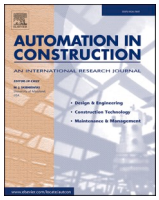
Automation in Construction 139 (2022) 104261

Contents lists available at ScienceDirect

Automation in Construction

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



IoT-based platform for automated IEQ spatio-temporal analysis in buildings using machine learning techniques

Francisco Troncoso-Pastoriza [a], Miguel Martínez-Comesaña [b,*], Ana Ogando-Martínez [b], Javier López-Gómez [b], Pablo Eguía-Oller [b], Lara Febrero-Garrido [a]

[a] Defense University Center, Spanish Naval Academy, Plaza de Espãna, s/n, 36920 Mar'ın, Spain
[b] Department of Mechanical Engineering, Heat Engines and Fluids Mechanics, Industrial Engineering School, University of Vigo (Universidade de Vigo), Maxwell s/n, 36310 Vigo, Spain

ARTICLE INFO

Keywords:
IEQ
Sensor network
MLP
Random Forest
Support vector regression

ABSTRACT

Providing accurate information about the indoor environmental quality (IEQ) conditions inside building spaces is essential to assess the comfort levels of their occupants. These values may vary inside the same space, especially for large zones, requiring many sensors to produce a fine-grained representation of the space conditions, which increases hardware installation and maintenance costs. However, sound interpolation techniques may produce accurate values with fewer input points, reducing the number of sensors needed. This work presents a platform to automate this accurate IEQ representation based on a few sensor devices placed across a large building space. A case study is presented in a research centre in Spain using 8 wall-mounted devices and an additional moving device to train a machine learning model. The system yields accurate results for estimations at positions and times never seen before by the trained model, with relative errors between 4% and 10% for the analysed variables.

1. Introduction

Meeting occupants' needs is a key requirement in the Architecture, Engineering, Construction, and Operation (AECO) sector. Buildings and infrastructure directly affect their users [1]. Factors such as health, comfort, accessibility and productivity are essential to provide an adequate environment for occupants [2]. Comfort, specifically, is especially relevant in terms of social, environmental, and economic aspects [3]. Indoor environmental quality (IEQ) has a direct impact on the occupants of a building, and IEQ-based factors can be used to determine the range of acceptable comfort levels [4]. In fact, indoor air pollution is the leading cause of 1.6 million premature deaths per year, according to the World Health Organization. However, it is infrequent to systematically recognise IEQ and health as key issues in localised green building codes, especially in the developing world [5].

In this context, many companies have started the development of applications based on the Internet of Things connected to low-cost sensors to monitor IEQ variables [6]. The selection of measured parameters and sensors is important to provide relevant results. Most studies that analyse IEQ include indoor thermal comfort assessment (temperature and relative humidity), CO_2 sensing and particle concentration [6]. Lighting is also an important factor that may significantly influence IEQ. [7]. External weather conditions are another critical source of information for IEQ analyses. Variables such as air temperature, air relative humidity or solar irradiation have a strong effect on the thermal fluxes entering and exiting a building, and hence affect the IEQ of indoor spaces, both directly and through their effects on the heating, ventilating and air conditioning needs [8]. Regarding sensors, there are several low-cost options available in the market, with varying levels of performance [9,10].

Tools used to study these comfort-related parameters should provide good interoperability and visualisation methods to identify potential problems [2]. Moreover, the evaluation of the IEQ conditions should not be limited to a few points in a specific building zone when aiming to provide a detailed analysis, especially in large open spaces. Thus, many studies present mobile devices, or fixed low-cost sensors to acquire values at different locations across the building space [11–13]. However, those methods relies on the constant use of the mobile devices to provide inputs. To address this problem, artificial intelligence can be used to obtain values at many points in the entire room. This is done

* Corresponding author.
E-mail address: migmartinez@uvigo.es (M. Martínez-Comesaña).

https://doi.org/10.1016/j.autcon.2022.104261
Received 15 June 2021; Received in revised form 1 March 2022; Accepted 10 April 2022
Available online 26 April 2022

without requiring the continuous use of mobile sensing devices while preserving a comparable accuracy [14,15].

Nowadays, the use of artificial intelligence in energy efficiency and indoor environment analysis in buildings is widespread. The algorithms within this approach, know as black box or machine learning models, are characterised by their ease of implementation, robustness and high performance [16–18]. They are capable of replicating complex patterns without requiring specific knowledge about detailed physic relations behind the study subject [17]. Moreover, there exist numerous algorithms, each with its own characteristics, to carry out the learning process but the best known are the Artificial Neural Networks (ANN). ANN are mathematical models that aim to emulate the behaviour of the biological neurons. They have a remarkable ability to model non-linear relations between the inputs and the outputs of the model thanks to their extensive inter-connectivity [19]. There are also different types of ANNs, but the most widely used is the MultiLayer Perceptron (MLP) neural network. This specific type of model is characterised by its structure divided into layers connecting the inputs to the targeted output [20]. In recent years MLP neural networks have been applied in numerous related fields such as indoor environment [21,22], sensors [23,24], energy efficiency in buildings [18,19] or computer communications [25,26].

Other machine learning algorithms have also demonstrated their efficiency extracting patterns of data. Along with MLP neural networks, Random Forest (RF) and Support Vector Regression (SVR) are some of the most well-known algorithms in the context of building analysis. On the one hand, the RF algorithm is based on individual regression trees. Through techniques such as *bagging* the original data or *splitting* the trees, the most significant information is selected to yield the final predictions. Its great adaptability and simple implementation have allowed its expansion to a wide variety of scientific fields as building energy consumption [27], health [28,29], environment [30,31] or software controller [32,33]. On the other hand, the SVR algorithm differs from others by attempting to minimise an upper bound on the generalisation error instead of minimising the prediction error (known as empirical risk minimisation) [34,35]. In addition, considering the dual form of the problem and using a specific loss function also makes it stand out [18,36]. Thus, this algorithm has been extended to different study areas such as energy analysis in buildings [18,37], renewable energies [35,36], industrial processes [34,38], finance [39] or electrical engineering [40,41]. The selection of these particular algorithms, together with linear regression as a classical technique, is due to the aim of comparing algorithms with significantly different characteristics and which were already applied in similar fields.

The aim of this paper is to present an automated low-cost system to monitor the IEQ of a large building space, combined with the implementation of machine learning models that generate estimations throughout the entire analysed space (horizontally and vertically). Thus, we will be able to accurately control the environmental conditions of a building in real time. In the present study, this system is tested on a research centre located in north-western Spain. From the set of IEQ-related variables that could be monitored and analysed, three are selected: temperature, relative humidity and CO2 concentration. In this case, the available data are four months of minute frequency values for each of the variables analysed. Three different models, based on three different machine learning algorithms, are trained with the monitored data to analyse and compare their efficiency interpolating the selected variables. In addition, three temporal variables (hour of the day, day of the week and day of the year) and 10-min frequency weather data of the area are also considered as model inputs. Thus, the trained machine learning models will be capable of detecting changes in IEQ distribution patterns such as opening door or windows [16]. First, the efficiency analysis consists of a positional cross-validation experiment, in which the model estimates values at each of the positions where the mobile device was placed, previously extracting these information from the training sample. Second, once the best algorithm is identified, a temporal analysis focused on interpolating the values of the variables throughout two entire days is presented. The metrics selected to evaluate the efficiency of the machine learning models are the computational time required for training, the the Normalised Mean Biased Error (NMBE) and the Coefficient of Variation of the Root Mean Square Error (CV(RMSE)), as recommended by the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). Moreover, the possibility of reducing the number of sensors, used to produced interpolations through machine learning models, using multi-objective genetic algorithms was analysed in another study [42]. This allows reducing the cost of the installation and the possible interference with the normal activities of the building's users.

The novelty of this work lies in the implementation of an automated monitoring system alongside a machine learning algorithm to collect, control and estimate the indoor environmental conditions at every position in a large multi-zone building space with only a few acquisition points that may be positioned outside the occupation area. The proposed system provides a method to show, at every minute, a real-time image of the variables that summarise the IEQ and their distribution throughout the building. In this way, this information could be integrated in a digital twin representation of the analysed building [43].

2. Materials and methods

The objective of the proposed system is to provide detailed environmental conditions for the entire volume of a building space based on the information acquired from a small predefined set of fixed points inside it. It comprises a set of hardware and software components to acquire indoor and outdoor environmental information, process and analyse these data, and visualise all the collected information.

To deploy the platform and obtain accurate results, first an initial acquisition step is performed with the wall-mounted devices at fixed locations and the mobile device obtaining data for different positions in the room. The information gathered in this step feeds the training of the system, in order to produce accurate results for different spatial locations. After that, the mobile device is removed and the only available information comes from the wall-mounted devices and the initial training. Thus, no further human intervention is required for the system to produce the desired data.

This section describes the hardware used to physically obtain and send the information in a given building space and the software platform to acquire additional information and handle the communication, storage, analysis and user interaction.

2.1. Hardware components

As previously mentioned, ad-hoc hardware components were developed to obtain indoor environmental conditions. Different types of devices were designed to accommodate to several acquisition requirements: first, wall-mounted devices acquire data at fixed locations in the space with different sets of sensors; second, a mobile device is used to obtain data at different positions.

The core of the devices, common to all of them, comprises a set of sensors connected to a Raspberry Pi Zero to do the processing and the transmission via WiFi. Each device has a different collection of sensors, detailed in Table 1, using either I^2C-based or serial-based communication, with a total of four models: WMA, WMB and WMC for the three versions of the wall-mounted devices, and MOB for the mobile device. The structural components of all the devices were 3D printed using polylactic acid (PLA) as the base material, with additional metal parts for the pole and handles of the mobile device to provide the required structural integrity to support the moving pan and tilt unit.

2.1.1. Wall-mounted devices

These devices are designed to be mounted in a vertical surface to provide continuous environmental information of a single point in

Table 1
List of sensors included in the devices.

Model	Measurements	Features	WMA	WMB	WMC	MOB
adxl345	Acceleration	Resolution: ±2g – ±16 g				H
amg8833	Temperature	Range: 0°C – 80°C; Acc.: ±2.5°C				H
bme680	Temp./Rel. humidity/ Pressure	Acc.: ±1.0°C, ±3%H, ±1 hPa	X	X		X
hmc5883l	Magnetic field	Range: −8 – +8 G				H
itg3200	Angular velocity	Acc.: 2%				H
mhz14	CO_2 concentration	Range: 0–10,000 ppm; Acc.: ±50 ppm ±5%	X	X	X	X
mlx90614	Surface temperature	Range: −70°C – +380°C; Acc.: ±0.5°C	X	X	X	H
pt100	Radiation temperature		X			
rd200m	Radon concentration	Range: 0.2–99.9 pCi/L; Acc.: ±10%	X			
sds011	Particle concentration	Range: 0.0–999.9 $\mu g/m^3$; Acc.: max(±10$\mu g/m^3$, 15%)	X			
sht31d	Temp./Rel. humidity	Acc.: ±0.3°C, 2%H	X	X	X	X
TFmini	Distance	Range: 0.3–12 m; Acc.: ±4cm (0.3–6 m), ±6 cm (≤12 m)				H
tsl2561	Illuminance	Range: 0.1–40,000 lx	X	X		H
tsl2591	Illuminance	Range: 188 μlux – 88,000 lx			X	

WMA, WMB, WMC: wall-mounted devices with different sets of sensors; MOB: mobile device. "H" indicates that the sensor is placed in the head of the mobile device.

space. They are composed of a rigid box with ventilation and perforations to enable the acquisition of external environment information. Fig. 1 shows these devices, both internally and externally, and the corresponding CAD model used for 3D printing and prototyping.

Model WMA contains additional sensors with higher costs for parameters with a small variations relative to the spatial location, such as radon concentration. Modesl WMB and WMC have minor differences, sharing the same set of measurable parameters.

2.1.2. Mobile device

The mobile device comprises the pan and tilt head with sensors mounted on a movable elevating platform. The design is divided into three sections: base, elevating platform, and pan and tilt head. The complete system is shown in Fig. 2, including the CAD model and the internal components of the head.

The base contains the main structural components of the device. It includes the wheels, handle, pole and battery receptacle. There are two battery slots to enable hot swapping, with one LED for each of them to indicate charge status. The battery swapping logic is controlled by an Arduino Nano, that switches the power input depending on the presence and voltage of the batteries. The pole includes a toothed belt for the vertical movement of the elevating platform.

The platform contains another Arduino Nano that controls the stepper motor for its own vertical movement and the two servomotors for the pan and tilt structure to change the orientation of a head with additional sensors. The platform contains the same logic design as the wall-mounted devices, with a different physical arrangement to separate the sensors that are placed in the head from those that are in the main non-rotating part.

2.2. Software platform

The system comprises several software modules to send, store and process the data obtained from the sensors. The general client-server architecture is illustrated in Fig. 3, containing client nodes that gather data from sensors and a central server node to store and process all the information from the clients. Additionally, the server node collects data from third-party meteorological sources to obtain environmental information of the building exterior.

The communication between the server and each client node inside the acquisition devices is based on the Advanced Message Queuing Protocol (AMQP) [44], using RabbitMQ message brokers on each side. There are two internal communication channels, one for sensor data and other for device status, that are processed in different ways on the server. To relay messages from client to server, the RabbitMQ shovel plugin is configured in the client nodes to redirect incoming messages from the internal exchanges to the corresponding queues in the server. Messages are configured to be persistent and queues to be durable, using the corresponding configuration parameters, to ensure that the information is persisted to disk in case of network errors. Moreover, authentication, authorisation and access control are also configured to restrict communication to valid nodes.

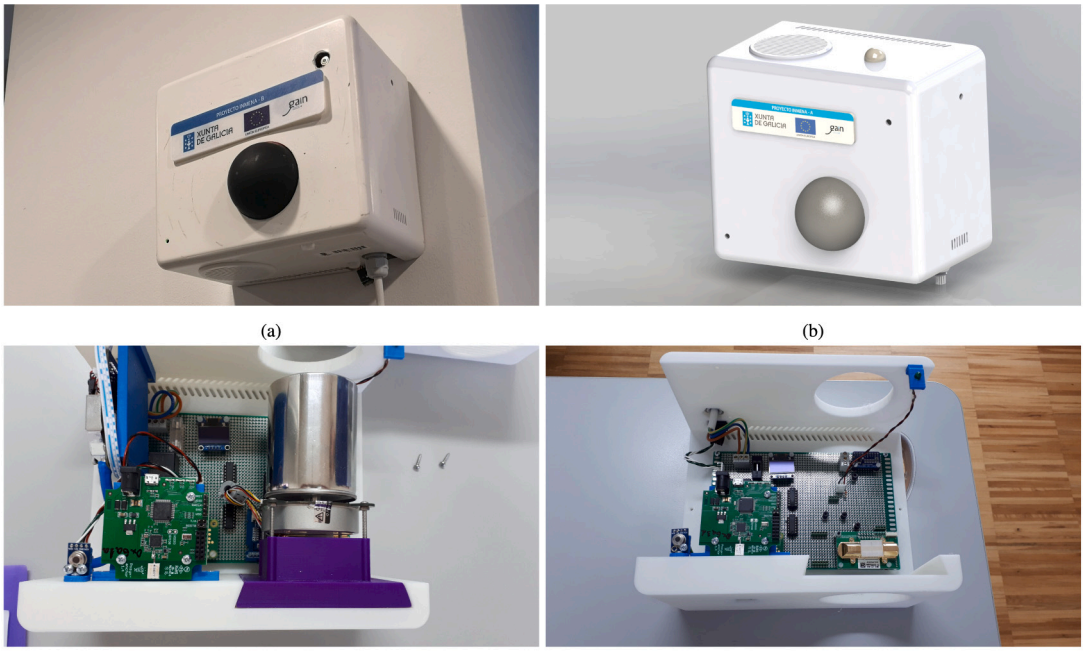
2.2.1. Client node

The client node is responsible for reliably sending sensor data and device status. It is composed of three main parts, as depicted in Fig. 4: (i) a Telegraf node that gathers internal information that is send to the "status" exchange; (ii) an ad-hoc sensor reading software that collects sensor values and sends them to the "data" exchange; (iii) a RabbitMQ broker that relays messages from both exchanges to queues in the server using the shovel plugin. The acquisition software is implemented in the Python programming language, using dedicated modules for each sensor in the client node. This software automatically connects to the appropriate drivers and schedules readings based on a configuration file that contains several parameters for each of the sensors in the system.

2.2.2. Server node

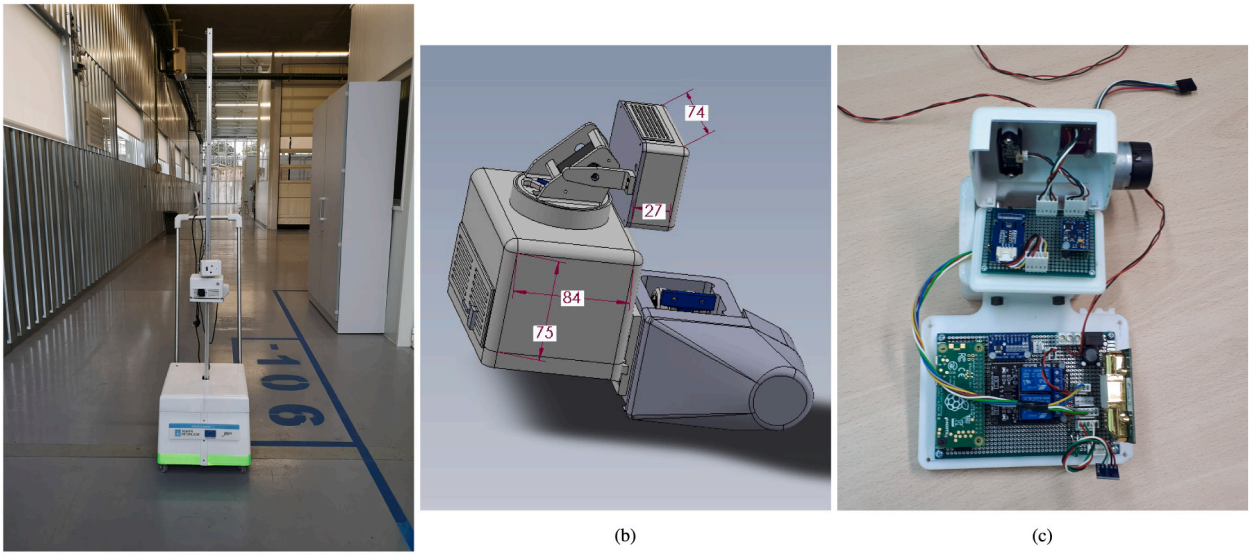
The server node comprises the set of logical components illustrated in Fig. 5. First, a RabbitMQ broker is configured with two queues, "data" and "status", to listen for incoming messages from the client nodes. These messages are forwarded to a Telegraf agent that relays them to different databases, depending on the input RabbitMQ exchange. At the same time, a collecting agent is used to actively request meteorological information from third party sources at regular intervals though the corresponding communication layer to provide up-to-date weather information.

The main storage system is an InfluxDB-based system [45] comprised of two main time series databases: "status", to store transient information about the operational state of the client devices, with a configured retention policy of 24 h; and "data", to save the acquired measurements from sensors and meteorological sources indefinitely. The "status" database contains information such as CPU and RAM usage, available disk space and uptime, with one InfluxDB measurement for each of these parameters. This allows for the identification of potential issues with the operation of the client devices. The internal structure of "data" is divided based on the origin of the information, with a different measurement for each sensor and third-party meteorological source. For each measurement, several InfluxDB tags are defined depending on the given



(a) (b) (c) (d)

Fig. 1. Wall-mounted devices: (a) installed device; (b) CAD model; (c) internal design for WMA and (d) for WMB, with WMC being almost identical to WMC.



(a) (b) (c)

Fig. 2. Mobile device: (a) assembled device; (b) CAD model of the header, with distances in mm; (c) internal design with the sensors.

measurement, with only one tag being mandatory for all measurements: "host", which indicates the client node in which the reading was performed for sensor data or the specific weather station identifier for meteorological information. InfluxDB fields include values for parameters such as temperature, humidity and CO_2 concentration, depending on the output values of the specific sensor. All this information is ready to be exposed for a digital twin representation of the building, including real-time values of the IEQ conditions.

Visualisation and management is done using three main nodes: Kapacitor, Chronograf and Grafana. Kapacitor enables alerting based on rules for the detection of an anomalous operation of the devices. It is configured to send automatic alerts based on unfeasible sensor values in "data" for each physical property and anomalous parameters in "status" for fast malfunction detection. Chronograf is the main management interface, allowing for data exploration, access control management and alert configuration. Finally, Grafana is used as the visualisation front-end, with preconfigured charts for the most relevant information. The visualisation is divided into two separate dashboards, corresponding to the two databases in the system. The first one shows relevant status information for each client device, while the second shows sensor values for different parameters. Examples of these two dashboards are displayed in Fig. 6. The visualisation system presented in this work is just one of many potential integrations with other tools that could acquire the input information from the core database to provide additional insights and analytics.

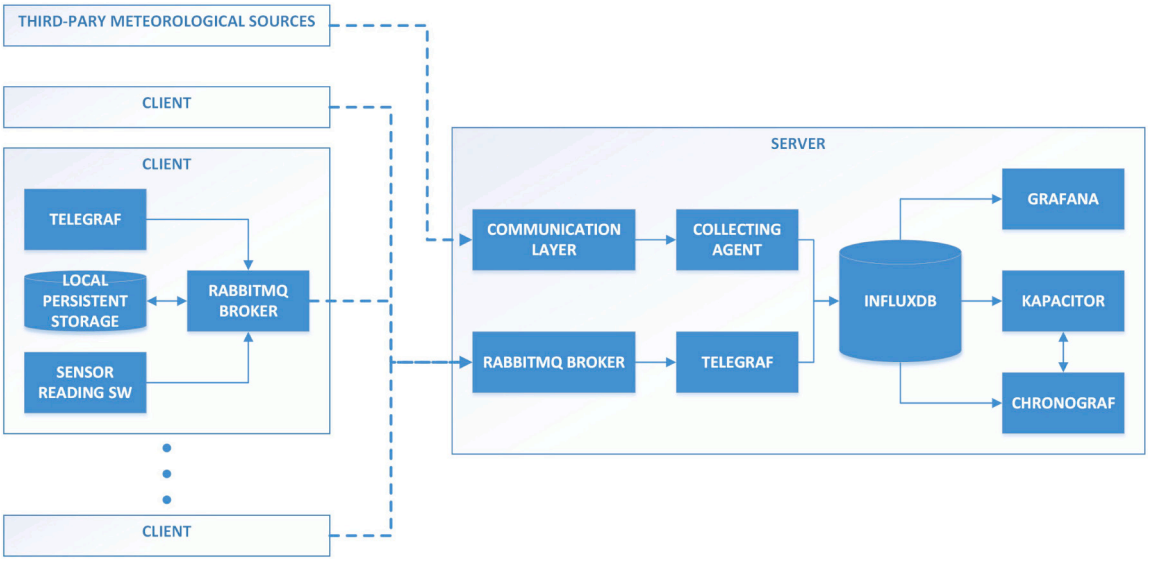


Fig. 3. High-level block diagram of the software platform, exposing the underlying client-server architecture.

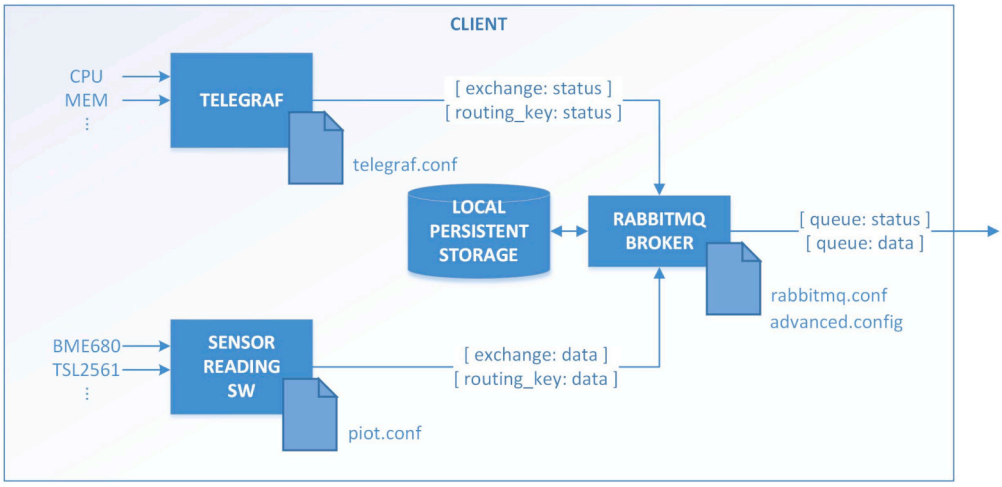


Fig. 4. High-level block diagram of the client node.

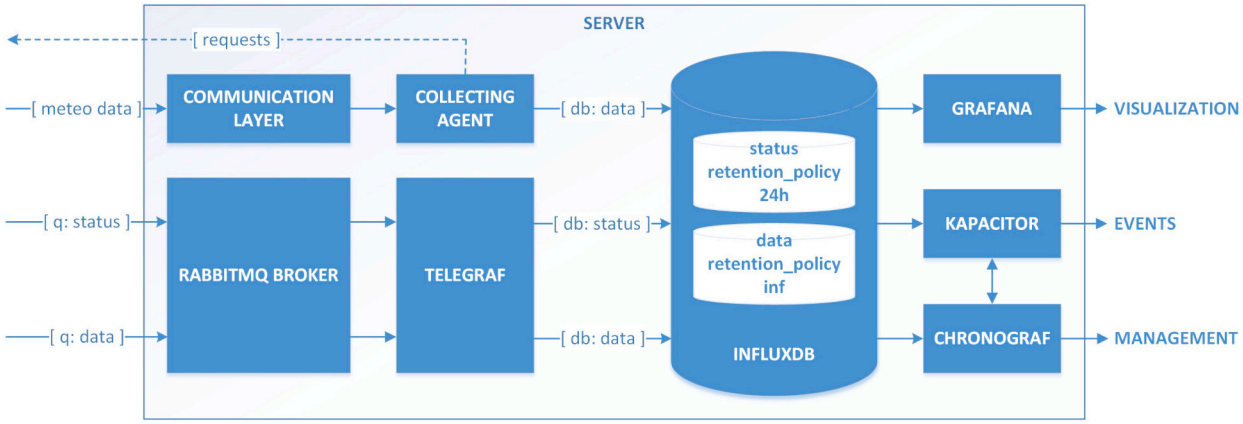


Fig. 5. High-level block diagram of the server node.

(a)

(b)

Fig. 6. Visualisation dashboards for (a) device status and (b) sensor data.

2.3. *Artificial intelligence algorithms*

In this paper, three different artificial intelligence algorithms, in addition to the classical linear regression introduced as a comparison, were tested to try to interpolate the environmental conditions inside a building. In particular, the machine learning techniques that were taken into account are: MultiLayer Perceptron (MLP) neural network, Random Forest (RF) and Support Vector Regression (SVR).

2.3.1. *MultiLayer perceptron neural network*

MLP neural networks are the most widely used Artificial Neural Network (ANN) among feed-forward ANNs and they are composed of three different types of layers (see Fig. 7). The model inputs are introduced in the first or *input layer* and the results yielded by the model are given by the last or *output layer*. Moreover, there are also intermediate layers, known as *hidden layers*, which are interconnected and can be zero, one or more [19,46]. In each of the layers a specific number of neurons are distributed and this number is conditioned by the inputs and outputs of the model. As presented in Fig. 7, each neuron of the layers, except for the input layer, is fed from the previous layer's neuron [47–49].

In addition, Table 2 shows the specific MLP architectures used in this study to estimate each of the variables analysed. As in previous studies, such as [50,51], these architectures were obtained using the Non-Dominant Sorting Genetic Algorithm (NSGA-II), which significantly reduces the number of evaluations needed to yield the optimal result.

The aim of the built MLP neural network is to efficiently fit the internal parameters of the network to be able to obtain a predicted value $\hat{y}_i$ close to the real one y_i for $i = 1, \ldots, N$ (where N is the sample size). The task assigned to each network node is to calculate a weighted sum of its inputs and pass the sum through an activation function [47]. In this case, the activation function considered in model training is the Rectified Linear Unit function (reLU $= \max(0, \mathbf{x})$) [52]. The training process is performed with a backward propagation algorithm, which aims to

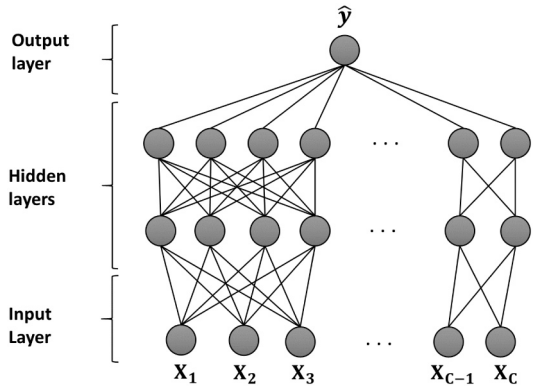


Fig. 7. MLP architecture with an input layer, several hidden layers and an univariate output layer.

Table 2
MLP architectures considered to estimate the values of each of the three variables related with the IEQ of the analysed building.

Variable	Hidden Neurons
Temperature	30–20
Relative humidity	110–10
CO_2 levels	20

minimise a specific cost function. Thus, the real values of the variables of interest must be known [19,53]. In this study the Mean Squared Error (MSE): $\frac{1}{n}\sum_{i=1}^{N}\left(\hat{y}_i - y_i\right)^2$ is the selected cost function. Furthermore, model training is performed through the *batch mode*: weights are updated with an average update, which is produced by incorporating all the patterns in the input file (an epoch) and accumulating all the individual updates [46]. On the other hand, due to its effectiveness for stopping the training process when the best generalisation is reached, the stop criterion selected is cross-validation [54]. This method stops the training when, after a certain number of epochs, the MLP performance, validated with a previously separated individual sample, begins to decrease or stagnates [18,19]. In this case the limit was set to 50 epochs. Lastly, every MLP built in this analysis was trained taking into account a Gaussian kernel initialiser, the Adaptive Moment Estimation (Adam) optimiser [55] and the mini-batch gradient descent algorithm [56]. Further information about MLP neural networks can be found in [19,49].

2.3.2. Random forest

RF algorithms are based on a set of D unpruned regression trees, built from a bootstrap sampling of the initial training data. The trees are structured in *root node*, *branch nodes* and *leaf nodes* [27,57] and, in each of the nodes, the optimal node splitting feature is sought among a set of c features (also randomly selected from the feature space with size C). On the one hand, $c < C$ causes a decrease in the correlation between the different trees, and therefore the average outputs is expected to have less variance than a single regression tree [38,58]. On the other hand, there exists a trade-off in the size of c: higher values of c can improve the predictive accuracy of individual trees but can also cause an increase in the correlation between trees, wasting any improvement in individual predictions. The RF training process (see Fig. 8) can be summarised as follows [31,38,58]:

1. *Bagging*: Based on the original training data set $S = \{(x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)\}$, *bagging* or *bootstrap aggregation* generates D new data sets S_i through N-size sampling with replacement of the original data

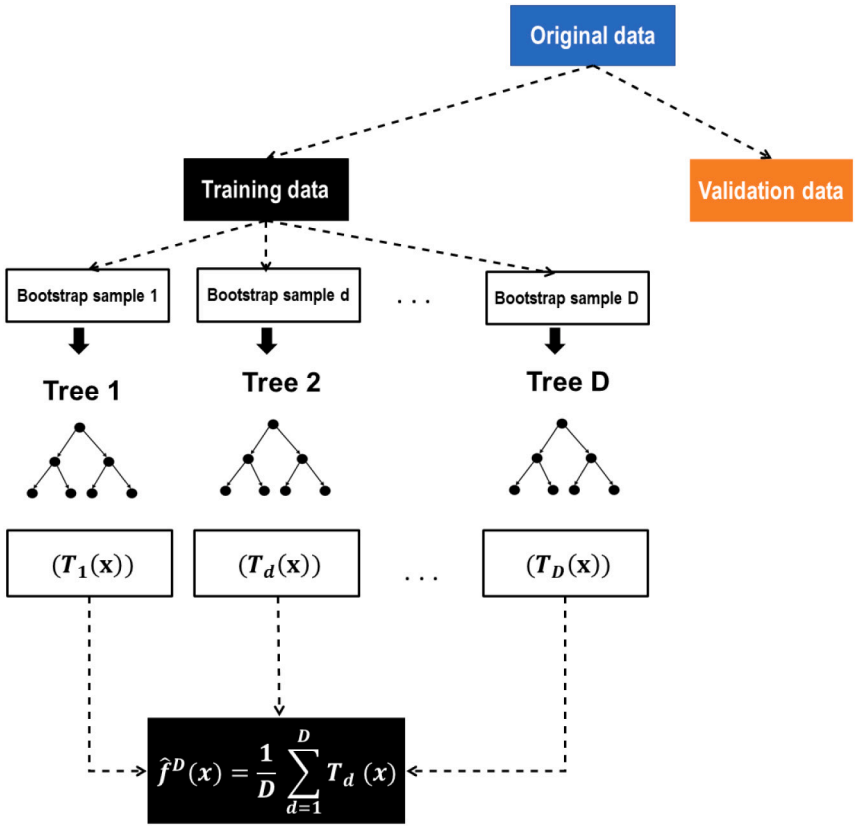


Fig. 8. Summary of the training process for the Random Forest algorithm. The way of obtaining the final output as a combination of the outputs of the individual trees ($T_d(x)$) is also presented.

set. The aim of this technique is to prevent over-fitting and to reduce the variance.

2. *Variables selection*: In each of the bootstrap samples S_i, an unpruned regression tree is formed as follows: in each node of the tree, c variables are randomly selected and then, the best split of them is chosen. This process is usually known as *feature bagging* [59].
3. *Process of splitting*: Given a partition of J regions R_1, R_2, …, R_J in which the output can be modelled by a constant a_j in each of the regions:

$$f(\mathbf{x}) = \sum_{j=1}^{J} a_j I(\mathbf{x} \in R_j), \tag{1}$$

the splitting criterion in each node is carried out by minimising the sum of squares ($E_{\mathbf{x},\,\mathbf{y}}(\mathbf{y} - f(\mathbf{x}))^2$). After several operations presented in [58] and in [38] and considering first a binary division, for any z and q, the solution for the minimisation problem is given by:

$$\widehat{a}_1 = \frac{\sum_{i=1}^{n} I(y_i | x_i \in R_1(z,q))}{n} \quad , \quad \widehat{a}_2 = \frac{\sum_{i=1}^{n} I(y_i | x_i \in R_2(z,q))}{n}. \tag{2}$$

In this way, the sample is divided into two regions, and this process will be repeated until some specific stopping criterion is reached.

4. *Stopping Criterion*: The splitting process continues until the size of S_i falls below a threshold. Once D regression trees T_d are constructed, the predicted value at a new point x is obtained through an average of the predictions from all the individual regression trees:

$$\widehat{f}^{D}(\mathbf{x}) = \frac{1}{D} \sum_{d=1}^{D} T_d(\mathbf{x}). \tag{3}$$

Furthermore, RF optimisation requires tuning multiple parameters. In this study, the optimal values of the parameters *max_depth*, *min_samples_split* and *n_estimators* (some of the most significant ones [60]) were found via a k-fold cross-validation process ($k = 5$). Further information about RF can be consulted in [27,58].

2.3.3. Support vector regression

SVR algorithms are characterised by creating a non-linear mapping of the input space into a feature space of a higher dimension and constructing a linear regression into this new feature space. This algorithm is focused on the *structure risk minimisation* (SRM) principle, which intends to minimise an upper limit of an overall error that takes into account the sum of training error and the confidence level [34,39]. Thus, SVR is highly effective at solving non-linear problems even with a small training sample [35]. Assuming that the sample set has the form $\{(x_i, y_i)\}_{i=1}^{N}$, SVR approximates the objective function as shown in Eq. 4 [36,37,39]:

$$y = s(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b \tag{4}$$

where $\phi(\mathbf{x})$ represents the non-linear mapping that connects the input space to a higher dimensional feature space. Additionally, $\mathbf{w}$ is the weight vector, b the bias term and $\mathbf{x}$ the input data. The aim of this algorithm is based on the search of a function $s(\mathbf{x})$ for which the highest deviation from training data is less than a predefined value ε, maintaining the maximum possible flatness [34,37]. Once the slack variable ξ_i, ξ_i^* is introduced, $\mathbf{w}$ and b can be obtained solving the minimisation problem presented in Eq. 5 [35,39]:

$$\begin{aligned} \underset{\mathbf{w},b}{\text{minimise}} \quad & G(\xi_i, \xi_i^*, \mathbf{w}) = \frac{1}{2} \| \mathbf{w} \|^2 + C \sum_{i=1}^{p} (\xi_i + \xi_i^*) \\ \text{subject to} \quad & y_i - \mathbf{w}^T \phi(\mathbf{x}) - b \leq \epsilon + \xi_i \\ & -y_j + \mathbf{w}^T \phi(\mathbf{x}) + b \leq \epsilon + \xi_i^* \\ & \xi_i, \xi_i^* \geq 0 \end{aligned} \tag{5}$$

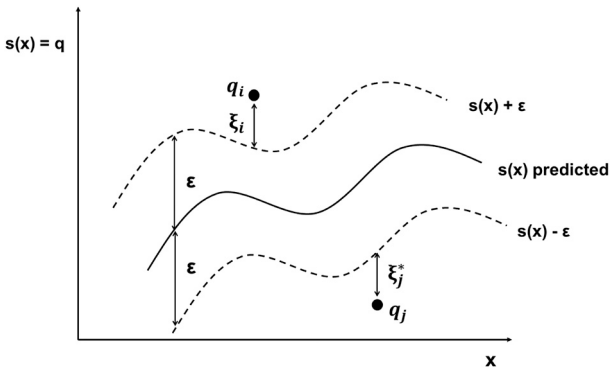


Fig. 9. SVR parameters considered in the training process.

being $\|\mathbf{w}\|^2$ a regularisation term, $C > 0$ the penalty parameter that determines the trade-off between model flatness and training error, and p the number of training patterns.

By means of the Lagrange duality, explained in Wei et al. [34] and Ahmad et al. [35], the optimised target function shown in Eq. 6 can be obtained (see Fig. 9:):

$$s(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b = \sum_{i=1}^{p} K(x_i, x)(\alpha_i^* - \alpha_i) + b \tag{6}$$

where $K(x_i, x)$ is the kernel function based on the inner product $\langle \phi(x_i), \phi(x) \rangle$ and α_i, α_i^* are the solutions to the dual problem (Lagrangian multipliers).

Lastly, SVR optimisation involves tuning certain parameters. In this case, the selected parameters to optimise through a k-fold cross-validation method ($k = 5$) were *epsilon*, *C* and *max_iter*. Further information about SVR can be consulted in [39,61].

3. Experimental system

The studied building is the Centre for Research in Technology, Energy and Industrial Processes (CINTECX), located at the University of Vigo in north-western Spain. A wide variety of activities take place inside this large room due to the different research groups working in it (energy, electronics, sustainability or automation). The installation of the monitoring devices is carried out on the ground floor of the building, where the laboratories are located. In this large multi-zone space, with approximately 825 m^2 and a height of 6 m, workers conduct the experimental part of their investigations. There is different equipment such as heaters, engines or boilers which have a significant influence in the environment quality. Specifically, the installed heating and cooling system comprises four fan coils placed throughout the room. Fig. 10 presents this building and some pictures of its interior.

The data considered for this study come from two different sources: the presented monitoring system and the nearest meteorological station, using a Numerical Weather Prediction model when real exterior weather values are not available [8].

3.1. Monitored data

The built system is composed of nine custom-made devices: 8 wall-mounted devices (see Fig. 1) and one mobile device (see Fig. 2). In this research, only sensors for temperature, relative humidity and CO_2 levels are considered. The mobile device (D0) allows collecting data from different positions in the work area. The fixed devices (D1 to D8) are placed at fixed locations at heights between 1.9 and 3.2 m where they do not interfere in the tasks of the building users.

All the collecting devices have the same sensors to monitor the indoor environmental conditions (see Table 1). In this case, SHT31-D

(a) Outside of the building. (b) Indoor corridor area. (c) Indoor working area.

Fig. 10. Photographs of the studied building.

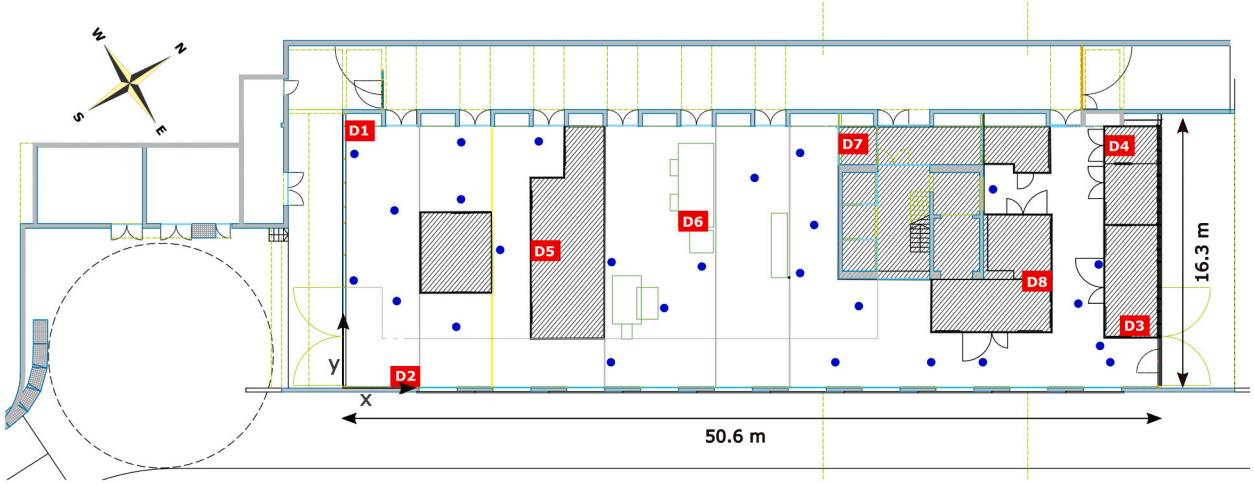


Fig. 11. Floor plan of the study area. Red boxes show the positions the wall-mounted devices, blue dots indicate the positions in which the mobile device has been temporarily located, and the hatched grey areas correspond to the closed laboratories that are omitted from the study. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

sensor is used to measure the temperature and the relative humidity and MHZ-14 sensor is used to measure CO_2 concentration. The specific positions where the fixed devices are placed are presented in Fig. 11 (length and width in the axes). The analysed room is a multi-zone open space although, in this research, closed areas are not considered and are represented in Fig. 11 hatched in grey. Moreover, this floor of the building has an envelope of concrete with numerous windows. The saw-toothed roof and the two shorter facades of the building are fully glazed. Additionally, one of the long sides of the building also has windows.

3.2. Weather data

The principal source of weather data was an automatic weather station belonging to the MeteoGalicia weather agency (MG). Said station is located 300 m northeast of the centre of the studied building, with an upward height difference of 43 m. The weather variables taken into account from this station are mean air temperature, relative humidity and global solar irradiation (measured at 1.5 m height). Accuracy, model and manufacturer of each sensor can be seen on Table 3.

Table 3
List of MG weather station sensors used as inputs in the machine learning models.

Sensor	Accuracy	Model	Manufacturer
Temperature	±0.25°C	HMP155	Campbell
Relative Humidity	±1% at 15 − 25°C, 0–90% RH		
Solar irradiation	<1.8%	SR01	Hukseflux

The station data are collected using an RSS service provided by MG [62]. These data are obtained every 10 min, with a time delay of 15–25 min. Additionally, the forecast values provided by the Global Forecast System surface flux (GFS sflux) model were used to fill missing or invalid data from the main source. GFS is a numerical weather model focused on global meteorological predictions and implemented by the National Oceanic and Atmospheric Administration (NOAA) of the United States. GFS is executed four times per day, at 00, 06, 12 and 18 h UTC, and their outputs are stored at the NOAA Model Archive and Distribution System (NOMADS) repository for 10 days [63]. Moreover, GFS sflux version offers forecasts each hour over a ≈ 13 km resolution horizontal grid. In addition, each time instant, weather conditions are extracted from the GFS sflux output file with the shortest available forecast step, at the nearest grid point. This point is located at coordinates 42.11519Â°N 8.67187Â° W; 5.94 km from the analysed building. A detailed

explanation of the model can be found at [64].

3.3. Data preprocessing

This study focuses on the use of the monitored data to interpolate, with machine learning algorithms, variables directly related to the IEQ of a building (indoor temperatures, relative humidity and CO_2 concentration). The available data are minute measurements from sensors placed in 8 wall-mounted devices on the one hand, and a mobile device on the other hand. The entire monitoring period was between 18 November 2020 and 10 March 2021, yielding a sample of about 135,000 observations (taking into account that many times not all wall-mounted devices are sending data) and considering 26 different positions in which the mobile device was placed. The mobility of this device allows the use of its distances, in three dimensions (x-, y- and z-axes), to the fixed devices as model inputs. Moreover, mobile device data (desired values) are temporarily interpolated because the device does not collect data during its vertical movement. Thus, a smoothing process of the devices data is performed to compensate this temporal interpolation and facilitate the model training process (see Fig. 12). Additionally, meteorological information of the area (section 3.2) and three temporal variables (hour of the day, day of the week and of the year) are also used as model inputs. Weather data values are replicated at every 10-min interval to generate data at the correct frequency required by the model. The structure of the model inputs and outputs implemented in this study is presented in Fig. 13. In this case, the model is trained to be capable of performing interpolations with up to three fixed devices malfunctioning.

Additionally, a comparison between the efficiency of the different machine learning algorithms presented is carried out (see section 4). It is based on the average errors yielded by each of them and the average computational times (measured in seconds) that they needed to train. Due to the correlation among the variables available for this analysis, the model inputs for each of the interpolated variables are different. Its structure is summarised as follows:

- Indoor temperature (45 inputs): Temperature values of fixed devices, outdoor temperature, solar radiation, time variables and distances to the fixed devices.
- Relative humidity (45 inputs): Relative humidity values of fixed devices, outdoor relative humidity, solar radiation, time variables and distances to the fixed devices.

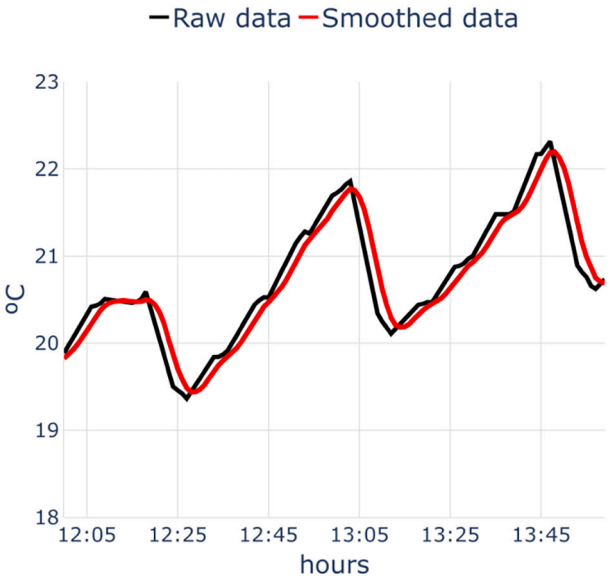


Fig. 12. Smoothing process applied to several hours of temperature data acquired with the mobile device.

- CO_2 levels (43 inputs): CO_2 values of fixed devices, time variables and distances to the fixed devices.

Finally, a time lag of one minute is considered in the indoor environmental variables due to the existing inertia in this type of variables [19].

3.4. Validation and error assessment

The validation metrics considered in this work to measure the accuracy of the machine learning models are the Coefficient of Variation of the Root Mean Square Error (CV(RMSE)) and the Normalised Mean Biased Error (NMBE):

$$\mathrm{CV(RMSE)} = 100\ x\ \frac{\sqrt{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2/N}}{\bar{y}}, \tag{7}$$

$$\mathrm{NMBE} = 100\ x\ \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)}{\sum_{i=1}^{N}(y_i)}. \tag{8}$$

They are both used to compare the performance of the models through a cross-validation process with average results presented in the next section. These metrics, which are recommended by ASHRAE, were used in similar studies such as [18,19,65]. In particular, it was demonstrated that CV(RMSE) is the best metric to analyse and evaluate building simulations [65].

4. Results and discussion

A system for estimating the environmental conditions in a large open space inside a building through data interpolation based on machine learning is presented in this paper. Specifically, as mentioned in the previous section the analysed building is a research centre and the available data are approximately four months of minute observations.

This section shows, first, the average accuracy and the average computational time required in the training process for each of the black box algorithms analysed through a positional cross-validation (see Table 4). This analysis is based on the estimation of the temperature, relative humidity and CO_2 concentration in each of the positions in which the mobile device was placed, comparing them with the real values. In this case, the models do not have any information about these positions, as these data were previously extracted from the training set. Then, the performance of the most efficient algorithm at estimating each of the three variables that summarise the IEQ of a building is analysed over two specific days (January 12 and February 2021). In this way, the algorithm parameters are adjusted to each of the interpolation problems (one per variable) creating three different models. Fig. 14 shows the capability of the selected models interpolating data from a temporal point of view. The results presented hereinafter are based on the Python programming language [66].

Regarding the temperature results,the RF model stands out from the other models in terms of relative errors and its variability in the results (see Table 4). While it obtained an average CV(RMSE) below 5% and a standard deviation of ±1.63, the liner regression, the MLP neural network and the SVR model show average CV(RMSE) values close to 10% with variabilities above ±6. Additionally, the average NMBE yielded by MLP and RF are very close to 0% and the variability of their results are ±5.83 and ±2.77, respectively. In contrast, SVR presents an average NMBE of 0.67% and the linear regression above 2%, both with higher standard deviations than the ones exhibited by MLP and RF. In terms of average computational times, Table 4 shows that the linear regression and SVR are the ones that obtained notably lower results (less than 30 s) than the values given by MLP and RF (higher than one minute). Nevertheless, this improvement in training time causes relative errors to increase significantly in these models, which make them unsuitable. In particular, the RF model is the most efficient because it

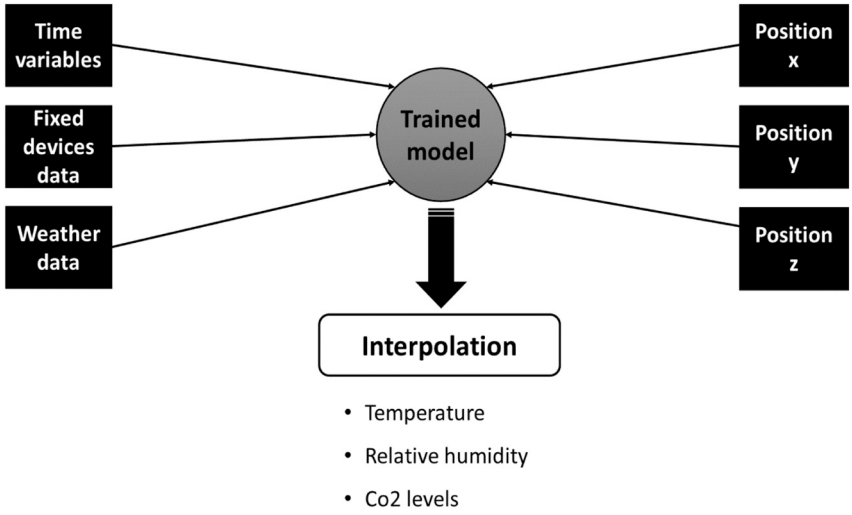


Fig. 13. Summary of the *interpolation* process after train the machine learning method. Black boxes represent the inputs and the list of variables the outputs from the *interpolation* process.

Table 4
Numerical results of the interpolations of the three variables studied through a positional cross-validation. The average CV(RMSE) and NMBE together with their standard deviations (SD) and their computational times (c.t.), in seconds, required to be trained are presented.

	Temperatures					Relative Humidity					CO_2				
Models	CV(RMSE) [%]	SD	NMBE [%]	SD	c.t. [sec]	CV(RMSE) [%]	SD	NMBE [%]	SD	c.t. [sec]	CV(RMSE) [%]	SD	NMBE [%]	SD	c.t. [sec]
Linear	10.85	17.01	2.44	10.03	0.17	6.19	6.24	0.83	2.74	0.17	20.72	28.18	2.49	11.98	0.09
MLP	9.96	10.49	0.04	5.83	85.21	6.46	4.00	1.19	4.32	92.49	16.18	9.34	1.25	9.82	82.08
RF	4.79	1.63	0.02	2.77	74.50	5.19	2.38	0.14	3.49	151.57	9.61	3.41	0.33	5.45	64.95
SVR	8.06	6.78	0.67	5.89	24.51	5.98	3.80	0.02	4.60	13.14	19.99	8.00	−11.12	13.54	2.96

achieves the lowest errors requiring an acceptable average computational time for training (see Table 4).

In the case of relative humidity, as for temperature, the RF model obtained average relative errors, as well as their respective variabilities, lower than those presented by the other models (see Table 4). The average CV(RMSE) yielded by RF is slightly above 5% with a standard deviations of ±2.38, whereas the average CV(RMSE) of the linear regression, MLP and SVR are all around 6% showing a variability of more than ±3. Furthermore, in relation to average NMBE results, all the models except for the MLP neural network present values lower than 1%, with best results produced by the RF and the SVR model, very close to 0 (see Table 4). In this case, the variability in the results of all the models is similar. With regard to the average computational times required for the training, the linear regression and the SVR model are again the fastest (times under 15 s) compared to the times needed for MLP and RF (approximately 92 and 152 s, respectively). Considering all of the above results together, the efficiency of the SVR model and the RF model is higher than the rest, and similar between them. Whereas RF shows, on average, better CV(RMSE) values, SVR shows a better average NMBE in addition to a significant lower computation time (see Table 4).

With respect to CO_2 concentration, the RF model is the only one that yielded an average CV(RMSE) lower than 10%. The others, being the linear regression which perform the worst, show an average CV(RMSE) higher than 15% (see Table 4). In addition, observing the variability in the CV(RMSE) results, the RF model also presents the lowest standard deviation (±3.41). Regarding average NMBE results, RF is again the only model that obtained an average value below 1% (0.33%). Linear regression and MLP neural network show average values of 2.49% and 1.25% respectively, while the average NMBE of the SVR model rises to −11.12%. As in CV(RMSE) results, there is less variability in the results obtained by the RF model, which has a standard deviation of ±5.45. Additionally, analysing the average computational times required by each model, it is observed that the linear regression is the fastest method (less than 1 s) and MLP neural network is the lowest (requiring approximately 82 s). In spite of this, the model that proves to be the most efficient is the RF model due to its lower relative errors and the fact that its computing time needed to train, on average, is reasonable (slightly above one minute, see Table 4).

Moreover, Table 4 shows that the efficiency of the built models varies among the studied variables. The main reasons are, on the one hand, their different daily behaviour and, on the other hand, the different quality of the sensors used to monitor each of the variables (see Fig. 14). While for temperature and relative humidity the overall performance is similar, for CO_2 concentration the performance of the models is worse. Observing the NMBE results it can be seen that, on average, the estimations given by the models are below the real values for all the variables analysed due to their positive values (except for the interpolations of the CO2 levels given by the SVR model). Lastly, according to the average relative errors obtained for each of the variables, together with the average computational time required in the training process, it is demonstrated that RF model is the most efficient model among the analysed ones.

On the other hand, Fig. 14 shows the performance of RF interpolating temperature, relative humidity and CO_2 concentration over two specific days. These days were selected as they represent different behaviour in the analysed variables. Thus, the accuracy of the model estimating in different real situations is tested. In this case, the trained model has information of the positions where the interpolations are made due to the mobile device is at the same position several days. Thus, in training sample there are values collected in the same positions as the test sample. For this reason the relative errors presented in Fig. 14 are lower than the ones shown in Table 4.

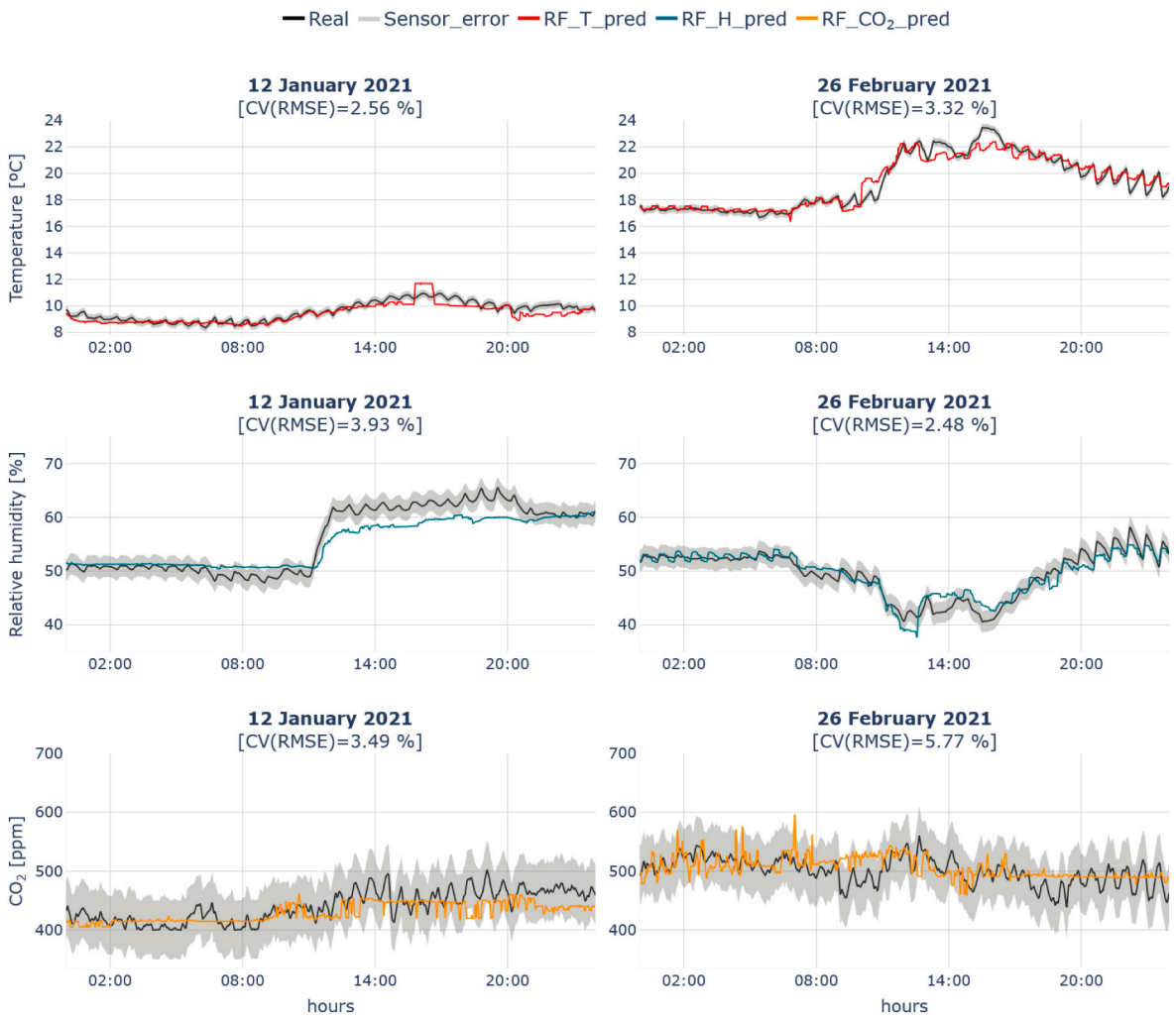


Fig. 14. Results of the interpolations made with the RF model in specific days and, thus, in positions known by the mobile device, together with the errors obtained. Indoor temperatures are presented in the first row, relative humidity in the second and CO_2 concentration in the third.

It can be seen that, in general, the RF model is capable of replicating, in different situations, the behaviour of the three variables efficiently. As in the positional cross-validation, the model built for temperature estimations performs better and the model used to estimate CO_2 values is the one that obtains the worst results. Fig. 14 shows that the built RF models interpolate temperatures with a CV(RMSE) below 3.50%, relative humidity values with a CV(RMSE) below 4% and CO_2 levels with a CV(RMSE) below 6%. The built models based on RF algorithm demonstrates its efficiency interpolating data of the three variables studied.

In summary, the results of both positional cross-validation and accuracy analysis on specific days demonstrate the efficiency of the RF algorithm interpolating temperature, relative humidity and CO_2 concentration. Specifically, the error values obtained by the built models, considering the ability to tolerate up to three malfunctioning wall-mounted devices, are in the same range or lower than the values presented in similar studies [16,21,67,68]. These studies, in which machine learning or simulation models are also used, yield CV(RMSE) values of approximately 6% for temperature, 4% for relative humidity and 10% for CO_2 concentration. Moreover, it is also shown that this type of models only requires a few minutes to be trained with data. Thus, the models used prove the usefulness of the built monitoring system and its ability to interpolate data both spatially and temporally in a large open space.

5. Conclusions

A low-cost system focused on automatically monitoring, instantly displaying the collected values, combined with machine learning models, which interpolate data of specific IEQ-related variables throughout a building, is presented in this paper. The study was conducted for approximately 4 months collecting data of temperature, relative humidity and CO_2 concentration inside the CINTECX building in north-western Spain. The system is based on 8 wall-mounted devices and a mobile device, which in this case was placed in 26 different positions. These devices are connected to a remote platform with storage and visualisation capabilities by means of message brokers to handle data transfer, a time series database to store sensor and status information and an analytics and monitoring front-end for data aggregation and charting. Furthermore, the monitored data, with minute frequency, are used to train different machine learning models with the aim of interpolating the values of the analysed variables. Thus, the quality and the usefulness of the data collected by the presented system is demonstrated. After a comparison between the different models, the RF algorithm is selected as the most efficient based on the average relative errors and the average computational time required to be trained. Specifically, a different RF model is trained to interpolate data for each of the variables analysed. To this end, the values collected by the mobile device are mapped to the values collected by the fixed devices, along with their distances to it. In addition, meteorological data of the area

and three temporal variables (hour of the day, day of the week and day of the year) are also considered as model inputs to improve the training process.

The results obtained by the built RF models show that, based on the data monitored by the presented system, it is possible to efficiently estimate the indoor environmental conditions in a building. On the one hand, a positional cross-validation is performed to demonstrate the capability of the trained models to spatially interpolate data. In the case of temperatures, the built RF model yields an average CV(RMSE) of 4.79% and an average NMBE of 0.02%. For relative humidity values, the average CV(RMSE) and the average NMBE obtained by the model are 5.19% and 0.14%, respectively. Lastly, with regards to CO_2 levels, the built RF model yields an average CV(RMSE) of 9.61% and a average NMBE of 0.33%. On the other hand, through a temporal analysis focused on specific days, it is also proved that the estimations produced by the trained models effectively replicate the behaviour of the selected variables. In this case the interpolations achieved a CV(RMSE) lower than 3.5% for temperature, lower than 4% for relative humidity and lower than 6% for CO_2 levels. The fact that these values are within the same range or lower than the error values obtained in similar studies demonstrate the quality of the monitored data. In addition, once the RF model is trained, no human intervention is required to collect the data that the model needs to perform the interpolations.

The main contribution of this paper is the design and development of an intelligent monitoring system that collects values of specific variables related to the IEQ. This platform, connected with a trained machine learning models, contribute to improving the control of the indoor environmental conditions throughout a space in a building. In this way, the detection of low comfort areas or where the users' health is at risk is possible. Moreover, this methodology is also useful to analyse possible deficiencies in the building envelope or in the heating and cooling system. The machine learning models are trained with the collected data so that the inputs required by the models are provided without relying on a mobile device that is constantly traversing the space and without the need to actually measure inside the occupied area. The main limitation of this research is the duration of the study and, although the size of the data sample is sufficient to present a consistent analysis, a larger sample would allow a more complete analysis to be carried out. On the one hand, the mobile device could be in more positions throughout the building and, on the other hand, data from different seasons of the year could be used. In addition, the fact that the analysed space is a large multizone open space makes the system, together with the interpolation methodology, applicable to similar buildings and, considering that the difficulty of the problem is reduced, also to smaller buildings. As future lines of research, other variables related to the IEQ of a building or an index summarising the indoor comfort, other combinations of fixed devices (positions, number of sensors, etc.) or even more accurate sensors could be considered to attempt a more complete analysis. Ultimately, alert systems that detect low comfort areas can be built to efficiently control the IEQ of a building in an automated way.

Acknowledgments and funding

This research was partially supported by the Ministry of Science, Innovation and Universities of Spanish Government under the SMARTHERM project (RTI2018-096296-B-C2). The authors also want to thank the Ministry of Science, Innovation and Universities (grants FPU17/01834 and FPU19/01187) and the University of Vigo (grant 00VI 131H 641.02). The authors thank the Defense University Center at the Spanish Naval Academy (CUD-ENM) for all the support provided for this research. Funding for open access charge: Universidade de Vigo/CISUG, Spain.

Declaration of Competing Interest

The authors declare no conflicts of interest.

References

[1] C. Roberts, D. Edwards, M.R. Hosseini, M. Mateo-Garcia, D.-G. Owusu-Manu, Post-occupancy evaluation: a review of literature, Eng. Constr. Archit. Manag. 26 (06) (2019), https://doi.org/10.1108/ECAM-09-2018-0390.

[2] H. Alavi, N. Forcada, R. Bortolini, D.J. Edwards, Enhancing occupants' comfort through bim-based probabilistic approach, Autom. Constr. 123 (2021) 103528, https://doi.org/10.1016/j.autcon.2020.103528. URL, https://www.sciencedirect.com/science/article/pii/S0926580520311080.

[3] A. Nawawi, N. Khalil, Post-occupancy evaluation correlated with building occupants' satisfaction: an approach to performance evaluation of government and public buildings, J. Build. Apprais. 4 (2008) 59–69, https://doi.org/10.1057/jba.2008.22.

[4] S. Wang, C. Yan, F. Xiao, Quantitative energy performance assessment methods for existing buildings, Energy Build. 55 (2012) 873–888, cool Roofs, Cool Pavements, Cool Cities, and Cool World, https://doi.org/10.1016/j.enbuild.2012.08.037. URL, https://www.sciencedirect.com/science/article/pii/S0378778812004410.

[5] R. Elnaklah, I. Walker, S. Natarajan, Moving to a green building: indoor environment quality, thermal comfort and health, Build. Environ. 191 (2021) 107592, https://doi.org/10.1016/j.buildenv.2021.107592. URL, https://www.sciencedirect.com/science/article/pii/S0360132321000081.

[6] J. Saini, M. Dutta, G. Marques, Sensors for indoor air quality monitoring and assessment through internet of things: a systematic review, Environ. Monit. Assess. 193 (2) (2021), https://doi.org/10.1007/s10661-020-08781-6 cited By 0. URL, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85100111018&doi=10.1007%2fs10661-020-08781-6&partnerID=40&md5 [?tjl]?>equals;064a70e71148100a257be908da78bebc.

[7] L. Bellia, F.R. d'Ambrosio Alfano, F. Fragliasso, B.I. Palella, G. Riccio, On the interaction between lighting and thermal comfort: an integrated approach to ieq, Energy Build. 231 (2021), 110570, https://doi.org/10.1016/j.enbuild.2020.110570. URL, https://www.sciencedirect.com/science/article/pii/S0378778820333569.

[8] J. López Gómez, F. Troncoso Pastoriza, E.A. Fariña, P. Eguía Oller, E. Granada Álvarez, Use of a numerical weather prediction model as a meteorological source for the estimation of heating demand in building thermal simulations, Sustain. Cities Soc. 62 (2020), 102403, https://doi.org/10.1016/j.scs.2020.102403. URL, https://www.sciencedirect.com/science/article/pii/S2210670720306247.

[9] I. Demanega, I. Mujan, B.C. Singer, A.S. Anđelković, F. Babich, D. Licina, Performance assessment of low-cost environmental monitors and single sensors under variable indoor air quality and thermal conditions, Build. Environ. 187 (2021), 107415, https://doi.org/10.1016/j.buildenv.2020.107415. URL, https://www.sciencedirect.com/science/article/pii/S0360132320307836.

[10] H. Chojer, P. Branco, F. Martins, M. Alvim-Ferraz, S. Sousa, Development of low-cost indoor air quality monitoring devices: recent advancements, Sci. Total Environ. 727 (2020) 138385, https://doi.org/10.1016/j.scitotenv.2020.138385. URL, https://www.sciencedirect.com/science/article/pii/S0048969720318982.

[11] M. Jin, S. Liu, S. Schiavon, C. Spanos, Automated mobile sensing: towards high-granularity agile indoor environmental quality monitoring, Build. Environ. 127 (2018) 268–276, https://doi.org/10.1016/j.buildenv.2017.11.003. URL, https://www.sciencedirect.com/science/article/pii/S0360132317305012.

[12] Y. Yang, J. Liu, W. Wang, Y. Cao, H. Li, Incorporating slam and mobile sensing for indoor co2 monitoring and source position estimation, J. Clean. Prod. 291 (2021) 125780, https://doi.org/10.1016/j.jclepro.2020.125780. URL, https://www.sciencedirect.com/science/article/pii/S0959652620358261.

[13] H. Shen, W. Hou, Y. Zhu, S. Zheng, S. Ainiwaer, G. Shen, Y. Chen, H. Cheng, J. Hu, Y. Wan, S. Tao, Temporal and spatial variation of pm2.5 in indoor air monitored by low-cost sensors, Sci. Total Environ. 770 (2021) 145304, https://doi.org/10.1016/j.scitotenv.2021.145304. URL, https://www.sciencedirect.com/science/article/pii/S0048969721003703.

[14] M.F. Antwi-Afari, H. Li, Y. Yu, L. Kong, Wearable insole pressure system for automated detection and classification of awkward working postures in construction workers, Autom. Constr. 96 (2018) 433–441, https://doi.org/10.1016/j.autcon.2018.10.004. URL, https://www.sciencedirect.com/science/article/pii/S0926580518303819.

[15] W. Yi, A.P. Chan, X. Wang, J. Wang, Development of an early-warning system for site work in hot and humid environments: a case study, Autom. Constr. 62 (2016) 101–113, https://doi.org/10.1016/j.autcon.2015.11.003. URL, https://www.sciencedirect.com/science/article/pii/S0926580515002289.

[16] M. Martínez-Comesaña, A. Ogando-Martínez, F. Troncoso-Pastoriza, J. López-Gómez, L. Febrero-Garrido, E. Granada-Álvarez, Use of optimised mlp neural networks for spatiotemporal estimation of indoor environmental conditions of existing buildings, Build. Environ. 205 (2021), 108243, https://doi.org/10.1016/j.buildenv.2021.108243.

[17] J.M. Helm, A.M. Swiergosz, H.S. Haeberle, J.M. Karnuta, J.L. Schaffer, V.E. Krebs, A.I. Spitzer, P.N. Ramkumar, Machine learning and artificial intelligence: definitions, applications, and future directions, Curr. Rev. Musculoskel. Med. 13 (1) (2020) 69–76, https://doi.org/10.1007/s12178-020-09600-8.

[18] M. Martínez-Comesaña, L. Febrero-Garrido, E. Granada-Álvarez, J. Martínez-Torres, S. Martínez-Mariño, Heat loss coefficient estimation applied to existing buildings through machine learning models, Appl. Sci. 10 (24) (2020), https://doi.org/10.3390/app10248968. URL, https://www.mdpi.com/2076-3417/10/24/8968.

[19] M. Martínez Comesaña, L. Febrero-Garrido, F. Troncoso-Pastoriza, J. Martínez-Torres, Prediction of building's thermal performance using lstm and mlp neural

networks, Appl. Sci. 10 (21) (2020), https://doi.org/10.3390/app10217439. URL, https://www.mdpi.com/2076-3417/10/21/7439.
[20] D. Li, F. Huang, L. Yan, Z. Cao, J. Chen, Z. Ye, Landslide susceptibility prediction using particle-swarm-optimized multilayer perceptron: comparisons with multilayer-perceptron-only, bp neural network, and information value models, Appl. Sci. 9 (18) (2019), https://doi.org/10.3390/app9183664. URL, https://www.mdpi.com/2076-3417/9/18/3664.
[21] Z. Yu, Y. Song, D. Song, Y. Liu, Spatial interpolation-based analysis method targeting visualization of the indoor thermal environment, Build. Environ. 188 (2021) 107484, https://doi.org/10.1016/j.buildenv.2020.107484. URL, https://www.sciencedirect.com/science/article/pii/S0360132320308519.
[22] J.H. Cho, Detection of smoking in indoor environment using machine learning, Appl. Sci. 10 (24) (2020), https://doi.org/10.3390/app10248912. URL, https://www.mdpi.com/2076-3417/10/24/8912.
[23] H.-S. Jo, C. Park, E. Lee, H.K. Choi, J. Park, Path loss prediction based on machine learning techniques: principal component analysis, artificial neural network, and gaussian process, Sensors 20 (7) (2020), https://doi.org/10.3390/s20071927. URL, https://www.mdpi.com/1424-8220/20/7/1927.
[24] S. Mishra, H.K. Tripathy, P.K. Mallick, A.K. Bhoi, P. Barsocchi, Eaga-mlp—an enhanced and adaptive hybrid classification model for diabetes diagnosis, Sensors 20 (14) (2020), https://doi.org/10.3390/s20144036. URL, https://www.mdpi.com/1424-8220/20/14/4036.
[25] K.A. Suaza Cano, J.F. Moofarry, J.F. Castillo Garcia, Proposal for the implementation of mlp neural networks on arduino platform, in: M. Botto-Tobar, M. Zambrano Vizuete, P. Torres-Carrión, S. Montes León, G. Pizarro Vásquez, B. Durakovic (Eds.), Applied Technologies, Springer International Publishing, Cham, 2020, pp. 372–385, https://doi.org/10.1007/978-3-030-42520-3_30.
[26] N. Eiamkanitchat, N. Kuntekul, P. Panyaphruek, Ensemble mlp networks for voices command classification to control model car via piface interface of raspberry pi, Int. J. Geomate 13 (2017) 9–15, https://doi.org/10.21660/2017.37.2817.
[27] M.W. Ahmad, M. Mourshed, Y. Rezgui, Trees vs neurons: comparison between random forest and ann for high-resolution prediction of building energy consumption, Energy Build. 147 (2017) 77–89, https://doi.org/10.1016/j.enbuild.2017.04.038. URL, http://www.sciencedirect.com/science/article/pii/S0378778816313937.
[28] P. Moore, T. Lyons, J. Gallacher, Random forest prediction of alzheimer's disease using pairwise selection from time series data, PLoS One 14 (2019), e0211558, https://doi.org/10.1371/journal.pone.0211558.
[29] B. Manavalan, T.H. Shin, M. Kim, G. Lee, Aippred: sequence-based prediction of anti-inflammatory peptides using random forest, Front. Pharmacol. 9 (2018), https://doi.org/10.3389/fphar.2018.00276.
[30] C.-L. Wei, G.T. Rowe, E. Escobar-Briones, A. Boetius, T. Soltwedel, M.J. Caley, Y. Soliman, F. Huettmann, F. Qu, Z. Yu, C.R. Pitcher, R.L. Haedrich, M.K. Wicksten, M.A. Rex, J.G. Baguley, J. Sharma, R. Danovaro, I.R. MacDonald, C.C. Nunnally, J. W. Deming, P. Montagna, M. Lévesque, J.M. Weslawski, M. Wlodarska-Kowalczuk, B.S. Ingole, B.J. Bett, D.S.M. Billett, A. Yool, B.A. Bluhm, K. Iken, B. E. Narayanaswamy, Global patterns and predictions of seafloor biomass using random forests, PLoS One 5 (12) (2011) 1–15, https://doi.org/10.1371/journal.pone.0015323.
[31] I.A. Ibrahim, T. Khatib, A novel hybrid model for hourly global solar radiation prediction using random forests technique and firefly algorithm, Energy Convers. Manag. 138 (2017) 413–425, https://doi.org/10.1016/j.enconman.2017.02.006. URL, http://www.sciencedirect.com/science/article/pii/S0196890417301036.
[32] W.M. Shao, X.Z. Liu, Research on drive control method of induction motor based on random forest, in: 2020 IEEE International Conference on Industrial Application of Artificial Intelligence (IAAI), 2020, pp. 444–450, https://doi.org/10.1109/IAAI51705.2020.9332861.
[33] K. Kirutika, V. Vetriselvi, R. Parthasarathi, G.S.V. Rao, Controller monitoring system in software defined networks using random forest algorithm, in: 2019 International Carnahan Conference on Security Technology (ICCST), 2019, pp. 1–6, https://doi.org/10.1109/CCST.2019.8888369.
[34] J. Wei, G. Dong, Z. Chen, Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression, IEEE Trans. Ind. Electron. 65 (7) (2018) 5634–5643, https://doi.org/10.1109/TIE.2017.2782224.
[35] M.W. Ahmad, M. Mourshed, Y. Rezgui, Tree-based ensemble methods for predicting pv power generation and their comparison with support vector regression, Energy 164 (2018) 465–474, https://doi.org/10.1016/j.energy.2018.08.207. URL, http://www.sciencedirect.com/science/article/pii/S0360544218317432.
[36] J. Fan, X. Wang, L. Wu, H. Zhou, F. Zhang, X. Yu, X. Lu, Y. Xiang, Comparison of support vector machine and extreme gradient boosting for predicting daily global solar radiation using temperature and precipitation in humid subtropical climates: a case study in China, Energy Convers. Manag. 164 (2018) 102–111, https://doi.org/10.1016/j.enconman.2018.02.087.
[37] H. Zhong, J. Wang, H. Jia, Y. Mu, S. Lv, Vector field-based support vector regression for building energy consumption prediction, Appl. Energy 242 (2019) 403–414, https://doi.org/10.1016/j.apenergy.2019.03.078. URL, http://www.sciencedirect.com/science/article/pii/S0306261919304878.
[38] D. Wu, C. Jennings, J. Terpenny, R.X. Gao, S. Kumara, A comparative study on machine learning algorithms for smart manufacturing: tool wear prediction using random forests, J. Manuf. Sci. Eng. 139 (7) (04 2017), 071018. arXiv:.
[39] B.M. Henrique, V.A. Sobreiro, H. Kimura, Stock price prediction using support vector regression on daily and up to the minute prices, J. Finance Data Sci. 4 (3) (2018) 183–201, https://doi.org/10.1016/j.jfds.2018.04.003. URL, http://www.sciencedirect.com/science/article/pii/S2405918818300060.
[40] P. Vrablecová, A. Bou Ezzeddine, V. Rozinajová, S. Šárik, A.K. Sangaiah, Smart grid load forecasting using online support vector regression, Comput. Electr. Eng. 65 (2018) 102–117, https://doi.org/10.1016/j.compeleceng.2017.07.006. URL, http://www.sciencedirect.com/science/article/pii/S0045790617320645.
[41] M.S. Shahriar, M. Shafiullah, M.J. Rana, Stability enhancement of pss-upfc installed power system by support vector regression, Electr. Eng. 100 (3) (2018) 1601–1612. URL, https://doi.org/10.1007/s00202-017-0638-8.
[42] M. Martínez-Comesaña, P. Eguía-Oller, J. Martínez-Torres, L. Febrero-Garrido, E. Granada-Álvarez, Optimisation of thermal comfort and indoor air quality estimations applied to in-use buildings combining nsga-iii and xgboost, Sustain. Cities Soc. 80 (2022) 103723, https://doi.org/10.1016/j.scs.2022.103723. URL, https://www.sciencedirect.com/science/article/pii/S2210670722000531.
[43] C. Boje, A. Guerriero, S. Kubicki, Y. Rezgui, Towards a semantic construction digital twin: directions for future research, Autom. Constr. 114 (2020) 103179, https://doi.org/10.1016/j.autcon.2020.103179. URL, https://www.sciencedirect.com/science/article/pii/S0926580519314785.
[44] OASIS, Advanced message queuing protocol (amqp) version 1.0 (accessed on 2 March 2022). URL, http://docs.oasis-open.org/amqp/core/v1.0/amqp-core-complete-v1.0.pdf, 2012.
[45] J. Shahid, Influxdb Documentation, 2019.
[46] A. Rahimi, Short-term prediction of no2 and nox concentrations using multilayer perceptron neural network: a case study of Tabriz, Iran, Ecol. Process. 6 (1) (2017) 4, https://doi.org/10.1186/s13717-016-0069-x.
[47] M. Kahani, M.H. Ahmadi, A. Tatar, M. Sadeghzadeh, Development of multilayer perceptron artificial neural network (mlp-ann) and least square support vector machine (lssvm) models to predict nusselt number and pressure drop of tio2/water nanofluid flows through non-straight pathways, Num. Heat Transfer, A: Applications 74 (4) (2018) 1190–1206, arXiv: https://doi.org/10.1080/10407782.2018.1523597.
[48] W. Jiang, G. He, T. Long, Y. Ni, H. Liu, Y. Peng, K. Lv, G. Wang, Multilayer perceptron neural network for surface water extraction in landsat 8 oli satellite images, Remote Sens. 10 (5) (2018), https://doi.org/10.3390/rs10050755. URL, https://www.mdpi.com/2072-4292/10/5/755.
[49] B.T. Pham, M.D. Nguyen, K.-T.T. Bui, I. Prakash, K. Chapi, D.T. Bui, A novel artificial intelligence approach based on multi-layer perceptron neural network and biogeography-based optimization for predicting coefficient of consolidation of soil, CATENA 173 (2019) 302–311, https://doi.org/10.1016/j.catena.2018.10.004. URL, http://www.sciencedirect.com/science/article/pii/S0341816218304314.
[50] M.A. Janati Idrissi, H. Ramchoun, Y. Ghanou, M. Ettaouil, Genetic Algorithm for Neural Network Architecture Optimization, 2016, pp. 1–4, https://doi.org/10.1109/GOL.2016.7731699.
[51] D.S. Yeung, J. Li, W.W.Y. Ng, P.P.K. Chan, Mlpnn training via a multiobjective optimization of training error and stochastic sensitivity, IEEE Trans. Neural Networks Learn. Syst. 27 (5) (2016) 978–992, https://doi.org/10.1109/TNNLS.2015.2431251.
[52] J. Rynkiewicz, Asymptotic statistics for multilayer perceptron with relu hidden units, Neurocomputing 342 (2019) 16–23, https://doi.org/10.1016/j.neucom.2018.11.097, advances in artificial neural networks, machine learning and computational intelligence URL, http://www.sciencedirect.com/science/article/pii/S0925231219301547.
[53] W. Sun, C. Huang, A carbon price prediction model based on secondary decomposition algorithm and optimized back propagation neural network, J. Clean. Prod. 243 (2020) 118671, https://doi.org/10.1016/j.jclepro.2019.118671. URL, http://www.sciencedirect.com/science/article/pii/S0959652619335413.
[54] E. Guresen, G. Kayakutlu, T.U. Daim, Using artificial neural network models in stock market index prediction, Expert Syst. Appl. 38 (8) (2011) 10389–10397, https://doi.org/10.1016/j.eswa.2011.02.068. URL, http://www.sciencedirect.com/science/article/pii/S0957417411002740.
[55] C.-J. Huang, P.-H. Kuo, A short-term wind speed forecasting model by using artificial neural networks with stochastic optimization for renewable energy systems, Energies 11 (10) (2018), https://doi.org/10.3390/en11102777. URL, https://www.mdpi.com/1996-1073/11/10/2777.
[56] M.W. Khan, M. Zeeshan, M. Usman, Traffic scheduling optimization in cognitive radio based smart grid network using mini-batch gradient descent method, in: 2019 14th Iberian Conference on Information Systems and Technologies (CISTI), 2019, pp. 1–5, https://doi.org/10.23919/CISTI.2019.8760693.
[57] J.H. Jeong, J.P. Resop, N.D. Mueller, D.H. Fleisher, K. Yun, E.E. Butler, D.J. Timlin, K.-M. Shim, J.S. Gerber, V.R. Reddy, S.-H. Kim, Random forests for global and regional crop yield predictions, PLoS One 11 (6) (2016) 1–15, https://doi.org/10.1371/journal.pone.0156571.
[58] R. Rahman, S.R. Dhruba, S. Ghosh, R. Pal, Functional random forest with applications in dose-response predictions, Sci. Rep. 9 (2019) 1628, https://doi.org/10.1038/s41598-018-38231-w.
[59] K. Khan, E. Ratner, R. Ludwig, A. Lendasse, Feature bagging and extreme learning machines: machine learning with severe memory constraints, in: 2020 International Joint Conference on Neural Networks (IJCNN), 2020, pp. 1–7, https://doi.org/10.1109/IJCNN48605.2020.9207673.
[60] Erwan Scornet, Tuning parameters in random forests, ESAIM: Procs 60 (2017) 144–162, https://doi.org/10.1051/proc/201760144.
[61] Z. Tan, J. Zhang, Y. He, Y. Zhang, G. Xiong, Y. Liu, Short-term load forecasting based on integration of svr and stacking, IEEE Access 8 (2020) 227719–227728, https://doi.org/10.1109/ACCESS.2020.3041779.
[62] X. de Galicia, Meteogalicia (accesed on 10 March 2021). URL, https://www.meteogalicia.gal/web/RSS/rssIndex.action, Mar. 2021.

[63] NOAA, NOAA operational model archive and distribution system (nomads) (accessed on 10 March 2021). URL, https://nomads.ncep.noaa.gov/, Mar. 2021.
[64] NOAA, The global forecast system (gfs) - global spectral model (gsm) (accessed on 10 March 2021). URL, https://www.emc.ncep.noaa.gov/emc/pages/numerical_forecast_systems/gfs/documentation.php, 2021.
[65] S. Martínez, P. Eguía, E. Granada, A. Moazami, M. Hamdy, A performance comparison of multi-objective optimization-based approaches for calibrating white-box building energy models, Energy Build. 216 (2020) 109942, https://doi.org/10.1016/j.enbuild.2020.109942. URL, http://www.sciencedirect.com/science/article/pii/S0378778819336850.
[66] G. Van Rossum, F.L. Drake, Python 3 Reference Manual, CreateSpace, Scotts Valley, CA, 2009, 978-1-4414-1269-0.
[67] A. Liguori, R. Markovic, T. Dam, J. Frisch, C. Treeck, F. Causone, Indoor environment data time-series reconstruction using autoencoder neural networks, Build. Environ. 191 (2021), 107623, https://doi.org/10.1016/j.buildenv.2021.107623.
[68] A. Pantazaras, S.E. Lee, M. Santamouris, J. Yang, Predicting the co2 levels in buildings using deterministic and identified models, Energy Build. 127 (2016) 774–785, https://doi.org/10.1016/j.enbuild.2016.06.029. URL, https://www.sciencedirect.com/science/article/pii/S0378778816305187.