ITM Web of Conferences **69**, 01013 (2024) https://doi.org/10.1051/itmconf/20246901013
MAIH2024

Performance evaluation of forecasting strategies for building occupancy prediction

Amine Maniar [1,2*], *Laurent* Delahoche[1], *Larbi* Chrifi-Alaoui[1], *Mourad* Zegrari[2], *Hamlich* Mohamed[2], *Bruno* Marhic[1] and *Jean-Baptiste* Masson[1]

[1]Laboratory of Innovative Technologies, University of Picardie Jules Verne, Amiens, France
[2]Laboratory of Complex Cyber Physical Systems, Hassan II University of Casablanca, Casablanca, Morocco

Abstract. Occupant behavior has been identified as a key factor affecting energy usage in buildings. Integrating occupancy data into HVAC control strategies presents an opportunity for substantial energy savings. The proposed study evaluates different occupancy prediction strategies with a focus on forecasting performance on highly variable signals such as CO2 concentration and noise levels. Our work compares single-step and multiple-steps prediction methods to analyze their impact on accuracy and reliability. The predicted signals can be used to identify future activity to improve occupancy forecasting. In this paper, we highlight the importance of accurate occupancy data and fitting forecasting strategy and propose future research directions to address current limitations in occupancy prediction models.

1 Introduction

Building energy efficiency is a popular domain in both academical and industrial worlds. Energy consumption in buildings represent 40% of total energy consumed, and their CO2 emissions are at 33% of emissions in developed countries **[1]**. The main issue is HVAC (Heating, Ventilation and Air-Conditionning) systems that represent more than half of the energy consumed in commercial buildings **[2], [3]**.There have been numerous simulation software to aid in building conception to reduce energy expenditures, however their predictions only represent one fifth of real consumption because they fail to describe every dynamic that influence energy usage in buildings **[4]**. Taking into account occupant activity could reduce energy consumption by 40% while improving thermal comfort up to 30% **[5]**, this conclusion has driven a lot of research interest for occupancy prediction.

The prediction algorithms discussed in literature usually fall in distinct strategies, like univariate or multivariate. The univariate approach predicts future values for a variable based only on the history of that same variable, whereas the multivariate approach feeds the prediction algorithm with additional features. This last approach is the most popular when using environmental data because a single sensor data cannot reliably be associated with occupancy status. Aside from features number, the horizon of prediction can also be vastly different depending to each model. Occupancy studies usually rely on one-step prediction to predict future values, which implies using a closed loop to feed the model its own predictions, which is bound to accumulate error and therefore unreliable for multiple time steps. Multiple-steps forecasting is much more challenging and rarely used **[6]**.

In this article, we compare two forecasting strategies using accessible prediction models to assess their impact on forecasting performance for occupancy-related variables such as CO2 concentration, noise levels, and temperature. The first strategy is single-step prediction, which involves using historical data to predict a single future value. The second, less commonly used, strategy is multiple-steps prediction, which involves making predictions over a predefined horizon. This comparative analysis highlights the importance of correctly structuring inputs and outputs for forecasting tasks, demonstrating that this is more crucial than the choice of model or hyperparameter tuning. This paper addresses gaps in the literature regarding the significance of forecasting strategy in occupancy prediction and provides guidance for selecting the optimal strategy for signals with high variability. These predictions can then be utilized to identify future occupancy states and implement efficient control strategies for building heating.

This paper is organized as follows: Section II provides an overview of related works in occupancy forecasting using different prediction models. Section III presents the data used in our experiments along with its preprocessing. Section IV describes the feature

* Corresponding author: maniar.amine@u-picardie.fr

engineering approaches used to build the models, as well as the prediction methods and forecasting strategies that are the focus of this study. Section V features the results and their analysis. Finally, a conclusion and future works prospects are provided.

2 Related works

Many different tools and methods are used for occupancy detection and prediction models, they are usually classified as deterministic, stochastic or machine learning models [7]. Deterministic models are simple and are based on long term monitoring to generate occupancy profiles for simulating software [8], but they can't reliably predict the randomness in occupants' activity. Stochastic methods use mathematical tools to model relations in historical data. They are used to generate realistic occupancy profiles based on probabilistic notions to better represent the uncertainties surrounding occupant behavior [9]. They often give good prediction results and can rank the features used as inputs by their importance for the prediction task, which makes them easily interpretable. Statistical models are however difficult to implement and rely on long studies to determine every relevant parameter. Machine learning techniques are used to model the distribution of conditional probabilities of an unknown variable Y given a known variable X. They can learn complex relationship between inputs and outputs and achieve a high prediction accuracy, but they require massive volumes of data and training them may require a high computing cost.

Table 1. Related works

Ref	Features	Method	Strategy
[7]	CO2, Noise, Time	Hidden Markov Model	Multiple steps
[8]	CO2, Noise, Outside temperature, Pressure	Long Short-Term Memory	Multiple steps
[9]	Heat Sensor	Convolutional Neural Networks	Multiple steps
[10]	CO2, Light, Temperature, Humidity	Convolutional Neural Networks (feature extraction) + Machine Learning (model prediction)	Multiple steps
[11]	Fan controls, Temperature, Humidity	Transformer	Single step

The papers listed in Table 1 all use different datasets and approach the problem through either time series prediction or occupancy state prediction, so comparing their performances is impossible as the results are specific to each case study. This paper aims to fill a gap in the literature pertaining to building occupancy prediction based on sensor data, which is how to choose the best model to predict time series data related to occupancy for different prediction horizons. Thus, this paper offers guidance for future research by focusing on the analysis of data structures and prediction strategies, rather than model hyperparameters.

3 Materials and Methods

3.1 Data Collection

In this work we use sensor data collected from two residential buildings in the Picardie region of France. CO2 concentration, noise, humidity, atmospheric pression, external temperature and humidity were collected in the living room of each building between 1 January 2017 and 31 December 2018 with a sampling frequency of 10 minutes. Table 2 describes this data.

Table 2. Summary of sensor data

	Mean	*Standard deviation*	*min*	*max*
CO2 (ppm)	620.37	149.09	294.73	2223.71
Noise (dB)	43.79	8.09	35.96	316.12
Temperature (°C)	13.17	7.86	-8.27	46.06
Humidity (%)	79.06	15.81	23.16	100.00

We can make a few preliminary observations from this summary: the CO2 signal is more variable than the other sensor signals and there is at least one outlier with a value over 2000 ppm, indicating either high activity or poor ventilation at a certain point in time. For these reasons, prediction models are hard to interpret and to generalize to different buildings or time frames.

3.2 Preprocessing and feature engineering

The raw sensor signals were resynced between all sensors to be labeled to the exact same timestamp and they were resampled to one-hour intervals and smoothed using an Epanechnikov kernel. Environmental data changes slowly overtime so we did not lose much information and we were able to reduce the number of computations needed to predict values further in the future. Then, to get a better insight on our data, we performed a seasonality study on our signals. A Fast Fourier Transform (FFT) was computed on the preprocessed signals with a sampling frequency of 1/year.

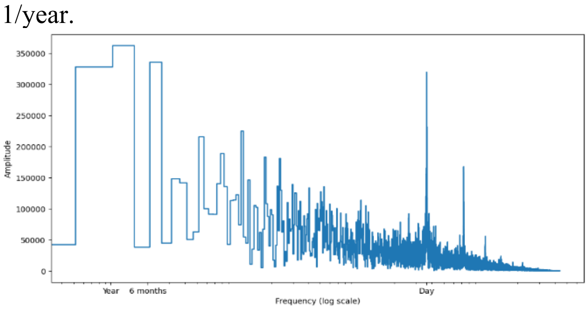

Fig. 1. FFT of CO2 signal

The CO2 signal has too many underlying frequencies to analyze, but we notice a few obvious frequencies corresponding to yearly, half-yearly and daily periodicities.

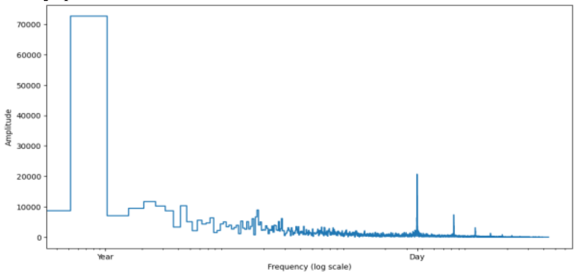

Fig. 2. FFT of Temperature signal

The temperature signal has a peak frequency corresponding to a yearly periodicity, as well as a non-negligible daily periodicity.

4 Methodology

Indirect sensor data like CO¬2 and Noise levels can be used to predict future occupancy states by labelling the data according to carefully chosen thresholds [12]. For example, increasing CO2 along with the detection of noise is indicative of a presence, constant CO2 and lack of noise indicate low activity, and decreasing CO2 can indicate an absence. Therefore, reliable occupancy detection algorithms can be developed using only indirect data [13]. Figure 3 shows the full process of occupancy prediction.

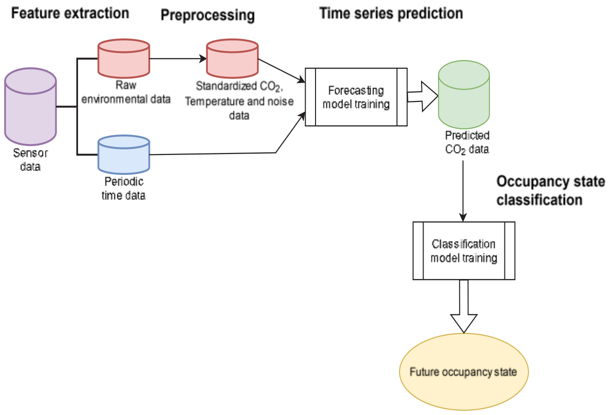

Fig. 3. Process of occupancy prediction

The scope of this paper concerns the prediction of sensor data which is necessary to apply classification algorithms on future data. Flexible models are needed to handle different types of time series for different prediction horizons to achieve good occupancy prediction at different scales [14]. For example, using granular data sampled at 10 minutes intervals, the model can learn to detect movement between rooms or the duration of presence or absence, but long-term prediction becomes much harder. With a 1 hour or higher sampling interval, daily routines like gatherings, sleeping or leaving for work can be modelled, at the cost of losing of finer details. In our study, we focus on modelling daily periodicities, so we used a 1 hour sampling interval.

4.1 Prediction models

Four different prediction models are considered in this paper. The persistence model, or baseline, returns the same value it was fed as an input without any modification. Sensor data does not have a high variability in short timeframes, so this method can still produce decent results for short term predictions [15]. The linear model applies a linear transformation between the input and the output, so the prediction depends only on the current state and no other parameters. The dense model consists of multiple layers of neurons that receive an input from every neuron in the previous layers and applies an activation function to the weighted sum of these inputs. This network is trained through backpropagation to minimize the loss between the labels and the final layer outputs. Finally, we chose a recurrent neural network (RNN) as the last model due to their ability to capture complex dependencies in the data and compare their performance with simpler models. Specifically, we chose the Long Short-Term Memory (LSTM) model, which processes data sequentially and manages an internal memory trained to keep only relevant information from the past to generate a prediction [16]. LSTM networks are distinct from other prediction models in that they were specifically designed to model sequential data and temporal dependencies, and the inherent memory management adds to their stability. Moreover, LSTMs can naturally handle sequences of varying lengths, making them flexible for different types of sequential data without needing significant modifications. Other neural networks like autoencoders or CNNs generally require fixed-size inputs and outputs. Transformers have been considered due to their flexibility and powerful feature extraction, however LSTMs were preferred for their advantages in terms of resource efficiency, simplicity and applicability to smaller datasets, where transformers often need large amounts of data to achieve good performance due to their large number of parameters, and their reliance on self-attention mechanisms requires processing entire sequences simultaneously which is extremely demanding in terms of memory and computational power [17].

4.2 Prediction strategy

Prediction strategy plays an important role in the quality of the results obtained by forecasting models. This encompasses various approaches and techniques used to make predictions, including the choice of algorithms, preprocessing methods, feature selection, and model tuning. Different prediction strategies can significantly impact the accuracy, reliability, and robustness of the forecasting results [18]. To understand the influence of prediction strategies, it is beneficial to compare different models applied to the same dataset but utilizing varied prediction strategies.

4.2.1 Single step prediction

This is the most popular strategy that predicts a single time step in the future. Depending on the model, the number of features and the input size can be different.

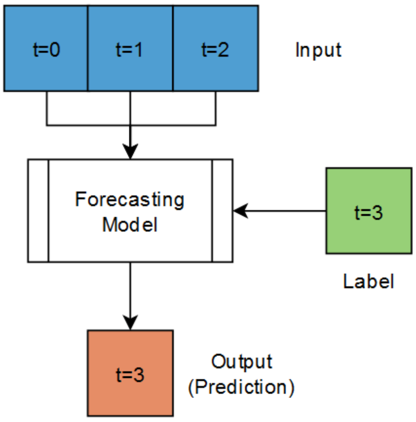

Fig. 4. Single-step prediction

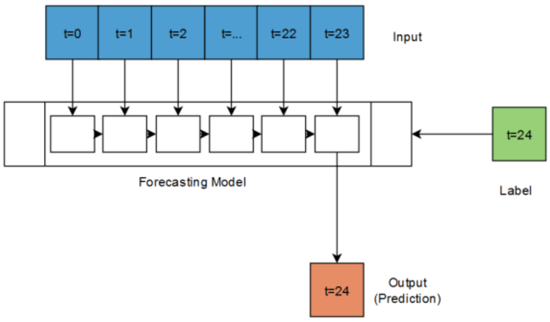

Fig. 5. LSTM Single-step prediction

The persistence model uses simply outputs the same value it was given as input. The linear model uses linear regression to find the coefficients that best describe the training data. The dense model can handle an arbitrary number of inputs and outputs, although defining optimal parameters require a lot of fine-tuning. Finally, the LSTM model takes every time step one by one sequentially while modifying its internal state at each step before producing a prediction based on the current state and the historical data represented by that internal state.

The final step to our preprocessing is normalization using Z-score, so the signals are more suitable to deep learning models that are sensible to high gradient values.

4.2.2 Multiple steps prediction

These models can be either direct or autoregressive. The direct models predict a sequence of futures values at once based on the history of data fed as an input.

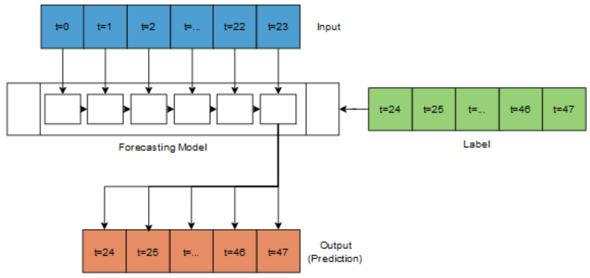

Fig. 6. Multiple-steps prediction

5 Results and Discussion

In this section we present the results obtained with the four forecasting methods, baseline (persistence), linear, dense and LSTM for each prediction strategy. Model training was performed using the Tensorflow Python library. For both the LSTM and dense model, we set a single layer of 64 neurons, followed by a single neuron output layer which learns to output the standardized prediction value. Multistep prediction is performed using a sliding window on real life data to produce the next prediction. MAE (Mean Absolute Error) is used for evaluation in both training and test sets, and the same data is used for every method for comparison purposes.

5.1 Single-step Prediction

The first experiment involves CO2 one-step predictions using sensor data as well as time variables as inputs. The data is sampled with a 1-hour frequency, with a 24-hour lookback window. MAE and RMSE (Root Mean Square Error) are measured by comparing the predictions 1 hour ahead of a 24-hour sequence with the test data. Tables 3 and 4 present the results of the comparison between the different models.

Table 3. Single step performances for CO2 Prediction

Method	MAE	RMSE (test)	Training Time (seconds)
Baseline	**0.1752**	**0.2535**	**2**
Linear	**0.0008**	**0.0009**	**40**
Dense	**0.0235**	**0.0157**	**47**
LSTM	**0.0106**	**0.0131**	**100**

Table 4. Single step performances for Temperature Prediction

Method	MAE (test)	RMSE (test)	Training time (seconds)
Baseline	**0.0464**	**0.0591**	**3**
Linear	**0.0474**	**0.0614**	**50**
Dense	**0.0529**	**0.0728**	**30**
LSTM	**0.0228**	**0.0376**	**100**

All the models except the baseline perform well for CO2 single step prediction in the test set, with the linear model being close to ground truth. This may be due to overfitting the neural networks, or to the bias-variance tradeoff induced by the high fluctuations and noise in the training data, which makes generalization harder for neural networks. Additionally, linear relationships often exist in environmental data like CO2 with a slow rate of change, which is advantageous to the linear model. However, the LSTM model achieved better results for temperature prediction when the linear and dense models were worse than the baseline, which might be because the LSTM model was able to better capture the time related features in the temperature signal since it contains less noise. The neural networks require however a much higher computational cost, so the choice of the model will depend on the accuracy needed, the available computing power as well as the particularities of the signal.

5.2 Multiple-step Prediction

The objective of the second experiment is to evaluate the performance of multi-step prediction models in forecasting future values over a 24-hour horizon. This experiment extends beyond the one-step prediction scenario, where models are trained to predict multiple future time steps given a window of past observations. The same training data with hourly samples and time-related variables are used, with the addition of lagged versions of the input variables to capture temporal dependencies. Different lookback windows and prediction horizons are tested, and performances are evaluated using MAE. The baseline model however is different in that it predicts the same values it has been fed, so its lookback window is always the same size as the prediction horizon. Table V shows the result of the experiment using lookback windows of 1 hour and 96 hours for prediction horizons of 12 hours, 24 hours, and table VI shows the results for the same lookback windows for 48 hours, and 96 hours prediction horizons. Multiple step performances for CO2 Prediction – 12 hours and 48 hours horizons. It is worth noting that training time for linear and dense models remained stable for different configurations, respectively around 35 seconds and 20 seconds which is due to how they treat inputs as independent of each other. Meaning even with changing the lookback window the models process all the input features simultaneously, making the time complexity less sensitive to changes in window size. The LSTM model however processes one time step at a time and must enroll the entire lookback sequence to backpropagate gradients, taking up to 3 minutes for longer inputs.

Table 5. Multiple step performances for CO2 Prediction – 12 hours and 48 hours horizons

	12h horizon		**24h horizon**	
Method	**MAE (1h input)**	**MAE (96h input)**	**MAE (1h input)**	**MAE (96h input)**
Baseline	0.7114	0.7114	0.2814	0.2814
Linear	0.2525	**0.2500**	0.2917	**0.2913**
Dense	0.2301	**0.2233**	0.2787	**0.2685**
LSTM	**0.2468**	0.2836	**0.2887**	0.3113

Table 6. Multiple step performances for CO2 Prediction – 48 hours and 96 hours horizons

	12h horizon		**24h horizon**	
Method	**MAE (1h input)**	**MAE (96h input)**	**MAE (1h input)**	**MAE (96h input)**
Baseline	0.3796	0.3796	0.4204	0.4204
Linear	0.3250	**0.3247**	0.3468	**0.3458**
Dense	0.3223	**0.3119**	0.3325	**0.3267**
LSTM	0.3269	**0.3078**	0.3498	**0.3357**

As expected, the linear and dense models have better performances when we use a longer lookback window in every case, however we notice that for LSTM, a bigger lookback window only results in better performances when the prediction horizon is longer, whereas short-term prediction have better accuracy with just a single step lookback. This means that the internal state of the LSTM after training is already sufficient to start predictions, and the lookback window becomes necessary when the dynamic of the signal starts changing in long term predictions.

We performed an additional experiment to specifically measure the effect of prediction horizon. Figure 7 shows the evolution of the MAE using the different models with a 1-hour lookback window and a gradually increasing prediction horizon.

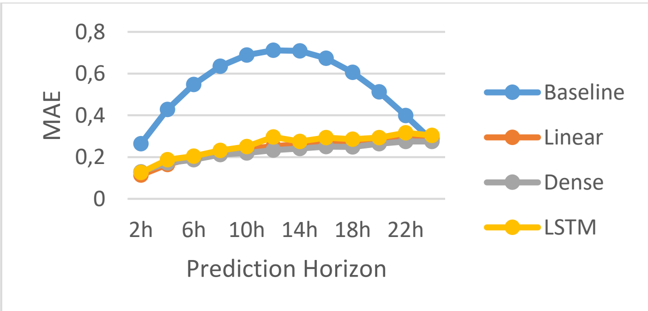

Fig 7. Multiple-steps prediction performance with a 24h loockback window

The baseline model shows an increasing error up to 12h then decreasing until 24h. This is due to the smoothing of the error over a 24h period, corresponding to the main frequency of the signal. This smoothing effect occurs because the baseline model relies on simple persistence, for which short term errors average out over a 24-hour period.

The more complex models perform are a lot less affected by the prediction horizon, showing only a slight increase in error with longer prediction horizons, meaning they are relatively robust to the output size, but they fail to capture short term fluctuations due to the lack of clear repetitive patterns and the reduced lookback window.

Our experimental results show that choosing the right forecasting strategy, i.e., fitting lookback windows and prediction horizons with a proper offset between the input and the output, is more important than the choice of the model itself and must be carefully considered before hyperparameters tuning.

6 Conclusion

In this paper, we proposed a method for predicting future occupancy states by using efficient occupancy detection algorithms on forecasted data. We investigated different strategies to predict the evolution of CO2 concentration in a residential building across different prediction horizons, enabling flexible granularity in classifying occupancy states. We compared the performances of four forecasting models (Persistence, Linear, Dense, LSTM) based on computational time and accuracy for one-step and multiple-steps prediction strategies. For short-term predictions, the persistence model provides at a low computational cost results that are close to the more complex algorithms. For mid-term or long-term predictions however, the LSTM achieve a higher accuracy thanks to their ability to handle long sequences of data and learn complex relationships in the data, while remaining simple enough for use in predictive command systems after their initial training. The error for multiple-steps prediction remains relatively high around 25% to 32%, mainly due to the non-linearity of the signals and their changing dynamics over time. Future works will address this problem by training autoregressive models guided by periodic data. This approach will allow for greater flexibility and the gradual update of the model to the changing dynamics of the signal.

References

[1] J. P. Tricoire, « Why Buildings Are the Foundation of an Energy-Efficient Future », in *World Economic Forum*, 2021.
[2] L. Belussi *et al.*, « A review of performance of zero energy buildings and energy efficiency solutions », *J Build Eng*, vol. 25, nº 100772, 2019.
[3] A. Bagnasco, S. Massucco, M. Saviozzi, F. Silvestro, et A. Vinci, « Design and validation of a detailed building thermal model considering occupancy and temperature sensors », in *2018 IEEE 4th International Forum on Research and Technology for Society and Industry*, RTSI: IEEE, 2018, p. 1-6.
[4] A. C. Menezes, A. Cripps, D. Bouchlaghem, et R. Buswell, « Predicted vs. actual energy performance of non-domestic buildings: Using post-occupancy evaluation data to reduce the performance gap », *Applied energy*, vol. 97, p. 355-364, 2012.
[5] B. Dong, Z. Li, et G. Mcfadden, « An investigation on energy-related occupancy behavior for low-income residential buildings », *Science and Technology for the Built Environment*, vol. 21, nº 6, p. 892-901, 2015, doi: 10.1080/23744731.2015.1040321.
[6] S. Hadri, M. Najib, M. Bakhouya, Y. Fakhri, et M. El Arroussi, « Performance Evaluation of Forecasting Strategies for Electricity Consumption in Buildings », *Energies*, vol. 14, nº 18, p. 5831, 2021.
[7] O. A. Kabbaj, L.-M. Péan, J.-B. Masson, B. Marhic, et L. Delahoche, « Occupancy states forecasting with a hidden Markov model for incomplete data, exploiting daily periodicity », *Energy and Buildings*, vol. 287, p. 112985, mai 2023, doi: 10.1016/j.enbuild.2023.112985.
[8] S. Mahjoub, S. Labdai, L. Chrifi-Alaoui, B. Marhic, et L. Delahoche, « Short-Term Occupancy Forecasting for a Smart Home Using Optimized Weight Updates Based on GA and PSO Algorithms for an LSTM Network », *Energies*, vol. 16, nº 4, Art. nº 4, janv. 2023, doi: 10.3390/en16041641.
[9] S. Arvidsson, M. Gullstrand, B. Sirmacek, et M. Riveiro, « Sensor fusion and convolutional neural networks for indoor occupancy prediction using multiple low-cost low-resolution heat sensor data », *Sensors*, vol. 21, nº 4, p. 1036, 2021.
[10] A. N. Sayed, Y. Himeur, et F. Bensaali, « From time-series to 2d images for building occupancy prediction using deep transfer learning », *Engineering Applications of Artificial Intelligence*, vol. 119, p. 105786, 2023.
[11] K. Sun, I. Qaisar, M. A. Khan, T. Xing, et Q. Zhao, « Building occupancy number prediction: A Transformer approach », *Building and Environment*, vol. 244, p. 110807, oct. 2023, doi: 10.1016/j.buildenv.2023.110807.
[12] M. K. Diarra, A. Maniar, J.-B. Masson, B. Marhic, et L. Delahoche, « Occupancy State Prediction by Recurrent Neural Network (LSTM): Multi-Room Context », *Sensors*, vol. 23, nº 23, Art. nº 23, janv. 2023, doi: 10.3390/s23239603.
[13] D. Calì, P. Matthes, K. Huchtemann, R. Streblow, et D. Müller, « CO2 based occupancy detection algorithm: Experimental analysis and validation for office and residential buildings », *Building and Environment*, vol. 86, p. 39-49, avr. 2015, doi: 10.1016/j.buildenv.2014.12.011.
[14] Z. Yang et B. Becerik-Gerber, « Modeling personalized occupancy profiles for representing long term patterns by using ambient context », *Building and Environment*, vol. 78, p. 23-35, août 2014, doi: 10.1016/j.buildenv.2014.04.003.
[15] T. Ahmad et H. Chen, « Deep learning for multi-scale smart energy forecasting », *Energy*, vol. 175, p. 98-112, mai 2019, doi: 10.1016/j.energy.2019.03.080.
[16] S. Hochreiter et J. Schmidhuber, « Long short-term memory », *Neural computation*, vol. 9, nº 8, p. 1735-1780, 1997.
[17] T. B. Brown *et al.*, « Language Models are Few-Shot Learners », 22 juillet 2020, *arXiv*: arXiv:2005.14165. doi: 10.48550/arXiv.2005.14165.
[18] S. Makridakis, E. Spiliotis, et V. Assimakopoulos, « Statistical and Machine Learning forecasting methods: Concerns and ways forward », *PLOS ONE*, vol. 13, nº 3, p. e0194889, mars 2018, doi: 10.1371/journal.pone.0194889.