessment and Occupant Satisfaction, 2022 - [10]	Thermal Comfort Indoor Air Quality Lighting Quality	Temperature Relative Humidity PM2.5, PM10, CO_2, TVOCs Lux Level	HOBBO HOBBO Air Mentor Pro Precision Gold Environment Meter
A high-fidelity residential building occupancy detection dataset, 2021 - [8]	Thermal Comfort Indoor Air Quality Lighting Quality	Temperature Relative Humidity CO_2, TVOCs Lux Level	Aosong DHT22 Aosong DHT22 Sensirion SGP30 Avago APDS-9301
Accurate occupancy detection of an office room from light, temperature, humidity, and CO2 measurements using statistical learning models, 2016 - [11]	Thermal Comfort Indoor Air Quality Lighting Quality	Temperature Relative Humidity CO_2 Lux Level	DHT22 DHT22 Telaire 6613 TSL2561

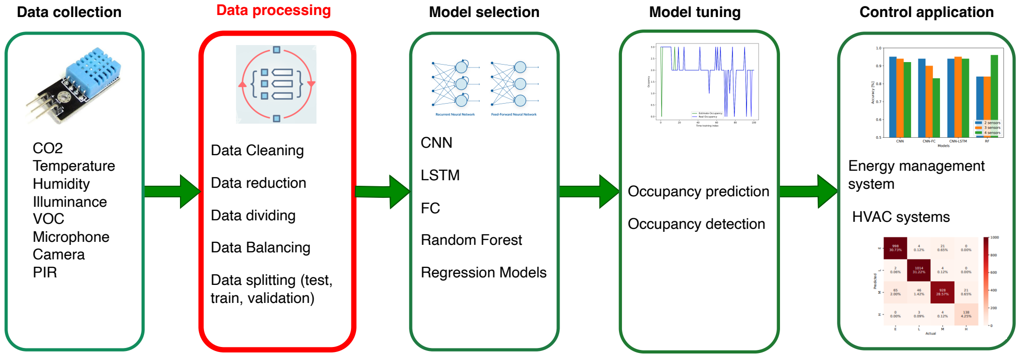

Fig. 1. The stages of a building occupancy modelling and prediction system

Where LWA is the value of weighted average method that will be used at training and validation model, N is the read step from the raw dataset, W_i is the weight value (the number of weights are selected in a stepwise manner) and S_i is the value of each sensor type [14].

The difference between the averages mentioned above is how the values of the weights are computed. For linear weights, the values were calculated using the random function on the interval [0; 10], followed by sorting the values. For the rest of type weights, the values were calculated using the uniform random function on the interval [0, π] – sinusoidal and [-π/2; π/2] – cosinusiodal, followed by applying the sine or cosine function to these values (formula 2 and 3).

$$W_{sin} = sin(random(0, \pi, N)) \tag{2}$$

$$W_{cos} = cos(random(\frac{-\pi}{2}, \frac{\pi}{2}, N)) \tag{3}$$

Where W_{sin} and W_{cos} are the weight values used and step is the reading step from raw dataset.

Reducing the dataset is only possible if it contains a very large volume of measurements, and training and evaluating the network would require a significant amount of time and computational resources.

C. Data dividing

Data dividing involves splitting it into fixed-length sequences. A neural network that receives sequences of data can be trained so that the evolution of the values collected from the sensors has a greater impact.

Splitting the input into smaller sequences for a neural network impacts memory management and enables training on smaller batches and parallel processing, enhancing overall efficiency. This approach is valuable for handling long sequences and simplifying the model's complexity.

In our application, the sequences were created based on a sliding window that contains 60 measurements and a duration of 15 – 20 minutes. For example, the first sequences were created with the following measurements: 0 – 60, 1 – 61, 2 – 62, 3- 63, etc. The process of sliding window can be better understood by exemplifying a sequence of 7 measurements in figure 2. Each number represents the values from the sensors for a measurement.

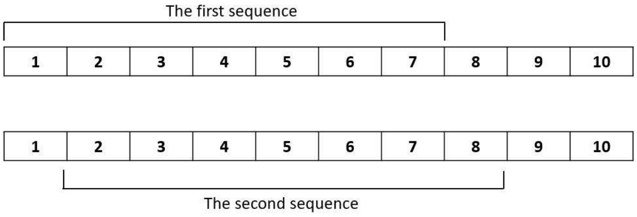

Fig. 2. Sliding window mechanism

The dataset must contain consecutive time data, for example if the difference time between two values is one hour, the algorithm will automatically create a new sequence starting from that value.

Another criterion used in dividing the data into sequences is the minimum number of measurements per sequence. When the current sequence ends due to a large time difference (one hour) and a new sequence begins, the length of both sequences can be smaller than the predetermined size of 60. Therefore, we have set a minimum number of measurements per sequence, which is equal to three-quarters of the maximum size:

$$Min_{seq} = (max\_sequence * 3)/4 \tag{4}$$

For a sequence with 60 measurements, the minimum number is equal to 45 measurements.

D. Data balancing

The next pre-processing step that has been performed and has a significant impact is balancing the dataset. If there are significant differences between the number of sequences in each prediction class: Empty, Low, Medium and High, the accuracy of the trained model can be influenced by these differences or there may be an overfitting problem where the model is trained more on a single class. The algorithm for balancing the dataset is the following:

- Compute the minimum (Min_{seq}) between the sequence sizes of the four prediction classes.

$$Min_{class} = Min(Seq_E, Seq_L, Seq_M, Seq_H) \tag{5}$$

- Compute the new sequence size for the classes that are a size very high compared to the minimum class size. Every new size is computed randomly based on an interval chosen experimentally: the lower value is 1 and the upper is 4, as it is also in the formula 6.

$$Length_{class} = int(random(l * Min_{class}, h * Min_{class})) \tag{6}$$

E. Data splitting

The final step in pre-processing the datasets consists of dividing the dataset into train, validation and test, thus suitable to the modelling stage through machine learning algorithms.

In general, before the actual occupancy estimation algorithm, the dataset is divided into training the network, another part is used for testing it, and the final part for validating the neural network. The largest volume of data is sent for training, specifically 60% of the dataset, while testing and validation have the same volume of data, which is 20%.

These percentage values should be carefully chosen so that there are enough data for training the network, as well as for testing and validation. After splitting the data, we have chosen to write them into a CSV file, making it easy to visualize them.

IV. Results

For the practical evaluation, the algorithms for pre-processing were implemented using the Python programming language, along with various specialised software libraries: datetime - for date and time conversion, os - useful functions for interactions with the operating system, json and csv - for parsing and reformatting structured data.

The reference code implementation is available on Github[2] for testing the processing and replicating the presented results. The main step of the work involves processing raw data before using it in neural network models that predict occupancy levels in buildings, such as cleaning, reduction, balancing, dividing and splitting. The datasets used are from Jacoby's work [8], which includes measurements from six residential houses. In each house, three to five hubs are installed, through which information about temperature, pressure, light, CO2 concentration, volatile organic compounds, and occupant presence is collected.

Pre-processing was required so that they could be used as inputs to the occupancy prediction app in buildings. The raw data is organized on 2 types of ".csv" files according to the information contained: the number of occupants, respectively the measurements from the environmental sensors. Pre-processing consists of extracting information of interest from "*.csv" files and converting certain measurements. The conversion of the information about the actual occupancy was made as follows:

- For 0 occupants the classification "empty" was chosen;
- For 1 occupant was chosen classification "low";
- For 2 occupants the classification "medium" was chosen;
- For 3, 4 or 5 occupants the "high" classification was chosen.

The "*.csv" files were created based on the day when the measurements were taken, so each day of each apartment, among the six included in the study, corresponds to a file. Since the data contained in a single file, representing a single day, would have been too small, and predicting occupancy levels from such limited data could lead to subjectivity (overfitting), a script was created. This script can combine multiple files, either creating datasets from multiple days or datasets from different areas within the apartment. For example, combining the hub positioned in the living room with the one in the kitchen can expand the coverage area, avoiding data overlap.

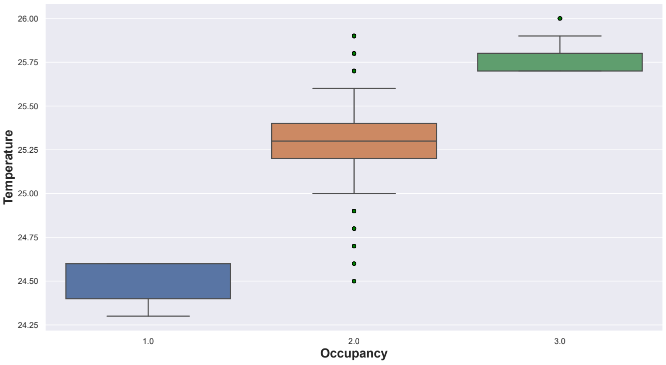

Fig. 3. Average temperature measurements

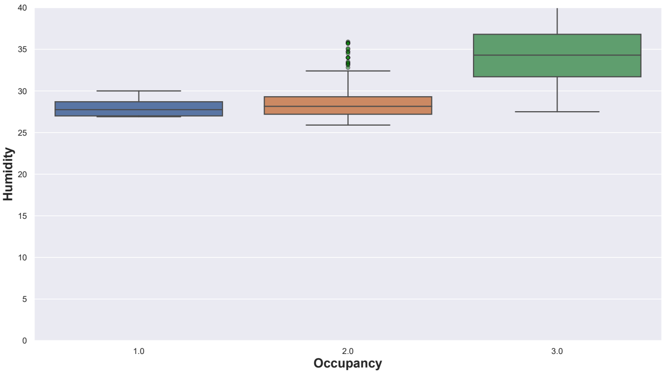

Fig. 4. Average humidity measurements

[2] https://github.com/gmcretu/IEEE_EnergyCon2024

TABLE II
DATASET PROPERTIES

Dataset	Timestep [s]	Data Splitting	Examples per occupancy class
2019-12-16-21-RS123-H1	20	train:26671, test:8895, valid:8890	'empty': 13311, 'low': 8046, 'medium': 8586, 'high': 14513
2019-12-09-14-RS14-H1	20	train:28866, test:9625, valid:9621	'empty': 14079, 'low': 11830, 'medium': 6522, 'high': 15681
2019-11-26-02-RS245-H1	20	train:29531, test:9848, valid:9842	'empty': 13974, 'low': 8886, 'medium': 11734, 'high': 14627
2019-12-07-14-RS135-H1	20	train:49439, test:16483, valid:16479	'empty': 21116, 'low': 23160, 'medium': 10308, 'high': 27817
2019-08-31-04-RS345-H3	20	train:26432, test:8815, valid:8809	'empty': 10242, 'low': 7524, 'medium': 7113, 'high': 19177
2019-08-30-04-RS215-H3	20	train:25949, test:16223, valid:22704	'empty': 11619, 'low': 9585, 'medium': 8919, 'high': 34753
2019-05-05-09-RS345-H4	20	train:32378, test:10796, valid:10792	'empty': 17220, 'low': 22578, 'medium': 7895, 'high': 6273

TABLE III
ACC AND LOSS AFTER CLEANING, REDUCING, DIVIDING AND SPLITTING (60%, 20%, 20%), AND BEFORE BALANCING

Data	No Examples	CNN		CNN-FC		CNN-LSTM	
		A	$\mathcal{L}$	A	$\mathcal{L}$	A	$\mathcal{L}$
2019-12-16-21-RS123-H1	64625	**0.8647**	**0.4273**	**0.8490**	**0.5029**	**0.8396**	**0.5019**
2019-12-09-14-RS14-H1	51722	0.9534	0.1254	0.9662	0.0827	0.9702	0.0814
2019-11-26-02-RS245-H1	90545	0.9013	0.2615	0.9206	0.2149	0.9144	0.242
2019-12-07-14-RS135-H1	103505	0.9099	0.2527	0.93073	0.2120	0.9037	0.2679
2019-08-31-04-RS345-H3	64625	0.9658	0.0945	0.9562	0.1115	0.9678	0.0940
2019-08-30-04-RS215-H3	77585	0.9059	0.2525	0.8863	0.3002	0.8958	0.2949
2019-05-05-09-RS345-H4	64568	**0.7977**	**0.5349**	**0.75814**	**0.6889**	**0.8042**	**0.5349**

TABLE IV
ACC AND LOSS AFTER CLEANING, REDUCING,BALNACING, DIVIDING AND SPLITTING (45%, 35%, 20%)

Data	No Examples	CNN		CNN-FC		CNN-LSTM	
		A	$\mathcal{L}$	A	$\mathcal{L}$	A	$\mathcal{L}$
2019-12-16-21-RS123-H1	44456	**0.8009**	**0.7558**	**0.8095**	**0.5984**	**0.8149**	**0.7392**
2019-12-09-14-RS14-H1	48112	0.9794	0.0707	0.9823	0.0500	0.9762	0.0753
2019-11-26-02-RS245-H1	49221	0.9092	0.2717	0.9276	0.2091	0.9132	0.2636
2019-12-07-14-RS135-H1	82401	0.9060	0.2542	0.9154	0.2390	0.9034	0.2672
2019-08-31-04-RS345-H3	44056	0.9504	0.139	0.9622	0.1048	0.9491	0.1410
2019-08-30-04-RS215-H3	64876	0.8915	0.2910	0.9108	0.2559	0.8887	0.3138
2019-05-05-09-RS345-H4	53966	**0.8259**	**0.4986**	**0.7987**	**0.5434**	**0.8153**	**0.4778**

TABLE V
ACC AND LOSS AFTER CLEANING, REDUCING, DIVIDING AND SPLITTING (60%, 20%, 20%), AND AFTER BALANCING

Data	No Examples	CNN		CNN-FC		CNN-LSTM	
		A	$\mathcal{L}$	A	$\mathcal{L}$	A	$\mathcal{L}$
2019-12-16-21-RS123-H1	44456	**0.9369**	**0.1724**	**0.9279**	**0.1873**	**0.9219**	**0.2157**
2019-12-09-14-RS14-H1	48112	0.9826	0.0520	0.9860	0.0432	0.9823	0.4960
2019-11-26-02-RS245-H1	49221	0.9160	0.2354	0.9338	0.1945	0.9235	0.2217
2019-12-07-14-RS135-H1	82401	0.9197	0.2228	0.9248	0.2086	0.9245	0.2219
2019-08-31-04-RS345-H3	44056	0.9549	0.1233	0.9706	0.0804	0.9719	0.0834
2019-08-30-04-RS215-H3	64876	0.9126	0.2430	0.9164	0.2190	0.9091	0.2467
2019-05-05-09-RS345-H4	53966	**0.8339**	**0.4251**	**0.8227**	**0.49060**	**0.82197**	**0.4344**

The figures that have been created are in the form of box plots, and highlight the values from each environmental sensor (temperature - Figure 3, humidity - Figure 4, light - 5 and CO2 values - 6) and their relation to the ground truth i.e. number of occupants. With the increase in the number of occupants, a slight increase in temperature, humidity or CO2 concentration value is also observed, the Figure 7 presenting the same idea. This observation highlights the direct impact of human activities on the environment inside buildings. Therefore, the generation of heat and humidity associated with the presence of more people leads to significant adjustments of ambient conditions. Also, increasing the CO2 level may indicate the need for the use of ventilation and air conditioning appliances. By analysing these changes, you can gain a clearer understanding of how employment affects the environmental conditions in buildings and provide guidance for optimizing them according to the needs of users.

Additionally, Table II provides an overview after the data processing stage of several datasets used, including the number of train, validation and test entries for each set, the timestep, and the distribution of entries relative to the type of occupancy class.

To exemplify the importance of data processing, we trained and evaluated multiple datasets with and without different stages of processing. One of the stages is Data Balancing, Tables III and V presented the accuracy results and loss function with and without this stage. An increase in accuracy is observed, especially for datasets with an accuracy lower than 90%, while for datasets with 95% accuracy, the difference is minimal. This demonstrates that processing can help increase metrics with low values, as expected results. Another stage is Data Splitting, the comparison between different values of the splitting being presented in Table IV and Table V. Two different sets of ratios, namely [45%, 35%, 20%] and [60%, 20%, 20%], were employed for dividing the data into training, validation, and testing subsets. This comparison underscores the importance of the training dataset size, as discussed in Section III.

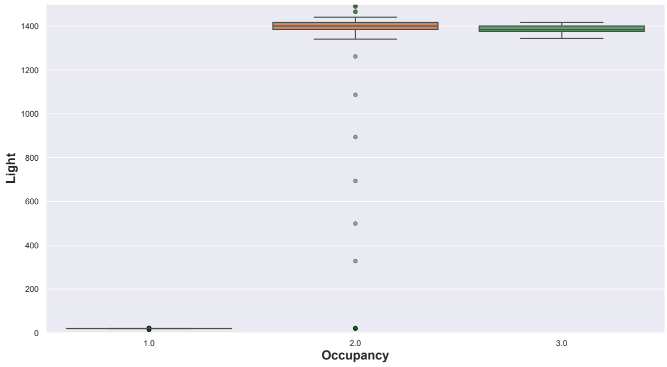

Fig. 5. Average light measurements

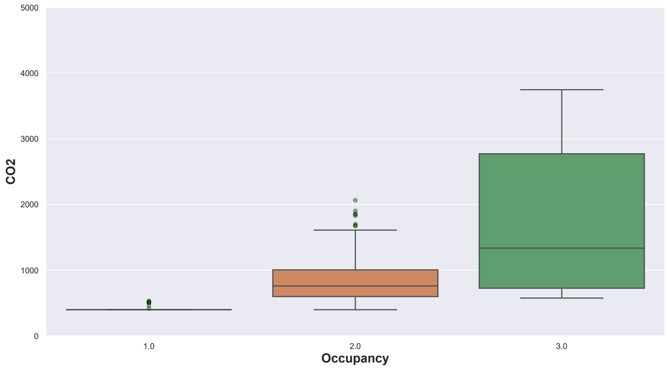

Fig. 6. Average CO2 measurements

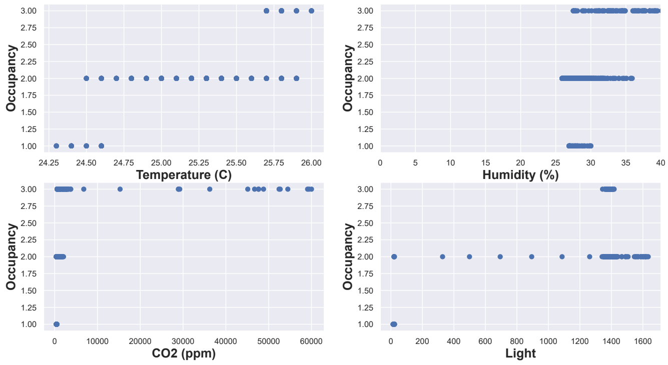

Fig. 7. The number of occupants vs environmental sensors

V. Conclusion

The presented work aims to improve the pre-processing stage in a data-driven modelling pipeline for building occupancy prediction. It has been argued that through better processing and understanding of the input variables, improved forecasting accuracy and robustness of the prediction models can be achieved. The impact of improved data quality for accurate forecasting models, in turn, will lead to higher energy savings in future smart buildings with a direct impact on grid energy management supporting a sustainable energy transition.

Acknowledgment

This work was supported by a grant from the National Program for Research of the National Association of Technical Universities - GNAC ARUT 2023.

References

[1] S. Malla and G. R. Timilsina, "Long-term energy demand forecasting in Romania : an end-use demand," The World Bank, Policy Research Working Paper Series 7697, Jun. 2016. [Online]. Available: https://ideas.repec.org/p/wbk/wbrwps/7697.html

[2] C.-L. Lorenz, M. André, O. Abele, B. Gunay, J. Hahn, P. Hensen, Z. Nagy, M. M. Ouf, J. Y. Park, N. S. Yaduvanshi *et al.*, "A repository of occupant-centric control case studies: Survey development and database overview," *Energy and Buildings*, vol. 300, p. 113649, 2023.

[3] S. Dabirian, K. Panchabikesan, and U. Eicker, "Occupant-centric urban building energy modeling: Approaches, inputs, and data sources - a review," *Energy and Buildings*, vol. 257, p. 111809, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378778821010938

[4] Y. Feng, J. Wang, N. Wang, and C. Chen, "Alert-based wearable sensing system for individualized thermal preference prediction," *Building and Environment*, vol. 232, p. 110047, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360132323000744

[5] X. Ding, A. Cerpa, and W. Du, "Exploring deep reinforcement learning for holistic smart building control," 2023.

[6] Y. Lei, S. Zhan, E. Ono, Y. Peng, Z. Zhang, T. Hasama, and A. Chong, "A practical deep reinforcement learning framework for multivariate occupant-centric control in buildings," *Applied Energy*, vol. 324, p. 119742, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261922010297

[7] B. Dong, Y. Liu, W. Mu, Z. Jiang, P. Pandey, T. Hong, B. Olesen, T. Lawrence, Z. O'Neil, C. Andrews, E. Azar, K. Bandurski, R. Bardhan, M. Bavaresco, C. Berger, J. Burry, S. Carlucci, K. Chvatal, M. De Simone, S. Erba, N. Gao, L. T. Graham, C. Grassi, R. Jain, S. Kumar, M. Kjærgaard, S. Korsavi, J. Langevin, Z. Li, A. Lipczynska, A. Mahdavi, J. Malik, M. Marschall, Z. Nagy, L. Neves, W. O'Brien, S. Pan, J. Y. Park, I. Pigliautile, C. Piselli, A. L. Pisello, H. N. Rafsanjani, R. F. Rupp, F. Salim, S. Schiavon, J. Schwee, A. Sonta, M. Touchie, A. Wagner, S. Walsh, Z. Wang, D. M. Webber, D. Yan, P. Zangheri, J. Zhang, X. Zhou, and X. Zhou, "A global building occupant behavior database," *Scientific Data*, vol. 9, no. 1, p. 369, 2022. [Online]. Available: https://doi.org/10.1038/s41597-022-01475-3

[8] M. Jacoby, S. Y. Tan, G. Henze, and S. Sarkar, "A high-fidelity residential building occupancy detection dataset," *Scientific Data*, vol. 8, no. 1, p. 280, 2021. [Online]. Available: https://doi.org/10.1038/s41597-021-01055-x

[9] M. S. Aliero, M. F. Pasha, D. T. Smith, I. Ghani, M. Asif, S. R. Jeong, and M. Samuel, "Non-intrusive room occupancy prediction performance analysis using different machine learning techniques," *Energies*, vol. 15, no. 23, p. 9231, Dec 2022. [Online]. Available: http://dx.doi.org/10.3390/en15239231

[10] Y. K. Kim, Y. Abdou, A. Abdou, and H. Altan, "Indoor environmental quality assessment and occupant satisfaction: A post-occupancy evaluation of a uae university office building," *Buildings*, vol. 12, no. 7, 2022. [Online]. Available: https://www.mdpi.com/2075-5309/12/7/986

[11] L. M. Candanedo and V. Feldheim, "Accurate occupancy detection of an office room from light, temperature, humidity and co2 measurements using statistical learning models," *Energy and Buildings*, vol. 112, pp. 28–39, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378778815304357

[12] G. Cretu, I. Stamatescu, and G. Stamatescu, "Building occupancy classification from indirect sensing with heterogeneous datasets," in *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, vol. 1, 2021, pp. 475–479.

[13] ——, "Evaluation of deep learning and machine learning algorithms for building occupancy classification on open datasets," in *2023 31st Mediterranean Conference on Control and Automation (MED)*, 2023, pp. 575–580.

[14] H. S. Hota, R. Handa, and A. K. Shrivas, "Time series data prediction using sliding window based rbf neural network," 2017.