

## Article

# Indoor Microclimate Monitoring and Forecasting: Public Sector Building Use Case

Ruslans Sudniks <sup>1</sup>, Arturs Ziemelis <sup>1</sup>, Agris Nikitenko <sup>1</sup>, Vasco N. G. J. Soares <sup>2,3,4,\*</sup> and Andis Supe <sup>1</sup>

<sup>1</sup> Faculty of Computer Science, Information Technology, and Energy, Riga Technical University, Azenes st. 12, LV-1048 Riga, Latvia; ruslans.sudniks@rtu.lv (R.S.); arturs.ziemelis@rtu.lv (A.Z.); agris.nikitenko@rtu.lv (A.N.); andis.supe@rtu.lv (A.S.)

<sup>2</sup> Polytechnic University of Castelo Branco, Av. Pedro Álvares Cabral n° 12, 6000-084 Castelo Branco, Portugal

<sup>3</sup> Instituto de Telecomunicações, Rua Marquês d'Ávila e Bolama, 6201-001 Covilhã, Portugal

<sup>4</sup> AMA—Agência Para a Modernização Administrativa, Rua de Santa Marta n° 55, 1150-294 Lisbon, Portugal

\* Correspondence: vasco.g.soares@ipcb.pt

**Abstract:** This research aims to demonstrate a machine learning (ML) algorithm-based indoor air quality (IAQ) monitoring and forecasting system for a public sector building use case. Such a system has the potential to automate existing heating/ventilation systems, therefore reducing energy consumption. One of Riga Technical University's campus buildings, equipped with around 128 IAQ sensors, is used as a test bed to create a digital shadow including a comparison of five ML-based data prediction tools. We compare the IAQ data prediction loss using Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) error metrics based on real sensor data. Gated Recurrent Unit (GRU) and Kolmogorov–Arnold Networks (KAN) prove to be the most accurate models regarding the prediction error. Also, GRU proved to be the most efficient model regarding the required computation time.

**Keywords:** indoor air quality; sensor network; Internet of Things; digital shadow; data forecasting; machine learning algorithms



Academic Editors: Arkaitz Zubiaga and Xinming Zhang

Received: 2 December 2024

Revised: 28 January 2025

Accepted: 31 January 2025

Published: 8 February 2025

**Citation:** Sudniks, R.; Ziemelis, A.; Nikitenko, A.; Soares, V.N.G.J.; Supe, A. Indoor Microclimate Monitoring and Forecasting: Public Sector Building Use Case. *Information* **2025**, *16*, 121. <https://doi.org/10.3390/info16020121>

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Balancing between lowering a building's energy consumption and maintaining optimal indoor air quality (IAQ) is one of the main challenges today toward not affecting occupants' health, cognitive abilities, and well-being. For example, the impact of IAQ on students has been studied, finding that poor IAQ has been linked to a range of negative effects, including cognitive decline, reduced focus, and decreased attendance [1]. Managing IAQ has become even more crucial as people tend to spend most of their time indoors [2]. Thus, a smart Heating, Ventilation, and Air Conditioning (HVAC) system capable of performing automated system monitoring and control is one of the stepping stones to this challenge. The increasing availability of technologies, such as affordable sensors, wireless data transmission, and cloud computing services has made it easier to collect IAQ data, thus enabling deeper system analyses, while managing and processing large amounts of sensor network data involves the use of machine learning (ML) models to gain valuable insights [3,4].

The unique aspect of the building as a control object is that it is a complex socio-technological system. The concept of a digital twin has emerged as a today's solution to manage the complexity of modern buildings where the system is influenced by both internal and external factors [5–7]. Digital twin can be a virtual replica of a building making

predictive adjustments, ensuring sustainability and helping building managers to monitor real-time conditions. ML plays a crucial role in this, as it can analyse historical data to uncover patterns and predict future conditions, allowing for proactive and intelligent building management [8].

The research presented in this paper aims to develop public sector buildings IAQ monitoring and analysis tools that corresponds to digital shadow [8] definition (automated data acquisition, sorting, analysis, prediction). We compare five ML-based data forecast models (Prophet, Transformer, Kolmogorov–Arnold Networks (KAN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU)) for IAQ parameters prediction with the potential use for control of HVAC systems in public sector buildings. The selected models represent different types of algorithms regarding their complexity, speed, and accuracy [9]. One of Riga Technical University’s faculty buildings, equipped with a network of IAQ sensors, was selected as an experimental environment. This building has office rooms, laboratories and lecture auditoriums where each group has different typical use.

The novelty of this research is the introduction of the KAN model and an additional integrated hyperparameter optimization system using Optuna. A comprehensive methodological approach has been implemented to systematically fine-tune and improve IAQ time series models. Our study examines a different type of rooms that have different workloads and usage patterns in public sector buildings. The application of machine learning-based adaptive control algorithms for HVAC systems holds significant potential for improving building energy efficiency.

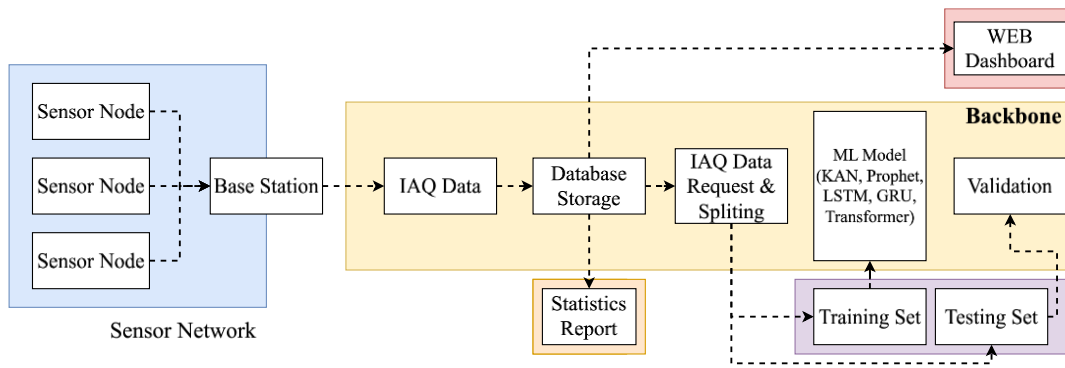
## 2. Sensor Network

### 2.1. Description of the Existing IAQ Sensor Network

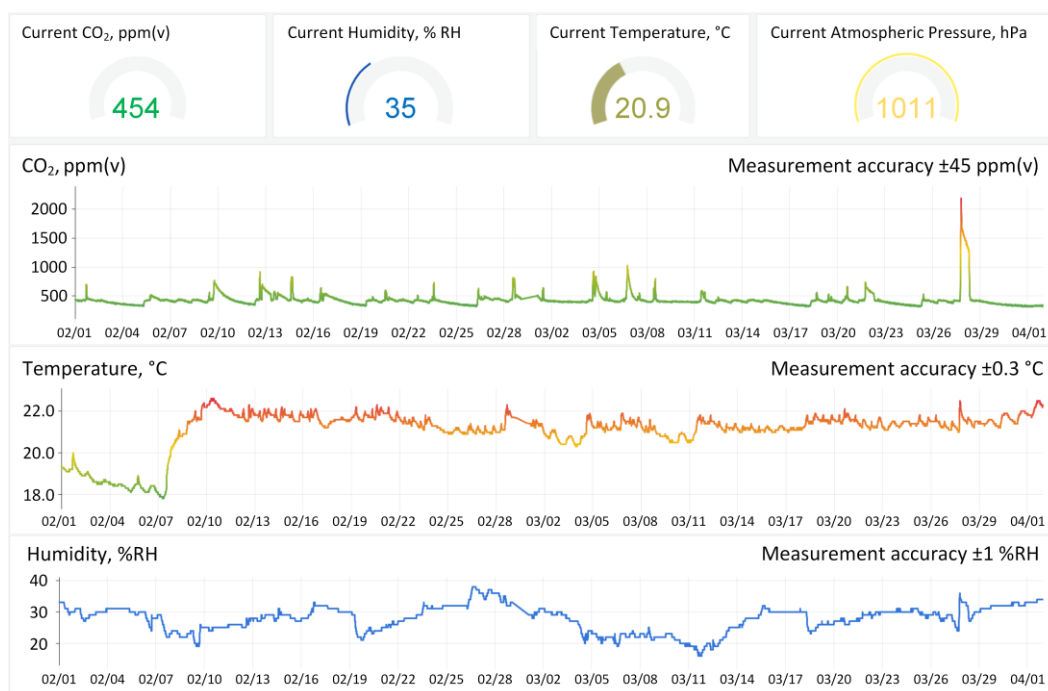
For this specific use case, we chose one of the faculty buildings on the Riga Technical University campus which is equipped with Aranet4 sensor nodes [10]. Each sensor node measures four IAQ parameters: temperature ( $^{\circ}\text{C}$ ), relative humidity (% RH),  $\text{CO}_2$  (ppm by volume), and atmospheric pressure (hPa) every 10 min.  $\text{CO}_2$  concentration measurement: range 0–9999 ppm(v), resolution 1 ppm(v), accuracy  $\pm(30 \text{ ppm(v)} + 3\% \text{ of reading})$ . Temperature measurement: range 0–50  $^{\circ}\text{C}$ , resolution 0.1  $^{\circ}\text{C}$ , accuracy  $\pm 0.3 \text{ }^{\circ}\text{C}$ . Relative humidity measurement: range 0–85%, resolution 1%, accuracy  $\pm 3\%$  [11]. Data are transferred between sensors and base stations via the proprietary “Aranet Radio” protocol, which uses LoRa modulation and runs in the 868/920 MHz bands [12]. Aranet enables complete control over the IoT sensor network, including base station, cloud, and sensor management.

Each room has one IAQ sensor device near the entrance door, and there are three sensor nodes in the corridors on each of the six floors. The Aranet4 sensor devices perform automated  $\text{CO}_2$  measurement self-calibration every 30 days of operational period [4].

Figure 1 shows block diagram of our digital shadow model for public sector building. The IAQ sensor network is the source of data containing 128 sensor nodes, and each one sends the four values to the base station. IAQ data are then retrieved and stored in a database [13] through regular requests every 10 min. At this stage it is possible to represent real time and/or historical datasets through a Grafana-based dashboard that can be accessed publicly (see Figure 2). Or generate statistics reports of building’s indoor climate performance (examples given in Figures 3 and 4). The final stage is data processing. We propose the use of an ML model to perform IAQ data prediction. Accordingly, patterns hidden in historical data potentially could reduce the building’s HVAC systems energy consumption. In this study, we consider five ML models (see Section 3). Historical data are used to train and test ML models. The validation stage is the comparison of prediction accuracy between different models.



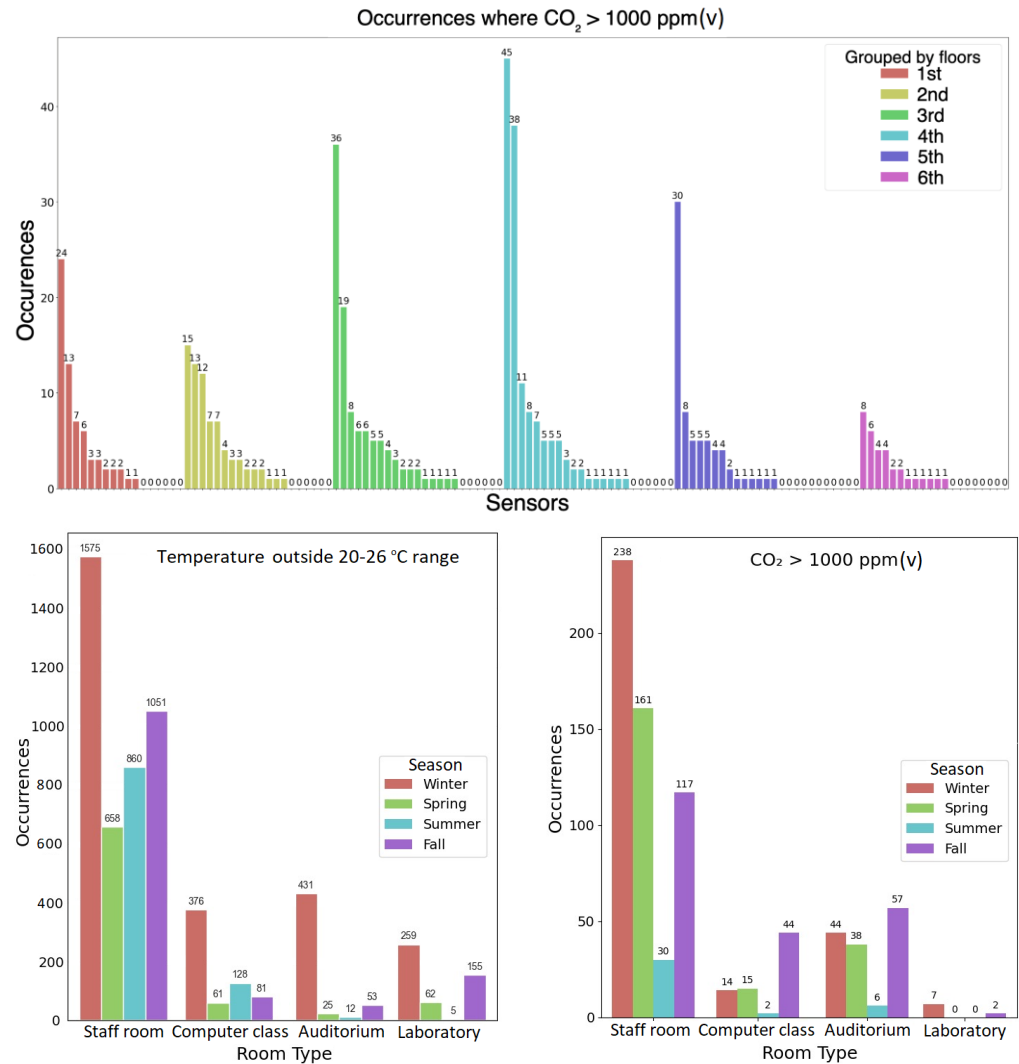
**Figure 1.** Faculty building’s digital shadow block diagram. Used acronyms: indoor air quality (IAQ), machine learning (ML), Kolmogorov–Arnold Networks (KAN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU).



**Figure 2.** IAQ sensor data dashboard (data from one of the sensors).

Figure 2 shows the Grafana-based sensors’ dashboard for the visual representation of historical data. The upper part shows a current reading of four IAQ metrics and below are three graphs of selected features. Colour coding in CO<sub>2</sub> and temperature graphs are used to highlight maximums.

One of our developed building’s digital shadow model’s features is to create IAQ statistics reports. For example, Figure 3 shows histograms of temperature and CO<sub>2</sub> data gathered from the faculty building. Here, we present only those cases when the temperature exceeds the 20–26 °C range and CO<sub>2</sub> > 1000 ppm. Based on various guidelines, it is stated that the optimal range for healthy working conditions is 400 ppm to 1000 ppm [2]. The Histogram on the top (see Figure 3) represents data over a two-month period from all 128 sensors grouped by floors. This graph highlights some extreme cases (rooms) where CO<sub>2</sub> regularly exceeds the optimal threshold. These particular cases highlight the problem addressed in this research paper. To improve the situation, a thorough inspection/maintenance or possible automation of the HVAC system serving this space should be a priority.



**Figure 3.** Histograms of faculty building’s temperature and CO<sub>2</sub> sensor data. The top histogram shows the CO<sub>2</sub> concentration above 1000 ppm over a two-month period grouped by the building’s floors. The two histograms at the bottom represent the seasonal effect on indoor temperature and CO<sub>2</sub> level. Here, sensors are grouped by types of rooms.

Two histograms at the bottom of Figure 3 represent the seasonal effect on temperature and CO<sub>2</sub> data. In these graphs, sensors are grouped by types of rooms: staff (office) rooms, computer classes, auditoriums, and laboratories. From this perspective, we see seasonality impact on different types of rooms. The maximum occupancy is during the autumn, winter, and spring months, being typical for academic buildings. For our study, we selected a two-month period overlapping with transition period from winter to spring (see Section 2.2). The selected time period overlaps with season of increased deviations of temperature and indoor CO<sub>2</sub>. Also, practical aspects such availability of uninterrupted data (sensor failure, network maintenance), as well as computer memory resources required for ML model were considered.

For ML-based temperature, humidity, and CO<sub>2</sub> level prediction models, historical IAQ data are split into training and testing sets. The training set is used for the model training,

which is then validated on the testing set by calculating Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) [14,15], as defined in the Equations (1) and (2):

$$MAE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} |y_i - \hat{y}_i| \quad (1)$$

where the following are defined:

$n_{samples}$ —data point count;

$y_i$ —forecasted value;

$\hat{y}_i$ —true value.

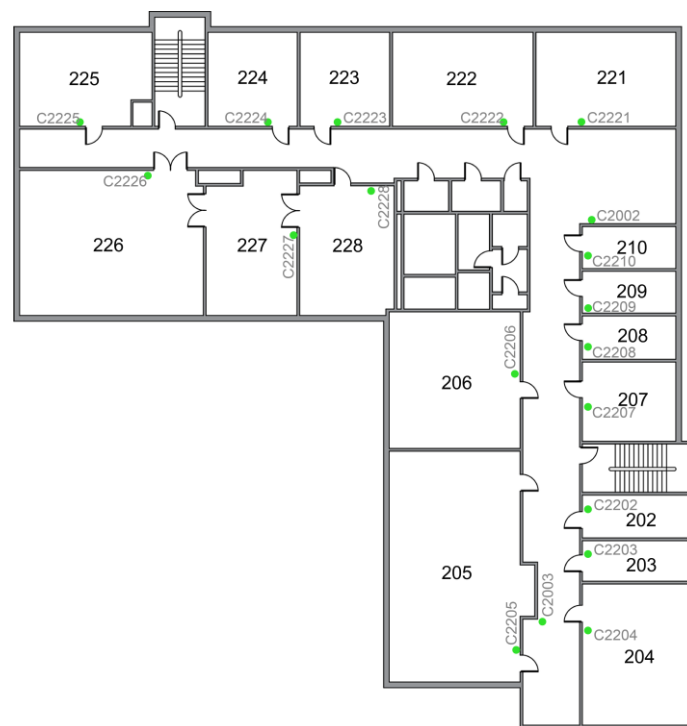
$$RMSE(y, \hat{y}) = \sqrt{\frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} (y_i - \hat{y}_i)^2} \quad (2)$$

where the following are defined:

$n_{samples}$ —data point count;

$y_i$ —forecasted value;

$\hat{y}_i$ —true value.



**Figure 4.** The 2nd floor plan of the faculty building shows sensor locations (green dots).

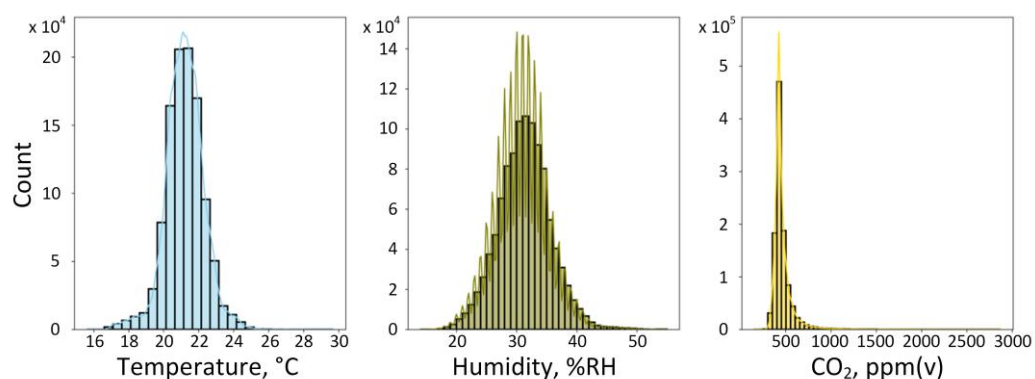
## 2.2. Analysis of Indoor Air Quality Sensor Datasets

In this Subsection, we find properties of the dataset of a 2-month period (February/March 2024) obtained from the building's IAQ sensor network nodes located in office rooms, auditoriums, laboratories, and hallways. Figure 4 shows one of the building's floorplans (all of the floors have approximately the same layout). The green dots indicate sensor node placements.

The selected 2-month time interval overlaps with the transitional period of the centralised heating, allowing for data collection under changing temperatures and yearly seasonal conditions. The outdoor temperature during the selected time interval changes in

the range from  $-7.3$  to  $+21.4$  °C [16]. In our previous work [17], it was found that indoor temperature and humidity have a stronger patterning effect, while CO<sub>2</sub> is more stochastic. The number of people indoors changes the CO<sub>2</sub> concentration level more rapidly compared to temperature and humidity, respectively. Meanwhile, the other two parameters change according to hourly and day-of-week rates. Calculated correlations between these features are weak, except for a moderate positive correlation between temperature and CO<sub>2</sub> and a slight negative correlation between humidity and CO<sub>2</sub> [17].

The histograms in Figure 5 indicate that the temperature data are normally distributed, with most values clustered around 20–22 °C, showing stable indoor conditions. Humidity is slightly skewed to the right, with readings mostly around 30–35%, and some higher levels. CO<sub>2</sub> levels are heavily skewed to the right, with a peak around 450–500 ppm and a long tail towards higher values, but a few readings exceed 1000 ppm. This skewness in CO<sub>2</sub> suggests occasional spikes, due to specific events. Overall, humidity and CO<sub>2</sub> levels are well-controlled, with minimal fluctuations. CO<sub>2</sub> levels fluctuate more than temperature and humidity, with some concerning peaks.



**Figure 5.** Histograms of temperature, humidity, and CO<sub>2</sub> sensor readings with distribution mean value curve.

### 3. Data Forecasting Models

This section covers a short review of five selected data forecast models: KAN, Prophet, LSTM, GRU, Transformer. The selection is based on assumptions of computational complexity and model suitability for IAQ forecasting.

**KANs** leverage the Kolmogorov–Arnold representation theorem to efficiently model high-dimensional data with complex dependencies [18]. As promising alternatives to Multi-Layer Perceptrons (MLPs) [19], KANs also have a strong mathematical foundation. While MLPs are based on the universal approximation theorem, KANs rely on the Kolmogorov–Arnold representation theorem. The key difference is that KANs apply activation functions on edges, whereas MLPs apply them on nodes. This structured approach in KANs enhances both the network’s generalization capabilities and interpretability, making them ideal for the precise modelling of complex systems, such as in finance and engineering.

**Prophet** is a forecasting method for time series data that uses an additive model to account for non-linear trends, incorporating yearly, weekly, and daily seasonality, along with holiday effects. It performs optimally with time series that exhibit strong seasonal patterns and have multiple seasons of historical data. Prophet is also resilient to missing data, and trend changes, making it effective in managing outliers [20].

**LSTM**, first introduced by Sepp Hochreiter and Jürgen Schmidhuber in 1997, is an advanced type of recurrent neural network (RNN). Unlike standard RNNs, the LSTM includes an input gate and a forget gate. These gates help the network manage both long-term and short-term memory at different time steps [21].

**GRU** is an improved version of LSTM with a faster training process. It is simpler than LSTM with less computational complexity. GRU consists of gates that are collectively involved in balancing the interior flow of units' information. Input gate and forget gate are combined and form a new gating unit, typically called as update gate. The update gate focuses on balancing the state between the previous activation and the candidate activation [22].

**Transformer** is a deep learning architecture designed for use in tasks such as reinforcement learning, large-scale processing of natural language and other sequential data. It is extremely parallelizable and effective at managing long-range dependencies since it uses self-attention mechanisms rather than recurrent layers (as in RNNs or LSTMs). Positional encoding, which preserves order information in sequences, multi-head attention for capturing various aspects of the data, and an encoder–decoder structure are the essential elements. Applications such as text production, translation, and time series analysis make extensive use of it [23].

The chosen models offer a variety of features regarding IAQ forecasting. Compared to linear models, KAN offers better accuracy and interpretability, making it ideal for capturing intricate non-linear interactions. Prophet is useful for recording daily and seasonal variations because it effectively manages seasonal trends in IAQ data. For sequential applications involving temporal dependencies, GRU offers a simpler and more computationally efficient architecture than LSTM and has a faster training time. Transformer is particularly well suited for complex prediction problems due to its self-attention mechanism, which allows it to capture complex interactions. In Section 4, we show the results from all five time series data forecast model accuracy tests.

## 4. Data Prediction Results

### 4.1. Experiment Description

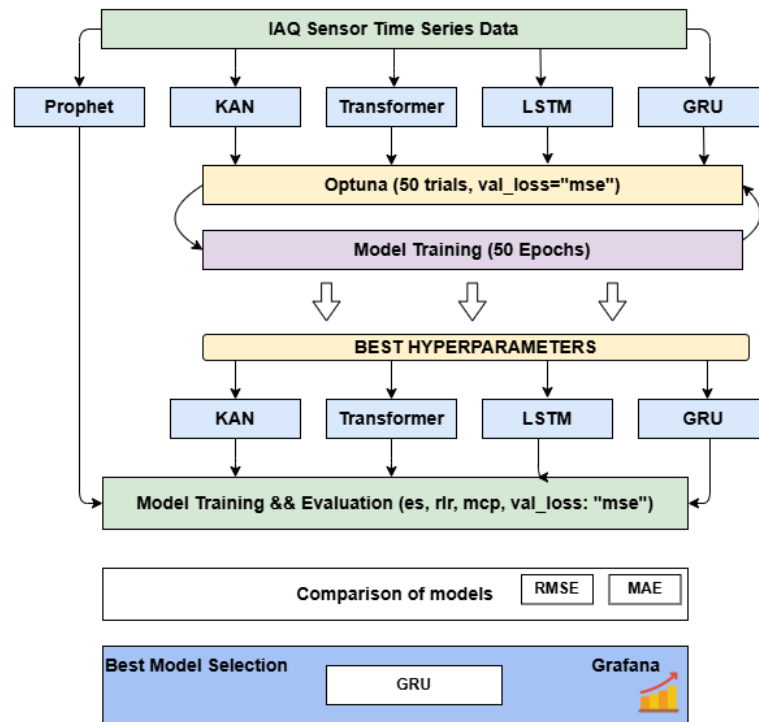
The models discussed in Section 3 adopt various approaches to predict IAQ characteristics. Since the Prophet model is restricted to univariate time series forecasting and KAN, being an expert model, has notable architectural limitations, it is challenging to evaluate these models consistently with other options like GRU, LSTM, and Transformer.

For this study, we use a dataset collected from 128 sensors, distributed across different floors and locations in the academic building. Sensors are located next to each room's entrance doors. The selected dataset covers a time window from a two-month period (February/March 2024). To perform a model comparison, we treat each sensor and data type as a single output and 20% of input dataset is used for testing model performance. The dataset is fed as the input to all five prediction models. Data forecasting is performed for three features of each sensor separately: temperature, CO<sub>2</sub>, and relative humidity.

The training process of the selected algorithms is based on the following steps. Initially, an empirical method is used to choose the models' hyperparameters, tweaking each one separately to create an adequate starting point. To identify the ideal hyperparameters, including the number of neurons, batch size, and number of layers, additional optimization is subsequently carried out using Optuna [24] on a single sensor as a case study. Figure 6 illustrates the workflow for model training, hyperparameter optimization with Optuna, and evaluation using validation loss. The hyperparameter optimization results are provided in Table 1.

The training procedure is improved by using the Adam optimizer. Several callbacks, including EarlyStopping, ReduceLROnPlateau, and ModelCheckpoint, are employed to improve the training procedure. EarlyStopping prevents overfitting by tracking validation loss and stopping training when no improvement is seen. When validation loss plateaued, ReduceLROnPlateau changes the learning rate, enabling the model to perform

more precise modifications in smaller increments. To guarantee that the final model is the best-performing one, ModelCheckpoint stores the best version of the model based on validation loss. Compared to other models that require extensive hyperparameter adjustment, Prophet is the most straightforward and does not require any special tuning because it is designed to offer optimum performance with default settings. Therefore, it can be classified as a lightweight forecast model (regarding computation resources consumed). GRU also belongs to the class of lightweight models.



**Figure 6.** Hyperparameter tuning and evaluation of IAQ time series models. Used acronyms: EarlyStopping (es), ReduceLROnPlateau (rlr), ModelCheckpoint (mcp), Mean Square Error (mse), validation loss (val\_loss).

**Table 1.** Hyperparameter optimization results for GRU, LSTM, Transformer, and KAN models.

Parameter	GRU	LSTM	Transformer	KAN
Optuna	50 trials	50 trials	50 trials	50 trials
epochs/steps	50 epochs	50 epochs	50 epochs	20 steps
num_units/heads	256 neurons	192 neurons	2 heads	hidden dim = 12
num_layers	1	1	8	N/A
ff_dim	N/A	N/A	96	grid = 7
dropout_rate	0.1	0.2	0.1	N/A
recurrent_dropout	0.2	0	N/A	N/A
learning_rate	~0.0008965	~0.000128	~0.00059	~0.003544
batch_size	32	32	96	N/A
optimizer	adam	adam	adam	LBFGS
bidirectional	True	True	N/A	N/A
additional params	N/A	N/A	N/A	k = 3, lamb = 0.000622

Table 1 summarizes the GRU, LSTM, Transformer, and KAN models’ hyperparameter optimization results. Both the GRU and LSTM models favour sequential memory processing, support bidirectional layers, and recurrent dropout by controlling the number of

neurons per layer using num\_units. Even so, the Transformer design is more appropriate for capturing global dependencies since it emphasizes Multi-Head Attention (num\_heads) and Feedforward Dimensionality (ff\_dim), and it facilitates parallel processing through feedforward layers and attention techniques. KAN distinguishes itself by defining its distinct computational framework with grid size, steps, and hidden\_dim, which is comparable to neurons. GRU, LSTM, and Transformer all use comparable batch sizes and dropout strategies, but KAN is distinct in that it is not dependent on these variables.

The best and worst prediction errors as min/max RMSE and MAE values from all five models are summarized in Table 2. We present the best (min error) and worst (max error) prediction cases (one sensor/feature out of 128 sensors) for each model as well as the average error for the whole dataset (averaged value out of 128 sensors).

**Table 2.** Performance comparison of prediction models for CO<sub>2</sub> and temperature datasets.

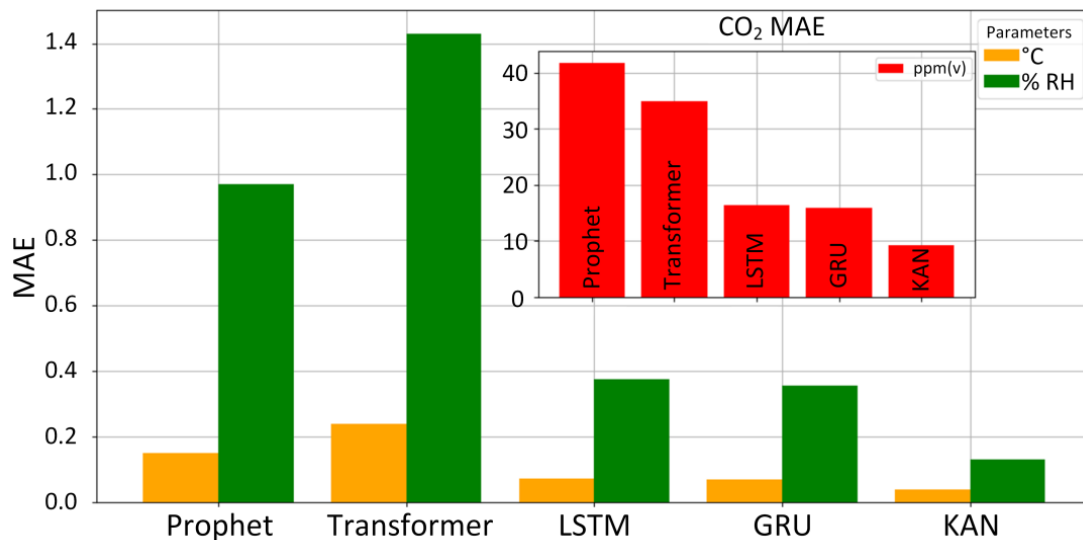
Model	Feature	RMSE			MAE			Computation Time
		Min	Max	Mean	Min	Max	Mean	
Prophet	Temp	0.07	0.45	0.21	0.06	0.34	0.15	18 min
	CO <sub>2</sub>	10.39	287.03	64.13	7.95	219.47	41.62	
	RH	0.76	2.65	1.34	0.56	2.12	1.02	
LSTM	Temp	0.05	0.34	0.11	0.03	0.19	0.07	17 min
	CO <sub>2</sub>	8.35	226.46	30.04	6.34	78.96	16.36	
	RH	0.37	1.83	0.62	0.17	0.83	0.38	
Transformer	Temp	0.06	0.87	0.31	0.04	0.74	0.24	51 min
	CO <sub>2</sub>	11.76	202.89	48.15	9.35	129.45	34.87	
	RH	0.49	5.21	1.79	0.33	4.55	1.43	
KAN	Temp	0.03	0.61	0.07	0.02	0.19	0.04	506 min
	CO <sub>2</sub>	6.39	37.37	15.00	4.88	17.27	9.20	
	RH	0.19	0.89	0.27	0.08	0.37	0.13	
GRU	Temp	0.05	0.34	0.11	0.03	0.19	0.07	14 min
	CO <sub>2</sub>	8.61	223.07	29.74	6.36	77.75	15.96	
	RH	0.36	1.75	0.61	0.15	0.74	0.36	

Computation time (the last column in Table 2) is measured using the same machine for all five models. This involves creating a virtual environment on Python 3.10 for each model. The key parameters of the computer hardware used in the tests are as follows: Intel(R) Core(TM) i9-13900KF Central processing unit, NVIDIA GeForce RTX 4070 Ti Graphics processing unit, Kingston 64GB 6000MT/s DDR5 CL30 DIMM Kit of 2, Samsung 980 PRO 1TB solid state drive disc.

In general, CO<sub>2</sub> predictions tend to be less accurate than temperature and RH, which are more predictable (stronger workday/seasonal patterns). Meanwhile, indoor CO<sub>2</sub> level depends directly on the number of visitors in each room. According to RMSE and MAE, the Prophet model has a higher CO<sub>2</sub> level prediction error (RMSE: 287.03, MAE: 219.47) because it cannot detect those emissions spikes. This makes it less effective for complex, fluctuating data such as CO<sub>2</sub>. In contrast, GRU and LSTM do this task better than Prophet. Since GRU has a more lightweight structure, with only two gates, it processes data faster, particularly with shorter datasets, where there are no hidden interconnections. LSTM also performs better than Prophet (RMSE: 226.46), although it requires more time and/or computation resources because it is more complex than GRU and Prophet.

KAN delivers the highest CO<sub>2</sub> prediction accuracy (RMSE: 37.37), but this comes at a considerable cost in computation time, making it less practical for time-sensitive or large-scale predictions.

Figure 7 shows averaged values (mean value of 128 sensor prediction errors). The overall outcome is that GRU is the fastest and the most efficient model, making it a solid option for aggregating small datasets with absence of hidden factors. KAN is the most accurate model, but its long runtime means it is only useful if you accept long delays or use larger computation resources for highly accurate prediction results. It should be noted that KAN is quite a new model that is currently being improved.



**Figure 7.** Comparison of average MAE metric for temperature, CO<sub>2</sub>, and RH.

#### 4.2. Data Forecast Using Clusterization Approach

According to the analysis of IAQ datasets in Section 2.2, we perform the clusterization of the temperature, CO<sub>2</sub>, and RH sensor data. Clusters are sets of sensors that exhibit similar behaviour in temperature, humidity and CO<sub>2</sub> statistics, facilitating the identification of different environmental patterns for targeted control. The Elbow Method and Silhouette Score estimation is used in the clustering process to optimize the number of clusters for balanced separation. Here, the whole building is considered as one dataset which is divided into nine clusters using K-means clustering [25]. The number of clusters is determined by the highest silhouette score, resulting in meaningful and interpretable groups. Every cluster organizes data points that share comparable environmental circumstances, as shown in see Figure 8a.

Our examination uncovers different clustering patterns for temperature and CO<sub>2</sub> levels, corresponding to sensors placements throughout the building. Out of nine clusters, we select three largest clusters from temperature feature (further indicated as cluster 0 to cluster 2). Accordingly, temperature is categorized into three mean temperature groups: low (17 °C to 19 °C), medium (19 °C to 22 °C), and high (above 22 °C). CO<sub>2</sub> mean levels are divided into two groups: below 500 ppm indicating good ventilation, and up to 800 ppm highlighting crowded areas or poorly ventilated rooms. The temperature clustering, shown in Figure 8b, gives a better understanding of sensors placement in areas like corridors and auditoria, making it the preferred measurement for additional spatial examination.

To improve the accuracy of our predictions, we divide the data into temperature clusters and use LSTM networks to forecast the ambient conditions both inside each cluster and over the whole dataset. Additionally, we test whether clustering yields improved results

by applying the same approach using GRU and Transformer models. The performance of the model is assessed by utilizing mean RMSE and MAE values (see Table 3).



**Figure 8.** Pairplot of sensor clusters ( $k = 9$ ) on the left side (a). Result of all temperature sensor data clusterization on the right side (b).

**Table 3.** Clusterization impact analysis.

Model	Cluster	Temperature		CO <sub>2</sub>		Humidity		Model Size, kBytes
		RMSE	MAE	RMSE	MAE	RMSE	MAE	
LSTM	Entire Dataset	0.29	0.24	46.30	26.97	1.04	0.77	1164
	cluster 0	0.16	0.12	35.80	25.40	0.70	0.53	915
	cluster 1	0.19	0.15	43.27	22.83	1.03	0.77	960
	cluster 2	0.26	0.20	40.50	23.53	1.11	0.82	849
Transformer	Entire Dataset	0.31	0.24	47.52	33.2	1.12	0.83	7025
	cluster 0	0.38	0.34	51.03	32.63	1.56	1.29	1151
	cluster 1	0.22	0.15	45.36	26.72	1.14	0.88	1830
	cluster 2	0.56	0.45	57.35	31.087	2.04	1.53	1260
GRU	Entire Dataset	0.16	0.12	28.51	17.97	0.75	0.57	890
	cluster 0	0.16	0.11	43.28	28.07	0.65	0.41	703
	cluster 1	0.13	0.09	32.91	12.60	0.71	0.46	737
	cluster 2	0.19	0.14	32.92	14.77	0.86	0.63	653

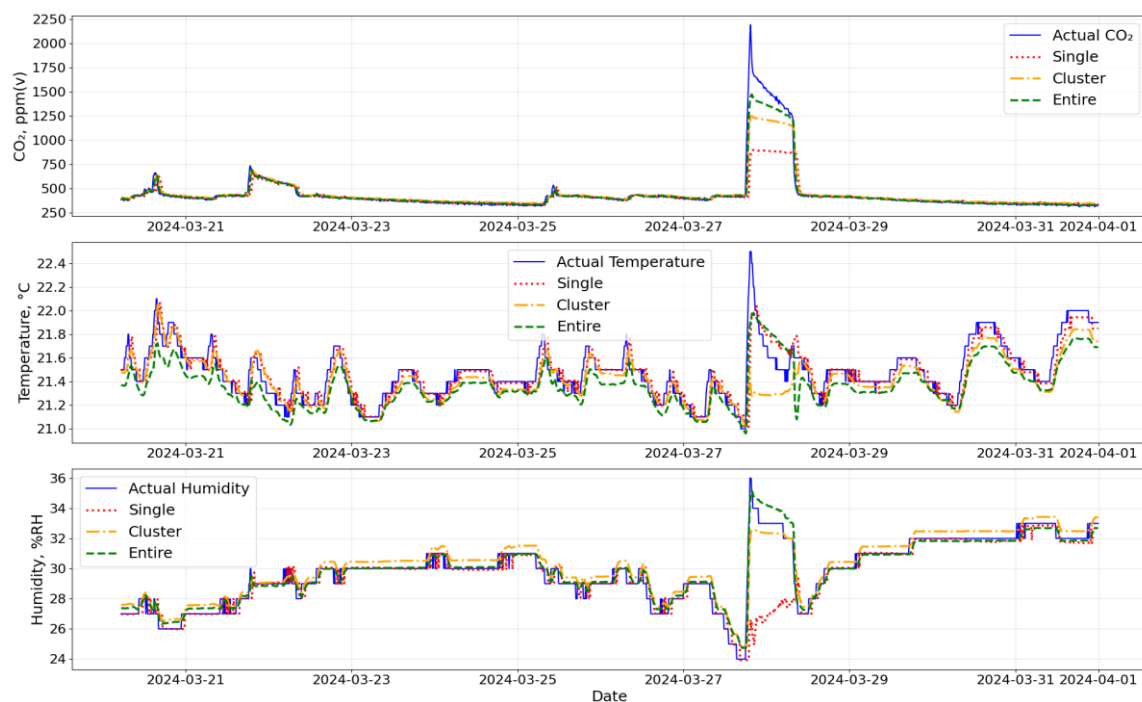
The LSTM, Transformer, and GRU models are used to analyse the dataset. Prophet and KAN could not handle many variables or returned NaN/infinity errors after clusterization.

The approach used in Section 4.1 is to create an individual data forecast model for each sensor to predict temperature, CO<sub>2</sub>, and RH. This makes forecasting more responsive in time and is the only way we can compare the Prophet model with the others. However, making individual models for each sensor takes more space. For example, 442 kB × 128 sensors are equal to ~56.6 MB in the case of LSTM. On the other hand, the model size for the entire dataset (2-month period of 128 IAQ sensor data) is around 1 MB up to 7 Mb (see the last column in Table 3).

In this part of the study, GRU performs better than LSTM and Transformer almost in every parameter. GRU prediction RMSE errors for temperature, CO<sub>2</sub>, and RH are 0.16, 28.51, and 0.75. Corresponding MAE values are 0.12, 17.97, and 0.57. Additionally, GRU performs well with the entire dataset and inside each cluster, especially in cluster 1 (medium temperature range).

Due to its attention mechanism, the Transformer model performs better across the board; however it has difficulty with higher temperatures in cluster 2, where it has larger errors (RMSEs of 57.35 for CO<sub>2</sub> and 2.04 for humidity).

While LSTM does quite well, GRU outperforms it overall due to its faster convergence and more accurate predictions. In conclusion, GRU proves to be the most effective model even without clustering, outputting more accurate predictions with the smallest error rate for CO<sub>2</sub>. A comparison of the single-sensor, cluster, and entire dataset model approaches for the same sensor is given in Figure 9 (here, the temperature and CO<sub>2</sub> variables are selected). The model based on single-sensor data performs the least accurate in the case of sudden event occurrence (the spike in the middle part of the graph). The best performance is achieved using the whole building's data for ML training. However, this technique requires the most computing resources. Therefore, sensor clusterization seems to be the most reasonable approach and can be adapted for specific use cases.



**Figure 9.** Actual (blue line) vs. predicted (dashed lines) data using GRU model for the sensor with the highest forecast RMSE (the case of a single-sensor model approach): single-sensor model approach (green); cluster model approach (orange); entire dataset model approach (red).

## 5. Conclusions

The study reported in this paper discusses the developed IAQ sensor data system designed for a public sector building, with monitoring, analysis, and prediction functionalities corresponding to the building's digital shadow. Whether IAQ levels meet specified health standards depends on the operation of the HVAC system, which in turn is one of buildings' most energy-intensive systems. One of Riga Technical University's campus buildings, equipped with an IAQ sensor network, was used as an experimental testbed. Our digital shadow model periodically updates the database with current IAQ sensor readings and

performs data representation, sorting, and report generation, and also forecasting. The main emphasis is put on sensor data prediction model selection. The novelty is the use of the KAN model and an additional fine tuning of integrated hyperparameter optimization using Optuna. We compare five state-of-the-art ML models (Prophet, LSTM, Transformer, KAN, GRU) according to the accuracy of predictions using RMSE and MAE metrics.

The key conclusions are summarized below.

- (1) KAN and GRU outperform other models (LSTM, Transformer, and Prophet) regarding prediction accuracy. This is due to the short (2-month) period of IAQ sensor data. It is believed that models like LSTM would perform much better if a longer input data period were used.
- (2) GRU is significantly more efficient than the KAN model regarding the computation time (14 min versus 8 h). However, it should be noted that the current KAN model is not optimized for speed.
- (3) Clusterization leads to stronger neuron network links, but results in worse MAE and RMSE values.
- (4) The LSTM prediction model size for a 2-month period of capturing 128 IAQ sensor data equals 1.164 MB. Creating individual data forecast models for each sensor requires more space (~56 MB) compared to one model for the entire building.
- (5) The forecasting system, especially GRU, is scalable and adaptable, making it suitable for application in other public sector buildings with a similar infrastructure.

Future research will include developing a hybrid system that optimises HVAC operations by using IAQ predictions (regression), including the buildings' occupant feedback through a voting system. The system will employ the LangChain AgentExecutor to autonomously manage the HVAC system in accordance with specified air quality criteria and monitor anticipated IAQ values. To maintain optimal conditions before the air quality deteriorates, the HVAC system will be turned on in advance when the prediction indicates that IAQ levels will surpass specified criteria. It is planned to test this approach on a real air conditioning unit to assess its effectiveness in balancing air quality management and energy efficiency. The use of LangChain AgentExecutor will enable dynamic, intelligent control, enhancing both adaptability and operational performance.

**Author Contributions:** Conceptualization, R.S., A.Z., A.N. and A.S.; methodology, R.S. and A.Z.; validation, R.S. and A.Z.; formal analysis, A.N. and V.N.G.J.S.; investigation, R.S., A.Z. and A.S.; writing—original draft preparation, R.S., A.Z. and A.S.; writing—review and editing, A.N. and V.N.G.J.S.; supervision, V.N.G.J.S. and A.S.; funding acquisition, V.N.G.J.S. and A.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** R.S., A.Z., A.N., and A.S. acknowledge that this research is funded by the Latvian Council of Science project “Smart Materials, Photonics, Technologies and Engineering Ecosystem” No VPP-EM-FOTONIKA-2022/1-0001. We also pay gratitude to the company SAF Ltd. for providing “ARANET” IoT sensor network. V.N.G.J.S. acknowledges that work is funded by FCT/MECI through national funds and when applicable co-funded EU funds under UID/50008: Instituto de Telecomunicações.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The original contributions presented in this study are included in the article.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

IAQ	Indoor air quality
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
GRU	Gated Recurrent Unit
KAN	Kolmogorov–Arnold Networks
ML	Machine Learning
HVAC	Heating, Ventilation, and Air Conditioning
LSTM	Long Short-Term Memory
MLPs	Multi-Layer Perceptrons
RNN	Recurrent neural network
es	EarlyStopping
rlr	ReduceLROnPlateau
mcp	ModelCheckpoint
mse	Mean Square Error
val_loss	Validation loss

## References

- Canha, N.; Correia, C.; Mendez, S.; Gamelas, C.A.; Felizardo, M. Monitoring Indoor Air Quality in Classrooms Using Low-Cost Sensors: Does the Perception of Teachers Match Reality? *Atmosphere* **2024**, *15*, 1450. [CrossRef]
- Dimitroulopoulou, S.; Dudzińska, M.R.; Gunnarsen, L.; Hägerhed, L.; Maula, H.; Singh, R.; Toyinbo, O.; Haverinen-Shaughnessy, U. Indoor air quality guidelines from across the world: An appraisal considering energy saving, health, productivity, and comfort. *Environ. Int.* **2023**, *178*, 108127. [CrossRef] [PubMed]
- Palaić, D.; Matetić, I.; Ljubic, S.; Štajduhar, I.; Wolf, I. Data-driven Model for Indoor Temperature Prediction in HVAC-Supported Buildings. In Proceedings of the 2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), Tenerife, Spain, 19–21 July 2023; pp. 1–6. [CrossRef]
- Saadatifar, S.; Sawyer, A.O.; Byrne, D. Occupant-Centric Digital Twin: A Case Study on Occupant Engagement in Thermal Comfort Decision-Making. *Architecture* **2024**, *4*, 390–415. [CrossRef]
- Mousavi, Y.; Gharineiat, Z.; Karimi, A.A.; McDougall, K.; Rossi, A.; Gonizzi Barsanti, S. Digital Twin Technology in Built Environment: A Review of Applications, Capabilities and Challenges. *Smart Cities* **2024**, *7*, 2594–2615. [CrossRef]
- Hauer, M.; Hammes, S.; Zech, P.; Geisler-Moroder, D.; Plörer, D.; Miller, J.; Van Karsbergen, V.; Pfluger, R. Integrating Digital Twins with BIM for Enhanced Building Control Strategies: A Systematic Literature Review Focusing on Daylight and Artificial Lighting Systems. *Buildings* **2024**, *14*, 805. [CrossRef]
- Qian, Y.; Leng, J.; Zhou, K.; Liu, Y. How to measure and control indoor air quality based on intelligent digital twin platforms: A case study in China. *Build. Environ.* **2024**, *253*, 111349. [CrossRef]
- Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access* **2020**, *8*, 108952–108971. [CrossRef]
- Saini, J.; Dutta, M.; Marques, G. Machine Learning for Indoor Air Quality Assessment: A Systematic Review and Analysis. *Environ. Model. Assess.* **2024**, 1–8. [CrossRef]
- Aranet. User Guide Aranet4 HOME/Aranet4 PRO. Available online: [https://aranet.com/attachment/273/Aranet4\\_User\\_Manual\\_v24\\_WEB.pdf](https://aranet.com/attachment/273/Aranet4_User_Manual_v24_WEB.pdf) (accessed on 27 December 2024).
- Aranet4 HOME. Available online: <https://aranet.com/en/home/products/aranet4-home/> (accessed on 27 December 2024).
- Aranet. Aranet Radio benefits, Aranet Radio vs. LoRaWAN. 8 November 2022. Available online: [https://pro.aranet.com/uploads/2022/11/aranet\\_radio\\_vs\\_lorawan\\_v4.pdf](https://pro.aranet.com/uploads/2022/11/aranet_radio_vs_lorawan_v4.pdf) (accessed on 27 December 2024).
- Arturs Ziemelis, Ruslans Sudniks, and Andis Supe, VPP Mote [Data set]. Kaggle. 2024. [CrossRef]
- Scikit-Learn. Metrics and Scoring: Quantifying the Quality of Predictions. Available online: [https://scikit-learn.org/stable/modules/model\\_evaluation.html#mean-absolute-error](https://scikit-learn.org/stable/modules/model_evaluation.html#mean-absolute-error) (accessed on 20 December 2024).
- Scikit-Learn. Root\_Mean\_Squared\_Error. Available online: [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.root\\_mean\\_squared\\_error.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.root_mean_squared_error.html) (accessed on 20 December 2024).
- Archive of Air Temperature and Wind Speed Data. Available online: <https://www.meteolapa.lv/arhivs/1217/riga/01-02-2024/31-03-2024> (accessed on 18 December 2024).
- Sudniks, R.; Ziemelis, A.; Spolitis, S.; Nikitenko, A.; Supe, A. Development of Building Indoor Air Quality Monitoring Based on IoT Sensor Network. In Proceedings of the 2024 Photonics & Electromagnetics Research Symposium (PIERS), Chengdu, China, 21–25 April 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–7. [CrossRef]
- KindXiaoming. KAN GitHub. Available online: <https://github.com/KindXiaoming/pykan> (accessed on 13 November 2024).

19. Popescu, M.-C.; Balas, V.; Perescu-Popescu, L.; Mastorakis, N. Multilayer perceptron and neural networks. *WSEAS Trans. Circuits Syst.* **2009**, *8*, 579–588.
20. Taylor, S.J.; Letham, B. Forecasting at scale. *PeerJ Prepr.* **2017**, *5*, e3190v2. [[CrossRef](#)]
21. Liu, K.; Zhang, J. A Dual-Layer Attention-Based LSTM Network for Fed-batch Fermentation Process Modelling. In *Computer Aided Chemical Engineering*; Elsevier: Amsterdam, The Netherlands, 2021; Volume 50, pp. 541–547. ISBN 978-0-323-88506-5.
22. Ebrahimi, Z.; Loni, M.; Daneshtalab, M.; Gharehbaghi, A. A review on deep learning methods for ECG arrhythmia classification. *Expert Syst. Appl. X* **2020**, *7*, 100033. [[CrossRef](#)]
23. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2023**, arXiv:1706.03762.
24. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; ACM: New York, NJ, USA, 2019; pp. 2623–2631. [[CrossRef](#)]
25. KMeans. Available online: <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html> (accessed on 18 December 2024).

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.