Effectiveness of neural networks and transfer learning for indoor air-temperature forecasting

Effectiveness of Neural Networks and Transfer Learning for indoor air-temperature forecasting

Andrea Bellagarda[a], Silvia Cesari[a], Alessandro Aliberti[a,*], Francesca Ugliotti[d], Lorenzo Bottaccioli[c,b], Enrico Macii[c], Edoardo Patti[a,b]

[a]*Dept. of Control and Computer Engineering,*
[b]*Energy Center Lab,*
[c]*Interuniversity Dept. of Regional and Urban Studies and Planning,*
[d]*Department of Structural, Geotechnical and Building Engineering,*
Politecnico di Torino, 10129 Torino, Italy

Abstract

Starting in 2007, EU set energy efficiency improvement targets in sectors with high energy-saving potential such as buildings. ICT allows innovative opportunities for energy consumption forecast to integrate with new control policies such as Demand/Response and Demand Side Management to reduce energy waste. However, such technologies must overcome challenges such as the lack of accurate historic data required for predictions.

This article proposes an innovative methodology supporting the energy management of HVAC systems, through Smart Building indoor air-temperature forecast. The applicability of innovative neural networks for time-series predictions is explored. These neural networks are first trained on an artificial but realistic dataset based on BIM simulations with real meteorological data. The inference phase is then carried out on a second dataset collected by IoT devices. Finally, Transfer Learning techniques are exploited to improve the performances predictions. Fanger's model is applied to validate results, showing consistent levels of accuracy and comfort.

Keywords: Transfer Learning, Artificial Neural Networks, Indoor Air-Temperature Forecasting, Smart Building, Energy Efficiency, Demand

*Corresponding author
Email address: alessnadro.aliberti@polito.it (Alessandro Aliberti)

Side Management

2010 MSC: 00-01, 99-00

---

1. Introduction

In the European Union (EU), more than 40% of the overall energy consumption and about 36% of total pollution is attributable to buildings, as reported by the European Union Directive on the *Energy Performance of Buildings* [1]. Furthermore, with this sector undergoing a constant expansion, its energy consumption and CO_2 emissions are also consistently rising [1]. The need to reduce this energy consumption is therefore mandatory, in order to both reduce the consistent increase in request for electricity and also the impact this sector has on climate change. As a result, research has been increasingly driven towards the design and implementation of buildings (either already existing or new ones) even more energy-efficient, moving forward to the Smart Building view [2].

With a large number of buildings which continue to use centralized heating systems with out-of-date control strategies, this sector still has significant room for improvement in energy management [3]. The Heating, Ventilation and Air Conditioning (HVAC) systems are responsible for almost half of the total energy consumption in buildings. The functioning of these systems is driven mainly by the indoor air-temperature, with the cooling system being acted upon in the summer and the heating system in the winter. The availability of a good indoor air-temperature prediction system, capable of effectively considering all variables affecting it, such as building occupancy, environmental conditions and season change, can give an important contribution towards smart energy management applications for building HVAC systems, which in turn can improve the building's energy efficiency. For example, effective indoor air-temperature prediction systems can allow to forecast in advance the energy required and cost and it can be used as input for load forecasting problems [4].

The benefits of such applications can further be expanded when applied to buildings supplied by a district heating system, since then can allow a bal-

ancing of the energy demand and a more efficient planning of energy supply. Within this framework, applications such as Demand/Response (DR) [5] and Demand Side Management (DSM) [6], which also take ambient comfort into consideration [7], can therefore include the building heating systems. Furthermore, thermal energy peaks can be reduced and thermal energy consumption can be reshaped, in case of district heating, by predicting energy profiles [8]. As analyzed by Bianchini et al. [9], HVAC is a particularly important actor in the Demand Response (DR) program, which consists of a set of mechanisms aiming to modify the consumer's demand energy profile over time in order to reduce the demand at peak times and match the overall energy demand with supply capacity. As a consequence, DR also allows to reduce operating costs from expensive generators in favor or Distributed Energy Resources, which include renewable energy sources, therefore also achieving a reduction in CO_2 emissions [10].

1.1. Building and thermal modelling

In order to control an HVAC system, a suitable model able to accurately predict building response to operational and environmental changes must be developed. The recent literature provides many methodologies to model buildings [11, 12, 13]. For example, white-box building models can be exploited in the building field. Indeed, 3-dimensional representations of the buildings can be created, including all those variables affecting building thermal response, such as materials and insulation of walls, roof and windows, windows size, ventilation scheduling, exposure to sun and so on.

Once the model has been developed, simulation software can be used to predict the building thermal behaviour. Among all, EnergyPlus [14] and TRNSYS [15] represent a milestone simulation tools for thermal energy performance in buildings in both design and refurbishment phases. In particular, EnergyPlus is a building energy simulation program for modeling building heating, cooling, lighting, ventilating, and other energy flows [16], its development is funded by the U.S. Department of Energy's (DOE) [17] and it is a building simulation program commonly used by researchers and designers [18].

However, even if these software provide very robust and accurate results they are extremely demanding, in terms of computational resources [19]. To overcome such limitations, literature provides new methodologies able at combining computational costs and thermal estimations accuracy [20]. Such methodologies start from an accurate model of the building and then obtain a more compact approximated representation via i) model order reduction, ii) model aggregation or iii) ad-hoc dynamics extraction [21, 22].

Alternatively, Bacher et al. propose another modelling approach, where the building is represented as a grey-box model [23]. In detail, the model is built using the resistor–capacitor representation of the heating dynamic of the building, meaning that its components are represented by resistances and capacitors. These kinds of simulations however require many assumptions regarding both their variables, such as ventilation scheduling, and on the calibration of these variables, and thus are seldom truly representative of the real environment. Further challenges include correctly taking into account surrounding buildings, heat sources and obstacles to air movement [24], together with the high computational cost of Computational Fluid Dynamics simulations.
For these reasons, neural networks are nowadays widely used for the modelling of building thermal response, including the prediction of indoor air-temperature, and are reaching promising results.

Other approaches focus on Very Large Scale Integration (VLSI) techniques to build a compact thermal model. Solutions based on this approach often exploit matrix pencil [25] and subspace identification [26] that do not consider physical restrictions and hence make the compact model very flexible. VLSI-based solutions take advantage of a detailed analysis of numerical simulations or real-world sampled data, making the training phase of the whole model very accurate. However, they are generally not suitable to deal with the non-linearity of a whole building thermal system. Again, this issue can be efficiently addressed by machine learning techniques, such as Artificial Neural Networks (ANNs).

1.2. Black-box modelling with Artificial Neural Networks

Neural networks are data-driven methods capable of modelling the underlying pattern of a system by leveraging only specific inputs and outputs. In this context, they can therefore be used to model and predict the indoor air-temperature with only previous indoor air-temperature measurements as inputs. As a consequence, neural networks require significantly less data compared to traditional modelling and simulation methods, which require a wide variety of different parameters, such as building and environmental parameters, in order to effectively model and simulate indoor air-temperature behaviour. Furthermore, another significant advantage of using neural networks in this context is their predisposition for transferability. As a consequence, the use of artificial neural networks in various applications related with energy management in buildings has been increasing significantly over the recent years [27].

Mateo et al. [28], in their investigation of indoor air-temperature prediction problems, compar classical machine learning models for time-series forecasting, such as Autoregressive, Multiple Linear Regression (MLR) and Robust MLR, to neural networks models, including Multilayer Perceptron with Non-linear Autoregressive Exogenous (MLP-NARX) and Extreme Learning Machine. All models are developed by using one year of simulated data generated by the TeKton 3D software. The results show that neural models outperform the other machine learning models, with the MLP-NARX neural network showing particularly positive performance for indoor air-temperature prediction.

This neural network is then further investigated in a research conducted by Aliberti et al. [29], introducing the Non-linear Autoregressive neural network (NAR) architecture able to base its prediction on a high number of regressors, highlighting an overall good performance in predicting indoor air-temperature values up to two hours. The research however leaves the opportunity to investigate the possibility of extending the forecasting horizon, in order to increase the applicability of the methodology for Smart Energy Management in buildings. Xu et al. [30] first investigate Long Short Term Memory (LSTM) neural networks to predict indoor air-temperature in a public building using data collected

by sensors, sampled each 5 minutes. The performances of the LSTM neural network is evaluated for two time horizons, 5 minutes (1 step ahead prediction) and 30 minutes (6 steps ahead prediction), and then compared with performances of Back Propagation Neural Network, Support Vector Machine and Decision Tree models. The results show that the LSTM neural network slightly outperforms all other networks in both cases, with the authors suggesting that the positive difference in performance could be even greater but such potential is not fully exploited due to the limited amount of data.

Kamel et al. [31] also developed a LSTM model for indoor air temperature prediction, and to detect the inputs with the highest impact, showing how a limited number of inputs are the most effective in generating an acceptable level of accuracy.

Deihimi et at. [4] also add another interesting input variable to their networks: whether the data is related to a weekend or a holiday, with their results showing greater prediction accuracy when only working days are considered.

Previous studies in this field, such as [32] and [29], were able to develop accurate models for indoor air-temperature prediction with neural networks, but also required a very large amount of data to effectively train such networks. The collection of such a large dataset with acceptable quality levels (clean data, absence of time jumps, absence of false measurements, etc.) is however still complicated, since it requires the application of different technologies and the ability to effectively orchestrate them together. Therefore, even if the research in the various sectors improves day by day and the availability of data is increasing, the application of artificial neural networks for indoor air-temperature prediction and energy management in buildings is still challenged by lack of large and reliable enough data.

Most recently, Cifuentes et al. [33] carried out a study to review different machine learning strategies present in literature for indoor air-temperature forecasting, with their advantages and disadvantages being highlighted and their research gaps being identified. Cifuentes et al. highlight how Deep Learning based approaches, for instance, require the acquisition of long time series or

complementary simulation systems to generate enough samples to perform the training-validation process. In particular to predict indoor air-temperature accurately, training- and test-set must report minimum three years and one year of sampling, respectively. This is also confirmed in [24, 34, 32, 29]. Alawadi et al. [35] also compares 36 Machine Learning (ML) algorithms that could be used to forecast indoor air-temperature in a smart building, highlighting how most of the algorithms such as ANN are highly sensitive to data noise and how increasing the forecasting time does not decrease the accuracy of the best model.

1.3. Black-box model transferability through leverage of Transfer Learning

In recent years, Transfer Learning has been increasingly investigated as a possible solution to overcome the challenge posed by lack of large and reliable enough data in a variety of fields, including Smart Energy Management for buildings. In [36], Jiang and Lee present the application of transfer learning on an LSTM network for Thermal Dynamic Modeling of building indoor temperature evolution and energy consumption. Their investigation shows that the transfer learning approach is effective to adapt the pretrained LSTM model from one building to another building, highlights better performance in prediction compared to the traditional model applied on a limited amount of data and presents LSTM ANNs as a suitable model on which to base such an application. However, their investigation pre-trains the initial model on a large amount of data collected from the initial building, and does not consider the end-user's comfort when evaluating the effectiveness of the model. Gao et al. in [37], instead present a transfer learning based multilayer perceptron model (TL-MLP-C*) to tackle the common data-shortage problem and boost the performance of thermal comfort prediction. Their investigation provides an interesting application of transfer learning on thermal comfort prediction, but goes directly to the prediction of thermal comfort itself rather than of indoor air-temperature, which is considered as one of the features to their ANN model. In [38], Xu et al present a novel transfer learning based approach to overcome the challenge posed by long training time for DRL (and many other data-driven learning methods), and to

effectively transfer a DRL-based HVAC controller trained for a source building to a controller for a target building with minimal effort and improved performance. While presenting an innovative transfer learning approach by taking the neural network which is the basis for the HVAC controller, and breaking-down its design into two sub-networks (a front-end network that can be directly transferred since it is independent from the building it is applied on, and a building-specific back-end network that can be efficiently trained with offline supervise learning), the replicability of this method on indoor air-temperature prediction must be evaluated. Deng and Chen in [39] also leverage transfer learning to transfer the logic and part of the occupant behavior model structure between buildings with different HVAC control systems, and to transfer an RL model from office buildings to residential buildings with a modification to the impact of air temperature on occupant behavior. Again, although their investigation demonstrates the effectiveness of transfer learning in the context of Smart Energy Management for buildings, it is not applied directly on indoor air-temperature prediction.

When looking at indoor air-temperature, Chen et al. in [24] also investigate the prediction of building indoor air-temperature through the application of two transfer learning techniques. They use an EnergyPlus simulation to generate two datasets: a multi-year dataset referred to a room in Beijing, which is then used as source dataset, and a 15-days only dataset referred to a room in Shanghai, used as target dataset. The two rooms have different characteristics. The source dataset is then used to train a Multilayer Perceptron neural network, onto which the following three transfer learning techniques are applied using the target dataset: i) re-train all the neural network, ii) re-train only the last layer and iii) re-train only the first layer. The results of the investigation show that a good prediction on the target dataset is achieved by all three transfer learning techniques, with the third one (re-train only the first layer) however consistently slightly outperforming the other two for all considered prediction horizons: 10, 30, 60 and 90 minutes. By comparing the research carried out in [40] and [24], both investigations use the technique named "soft-start" in [40], where all the

neural network is re-trained, while [24] explores the two other techniques, where only the first or the last layer of the neural network is retrained. Again, this research shows interesting results while leaving the opportunity to investigate the possibility of extending the forecasting horizon, in order to increase the applicability of the methodology for Smart Energy Management in buildings.

1.4. Contribution

The purpose of our research is to investigate and compare the effectiveness of ANNs to predicting indoor air-temperature. Since the investigation is carried out on time series data, the state-of-the-art ANNs which are most well suited for their analysis are LSTM and 1D-CNN. However, while LSTM has already been exploited in literature, successfully, to predicting indoor air-temperature, 1D-CNN has not, to the best of our knowledge, yet been applied to predicting indoor air-temperature, but is well suited for the analysis of time series data. Furthermore, the application of transfer learning on both LSTM and 1D-CNN brings further novelty since this method, to the best of our knowledge, has not yet been applied to predict indoor air-temperature. Furthermore, the research investigates the possibility of extending the forecasting horizon for indoor air-temperature prediction, in order to increase the applicability of the methodology for Smart Energy Management in buildings.

The neural networks are first pre-trained on a simulated dataset. A Building Information Model (BIM) of a real-world case-study building is developed, and subsequently used as an input to EnergyPlus, together with real meteorological data instead of Typical Meteorological Year (TMY) data, following the research carried out in [19]. This generates the simulated dataset, composed of 6 years of data, large and reliable enough to train the neural networks. The neural networks are then fine-tuned, through the application of transfer learning, on a real dataset, generated by Internet of Things (IoT) devices installed in the real-world building and used to collect one-month of indoor air-temperature measurements. All three transfer learning methods applied in [24] are used, in order to further evaluate how to improve the prediction accuracy.

Following the work carried out by [4], the neural networks are differentiated between those not considering the day of the week, and those considering it as an additional variable. All these different neural network architectures are used and their outcomes compared, in order to understand which architecture is most capable of delivering a model capable of most effectively predicting indoor air-temperature. The hyperparameters of each neural network are tuned following the same methodology, but each neural network is optimised individually, in order to identify the different set of parameters which allow each neural network to best fit the indoor air-temperature prediction. This approach increases the strength of the comparison, since it aims to maximize the accuracy of each neural network and make the accuracy of the results depend on the neural network architecture itself, rather than on the impact of poorly tuned hyperparameters.

To test our methodology, the proposed solutions were applied to a real-world school located in Turin (Italy). The experimental results are further validated by applying Fanger's model of thermal comfort [41], a method for evaluating thermal comfort in a building based on environmental and occupants, which is part of both ASHREE [42] and ISO 7730 [43] standards.

The rest of the paper is organized as follows. Section 2 presents the proposed methodology and its application in a real-world case-study building, which is a secondary school in Turin, Italy. Section 3 discusses the experimental results. Finally, Section 4 provides concluding remarks.

2. Material and methods

2.1. Enabling Technologies

This section will now provide a brief overview of the different neural network architectures investigated in this study, together with the different techniques applied to these architectures in order to train them and then to apply transfer learning on them.

2.1.1. One-Dimensional Convolutional Neural Network

Convolutional Neural Networks (CNN) are a type of network widely used in Computer Vision for image classification and object detection [44, 45]. A special type of CNN is the One-Dimensional Convolutional Neural Network (1D-CNN) [46]. The 1D-CNN is a novel architecture which, to the best of our knowledge, has not yet been applied to predicting indoor air-temperature. It is well suited for the analysis of timeseries data, like sensor measurements, signals or Natural Language Processing (NLP), especially with data that contains features which depend on short consecutive sub-sequences of the input. The 1D-CNNs differ from traditional CNNs in dimension of the input and in the way the filter slides across the data: as shown in Figure 1 the sliding operation is performed from top to the bottom, and there is no horizontal sliding. Otherwise, 1D-CNNs work similarly to traditional CNNs.

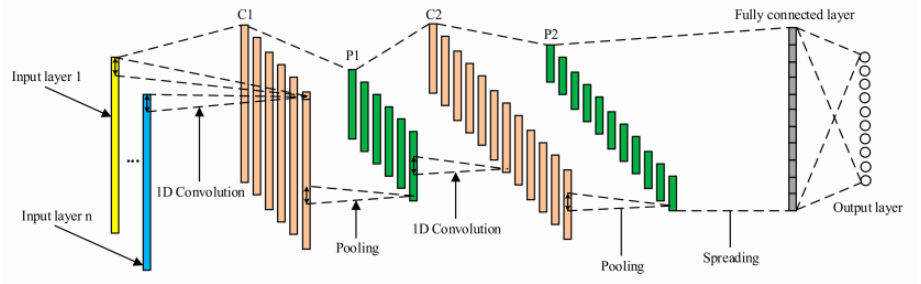


Figure 1: 1D-CNN architecture

The main advantage of applying a CNN is that it can automatically detect important features with a reasonable computationally cost, thus allowing it to deliver good performances in many applications.

2.1.2. Long Short Term Memory neural network

When considering a time-series, one of its fundamental characteristics is the strict dependence between a value at a given time t and the values in a window previous to t. In the neural network architectures previously discussed, all attributes are treated independently one from the other, thus losing this im-

portant relationship. On the other hand, Recurrent Neural Networks (RNN) address this problem [34]. In RNN there is one-to-one correspondence between the layers and the specific position in the sequence (referred as time-step). However, RNNs present an important training challenge due to the vanishing and exploding gradient problems [47], which cause them to have a good short-term memory and a poor long-term one, since they perform well when learning with short sequences, which require a small number of layers. The Long Short Term Memory (LSTM) neural network addresses this problem [48]. In the LSTM neural network architecture (Figure 2), the gate mechanism is introduced, thus modifying recurrence conditions on how the hidden states are propagated.

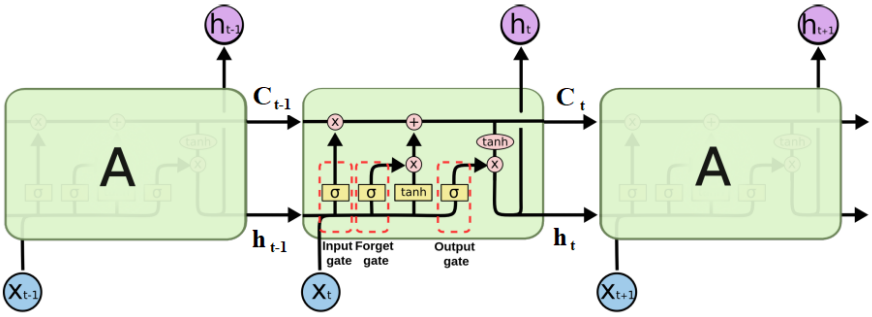


Figure 2: LSTM chain

2.1.3. Transfer learning

When humans approach the learning of new tasks, they leverage their previous knowledge. The more their previous knowledge is related to the new task, the easier it is to learn it. The same idea is at the base of transfer learning, a Machine Learning (ML) branch that consists of leveraging the knowledge learned in one or more source tasks and using it to learn a new related target task more accurately and quickly.

This is achieved by taking a model which has already been trained on a specific field, and continuing to train it using another training set. This has many advantages, including:

- **Less training data required for the new task / dataset**: if a model is trained to predict a certain outcome from a given dataset, and the new task is to predict the same outcome from a different dataset but related to a similar subject as the first dataset, the existing model is already trained to recognize common patterns in such datasets, and it thus requires far less data to tune the model to accurately predict the outcome also for the second dataset.

- **Faster training**: since less training samples are fed as input, the time required to train the model is far smaller.

- **Performances**: the transferred knowledge model (model trained on previous, similar data) shows better initial performance on the new target task compared to a brand new model. As a consequence, after the transferred model is trained on the new dataset it also performs better than a model trained from scratch.

When there is not enough data for training a neural network from scratch, transfer learning is a good alternative application, as shown in [40] and [24], since even in face of a small dataset it allows to create a predictive model capable of generating an accurate prediction.

Transfer learning is therefore applied also in this work, and three techniques are used:

- **Soft Start**: it consists of re-train the model with the target dataset, and having the pretraining on the source dataset initialize the weights.

- **Fine tune - last layer unfrozen**: using the pretrained model, the weights of all the layers are frozen except the ones of the last layer.

- **Fine tune - first layer unfrozen**: using the pretrained model, the weights of all the layers are frozen except the ones of the first layer.

2.2. Case study

This investigation is carried out on a primary school building of about 14.500 m^2, distributed on two floors, located in Turin, north-western Italy. The building is connected to the district heating system, which directly controls when the heating system is turned on and off, and is not equipped with a conditioning system. The windows in the building are installed on brick wall facades and are double glazed. Solar radiation provides substantial thermal energy contribution to both the east and west-oriented facades. The choice of this particular building typology makes the method applied on a large scale, evaluating possible retrofit actions on public assets. The energy management and simulation infrastructure developed to predict indoor air temperature in existing buildings aims to integrate heterogeneous data sources according to a challenging approach that combines Building Information Models, meteorological data, and IoT devices. In recent years, an increasing number of activities in public procurement has been managed with BIM tools. For this reason, we strategically consider the relevance of using digital models that reflect the actual characteristics of buildings to perform energy behaviour simulations in the Smart City scenario. As concerns the architectural and energy modelling of the building, we have adopted a BIM-based procedural workflow developed on previous research [49, 19] to optimize the activities and make the method replicable. The virtualization of the building was achieved using Autodesk Revit model authoring software and Design Builder. In contrast, the energy simulation in the dynamic regime was realized with the EnergyPlus engine. The BIM model is developed starting from archive documentation verified by an on-site survey to check the asset's current condition. Although the graphical representation must necessarily be simplified for energy purposes, the potential of a BIM model is to minimize misinterpretations and incorrect approximations of the building geometry encountered in traditional practice. To make it a repository of helpful, mutually consistent information for the analysis, the model must include: i) accurate building envelope characterizations in terms of correct stratigraphy, thermal and physical properties; ii) materials nomenclature standards, iii) thermal zones using rooms

entity; iv) facility management information (e.g. room type and occupants). The energy analysis model can be directly generated from the modelling software and exported through the green building eXtensible Markup Language (a.k.a gbXML) format. A complete exchange cannot be reached with the current level of software development, so a range of modelling solutions has been developed to maximize the interoperability process and speed up the subsequent re-input of data. Some deeper instructions on how to standardize the BIM includes the following: i) removal of all elements not required for simulation such as decorations, staircases, furniture; ii) walls must be correctly oriented and the walls joints must be configured; iii) selection of a representative stratigraphy for the external envelope and the internal partitions, iv) replace structural elements in architectural ones; v) the room bounding of architectural pillars must be disabled; vi) floor objects must be modelled following the centreline of walls; vii) aggregation of the areas by homogeneous thermal zones. Regarding the systems, only water radiators' dimensions were used to calculate the nominal capacity and flow rate for each radiator through schedules. The energy analysis model validation has been done by comparing the BIM component physical surfaces (i.e. walls, floors, roof, doors, windows) and the analytical ones according to an iterative optimization approach to obtain deviations lower than 10%. Design Builder is therefore used as the best user-friendly program interface to complete the input data filling needed for the simulation. Specifically, the critical energy-related inputs to manage concern materials which must be reassociated according to the software library, transparent surfaces that may not be correctly recognized, and systems. Data concerning the heating system as well as the energy load need to be manually set. All this information generates the building energy model through the IDF file, which is the output of Design Builder. It is used as input for EnergyPlus engine with real meteorological data instead of the default Typical Meteorological Year (TMY) to generate a large dataset of 6 years of synthetic data. The third-party weather data source from the nearest weather station (i.e. solar radiation, outdoor air temperature and humidity) has been used to allow for more accurate predictions, as demonstrated

by the research carried out in [19].

The building's internal rooms are analyzed to select those suitable for the study. Selection is based on symmetrical shapes and regular internal distribution, and the following three rooms are chosen:

- a classroom facing east (East Room);
- a classroom facing west (West Room);
- the corridor in the main entrance.

As far as such rooms are concerned, it is important to underline some particularities in both their occupancy and architectural structure. As far as the corridor is concerned, the number of people it contains undergoes significant variability during working days. The West and East rooms, on the other hand, are very large and characterized by many openings and glazed windows. The model of the building is shown in Figure 3.

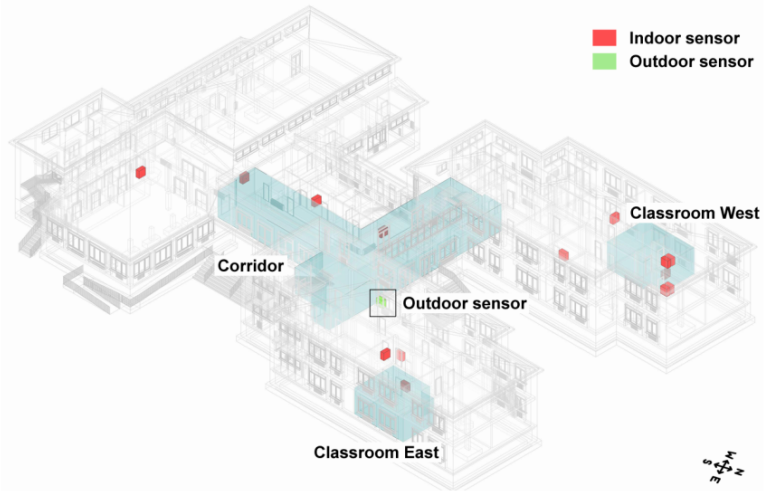


Figure 3: BIM model of the case-study building (reference rooms are highlighted in blue)

Each of the presented rooms generates its own dataset onto which the methodology is then applied. Furthermore, to carry out a more comprehensive study, the building as a whole is also considered through the generation

of a specific dataset named "Whole Building". The methodology, with which these datasets are generated, will be described below. The study is therefore carried out on the following 4 environments:

- Whole Building;
- East Room;
- West Room;
- Corridor.

The BIM model is leveraged in order to develop a specific dataset for each room: West Room, East Room, and Corridor. The BIM model of the whole structure is then leveraged in order to create the fourth dataset, "Whole Building". First of all, simulated temperature datasets are generated for all the rooms in the building. Non-significant spaces where students and staff are not expected to go on a daily basis, such as the location under the roof and underground rooms dedicated to archives and maintenance, are then excluded. The "Whole Building" dataset is then generated by averaging all other simulated temperature datasets.

For the real dataset, a mesh wireless sensor network with 12 prototype nodes was installed in the building in order to collect one month of real data during the winter season. Each prototype node, presented in [50], was developed using nodes assembled with off-the-shelf devices belonging to the STM32 NUCLEO ecosystem by STMicroelectronics: i) a NUCLEO-L152RE board equipped with a 32-bit microcontroller running at up to 32MHz; ii) an X-NUCLEO-IKS01A1 motion MEMS and an HTS221 temperature and humidity sensor; iii) an X-NUCLEO-IDS01A4 radio transceiver board equipped with the SPIRIT1 transceiver (868MHz); and iv) a STEV AL-ISV021V1 energy harvester board, equipped with an SPV1050 MPPT converter and battery charger, a 26.5 cm^2 solar panel and a 120mAh Lithium coin cell battery.

To collect the information, we used the distributed software infrastructure presented in [19]. It is a cloud-based software architecture that integrates het-

erogeneous hardware and software components for monitoring and modelling building energy behaviours. The infrastructure exploits pervasive geo-referenced IoT devices to collect environmental and energy related information. IoT devices are correlated with BIM, GIS and weather data exploiting middleware technologies. It allows energy simulations to evaluate both efficiency performances of buildings and effects of possible refurbishment actions. Furthermore, the data provided by the IoT devices can be used also to further monitor and control both environmental and energy parameters of buildings in (near-) real-time.

Figure 4 represents the "simulated" and "real" datasets. Both datasets are similar in structure: data, type of data and sampling time are all the same. Both the simulated and the real datasets are composed of indoor temperature readings at 15-minutes intervals. The datasets from the sensors located in selected rooms are then taken to carry out the transfer learning application. The "Whole Building" real dataset is instead created by averaging the temperature measurements from all the sensors installed in the school, not just the ones in the selected rooms.

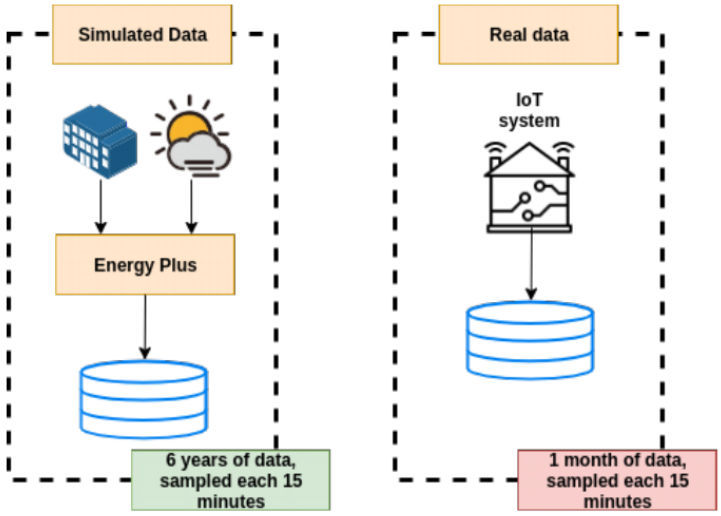


Figure 4: Datasets

After observing the indoor air-temperature's behaviour (Figure 5), a clear

difference between weekdays and weekends can be perceived. This is because the HVAC system is shutdown during weekends, resulting in both lower indoor air-temperature absolute values and variations. Furthermore, the behaviour on Monday also differs from the one of the other weekdays. This is because since during the weekend the HVAC system is shutdown, on Monday the indoor air-temperature requires more time to reach its required level. This behaviour is taken into consideration during the investigation, as explained.

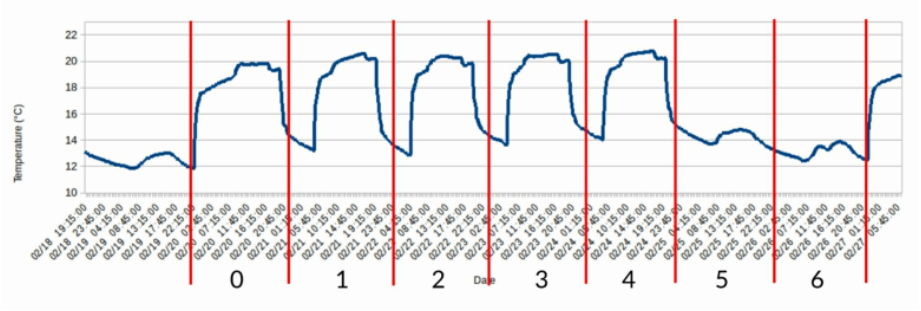


Figure 5: Day of the week behaviour

As previously explained, each environment listed above generates its own dataset. The different neural networks, or more specifically their hyperparameters, are therefore optimized for each environment, so for each neural network type there is a specific version for each environment. Further versions of the neural networks will also come from whether the different behaviour during the weekdays and weekends is taken into consideration.

Finally, it is important to understand why the investigation creates specific neural networks for the individual rooms, and for the building as a whole. At an individual room level, the performance of the neural networks can be evaluated in terms of their potential application as far as capillary control applied to single buildings is concerned, and is therefore related to Smart Building applications [2]. At building level, on the other hand, the results of the investigation can be evaluated for applications of Demand/Response (DR) [5] or Demand Side Management (DSM) [6] at city level, such as District Heating [8].

2.3. Evaluation metrics

When evaluating the performance of the neural networks, it is important to understand the objective of the application and set evaluation metrics which can effectively evaluate the performance compared to such application. The objective of this work is to evaluate whether the neural networks are effective in forecasting future indoor air-temperature within ranges which can be considered acceptable for thermal comfort (comfort analysis is better explained below). The achievement of extremely accurate prediction is not the objective of the study, but rather the ability of the prediction to remain within a certain threshold. Following such considerations, the metrics used to evaluate the performances of the neural networks are the Mean Absolute Error (MAE) and the Predicted Mean Vote (PMV), with $\hat{y}_t$ as the predicted values and y_t the observed ones. Furthermore, since the predictions are carried out on a number of different data subsets because of the difference between the size of the data set and the number of steps ahead of the prediction, the resulting MAEs and PMVs are analyzed by taking their mean, median, max, 1 standard deviation and 2 standard deviations.

In order to evaluate if the indoor air-temperature forecasts remain within ranges which can be considered acceptable for thermal comfort, the parameters studies by Fanger [41], the Predicted Mean Vote (PMV), which estimates the state of well-being of a group of individuals, and the Percentage of Person Dissatisfied (PPD), which expresses the percentage of people that are uncomfortable with thermal condition of the room, are used, following the logic used by current thermal comfort evaluation standards, both based on Fanger's studies, ASHREE [42] and ISO 7730 [43].

Fanger's studies on thermal comfort determine that in order to maintain the perceived thermal comfort of an occupant, the operative temperature should not vary more than 1.1 °C over 15 minutes and no more than 2.2 °C over 1 hour [41, 51]. As a consequence, the acceptable MAE threshold for acceptable prediction accuracy is set at 2 °C. Following the standards identified in literature, the PMV is therefore analyzed as an absolute value and its acceptable threshold for

acceptable comfort levels is set at 0.5.

2.4. Methodology

The purpose of this section is to discuss the methodology adopted during the implementation of the different neural network models, from the data collection phase to the final testing phase, together with explaining the motivations that led to certain choices during the implementation.

As represented in Figure 6, this paper investigates the application on two different neural networks: One-Dimensional Convolutional Neural Networks (1D-CNN) and Long Short Term Memory (LSTM) neural networks. After the initial training, the neural networks undergo transfer learning. Three different transfer learning techniques, presented in Section 2.1.3, are applied in order to investigate the best performance in terms of prediction.

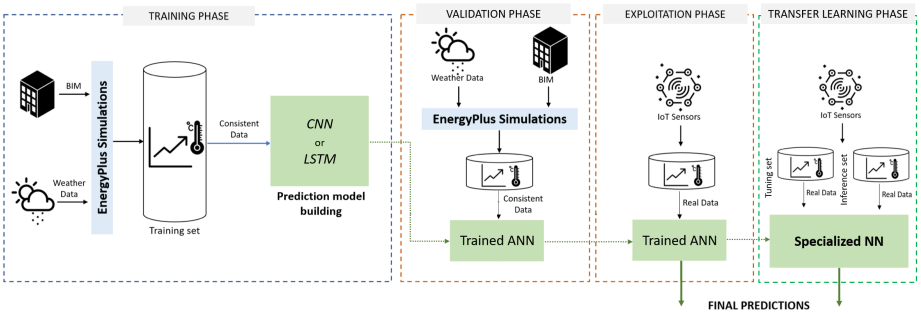


Figure 6: Schema of the proposed methodology

As introduced in section 1.4, the neural networks are first trained on a synthetic dataset, composed of 6 years of data, large and reliable enough to train the neural networks. To generate this synthetic dataset, which is referred to as "simulated", a Building Information Model (BIM) of a real-world case-study building is developed, and subsequently used as an input to EnergyPlus, together with real meteorological data instead of Typical Meteorological Year (TMY) data, following the research carried out in [19]. The testing and transfer learning phases are then carried out using the real dataset, which is referred to as "real". This real dataset is generated by Internet of Things (IoT) devices

installed in the real-world building and used to collect one-month of indoor air-temperature measurements.

The purpose of the investigation is the forecasting of indoor air-temperature when the HVAC system is functioning. The building in the case-study is equipped with a heating system and is not equipped with a conditioning system. For this reason, within the datasets only the datapoints from the days during which the heating system is turned on are considered. The building's heating system is connected to the district heating system, which directly controls when the heating system is turned on and off. The annual interval during which the district heating system is turned on is between October 15th and April 15th. Table 1 presents a detailed overview of the datasets, their size, the interval considered for each year and the size of these intervals. Finally, in order to allow first the traditional neural network training and then the fine tuning through transfer learning, the simulated dataset is split into a training set and a validation set, whereas the real dataset is split into a tuning and an inference set. More details on the way the datasets are split are found in section 2.4.1.

Table 1: Size of datasets

Year	Type of data	Number of datapoints	Data interval considered	Number of datapoints in the interval
2010	Simulated	35.040	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.472
2011	Simulated	35.040	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.472
2012 *	Simulated	35.136	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.568
2013	Simulated	35.040	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.472
2014	Simulated	35.040	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.472
2015	Simulated	35.040	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.472
2016 *	Real	35.136	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.472
2017	Real	35.040	01 Jan - 15 Apr; 15 Oct - 31 Dec	17.472

* Leap year

Following the previously described data collection, the four main phases required to develop any new predictive model are: i) data pre-processing, ii) training, iii) validation and iv) testing phase. Each phase is explained in more detail in the rest of this section.

2.4.1. Data pre-processing

Any dataset must first of all be pre-processed before being used to train any network or model. During preprocessing the data is cleaned, outliers are removed, then the data is scaled and finally split into datasets to be used for training, validation and testing. Errors in the data are caused mainly by sensor malfunction, which generate single out-of-scale readings (i.e. outliers), or no readings at all. Samples affected by these errores are replaced with linear interpolation between their previous and following values. Data pre-processing has to be applied only to the real dataset, since the simulated dataset is complete and consistent.

The data is then subsequently scaled, as suggested by Aggrawal in [52], and brought within a range of -1 and 1. Finally, in order to allow first the traditional neural network training and then the fine tuning through transfer learning, the datasets are split in the following way:

- **Training set**: simulated dataset, years 2010 to 2013;
- **Validation set**: simulated dataset, years 2014 to 2015;
- **Tuning set**: 80% real data;
- **Inference set**: 20% real data.

2.4.2. Training the neural network

The neural networks are then trained to carry out predictions for indoor air-temperature, based on a window of indoor air-temperatures received as an input. These predictions can be one-step ahead, meaning that only the successive temperature value after the last temperature input is being predicted, or multi-step ahead, where multiple time steps after the last temperature input are being predicted. The number of steps, or period of time in the future, of the prediction is called forecast horizon. The objective of this investigation is, considering a certain threshold of acceptability in prediction error determined by Fanger, to develop a model capable of predicting indoor air-temperature values with a forecast horizon as long as possible.

When carrying out a multi-step prediction, three different strategies can be used: i) *Joint method*, i.e. a single network is used to predict all forecast horizons; ii) *Independent method*, i.e. a dedicated network is used to predict each forecast horizon; iii) *Iterative method*, i.e. a single step-ahead model is used to iteratively generate forecasts. According to Kline's study [53], the Joint Method delivers better prediction accuracy for long forecast horizons (more than 4 steps-ahead). This investigation therefore uses the Joint method.

In order to be compatible with supervised learning algorithms, the time series data to be used as input to the neural network must be processed. This time series data, a sequence, is transformed into pairs of input and output sequences. Since the Joint method is being used, the sliding window algorithm is used to associate each sequence of input to a sequence of outputs (Figure 7).

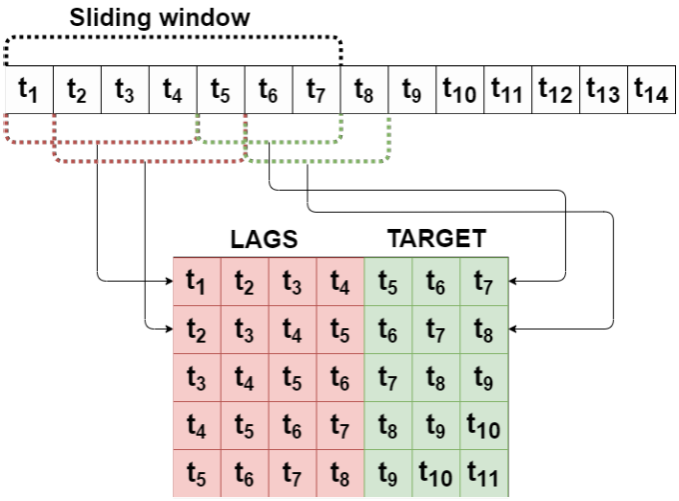


Figure 7: Reframed samples

Once the input data is processed and made suitable for network training, the network must also be prepared. The different neural networks' hyperparameters must be tuned, in order to develop network structures which best fit the prediction problem being investigated.

2.4.3. One-Dimensional Convolutional Neural Network

The chosen 1D-CNN architecture is based on the one proposed by Mehrkanoon in [54] and it is represented in Figure 8. It has the following layers:

1. Input: It accepts a dimension input *(steps, features)*, where *steps* is the prediction lag (the number of previous temperature values that are used as regressors) and the only feature is temperature.
2. Convolutional: It has a convolutional layer vector of size 2, which after the convolution operation generates an output of length equal to input dimension minus 1. The second dimensionality is equal to the number of filters, which is chosen based on the result of different trials.
3. Pooling: The pooling strategy used is the Max Pooling with pool size equal to 2, with the dimension of the output being halved as a consequence.
4. Flatten: Because of the multiple filters used there is a multidimensional output, so a flatten layer is used. In order to feed the data to the dense layer, the flatten layer is used to "unroll" the data.
5. Dense: In order to improve the learning ability of the network, an intermediate dense layer is used. The number of units of this layer is decided in the tuning phase.
6. Output: The output layer consists of a number of neurons equal to the number of forecasting horizons.

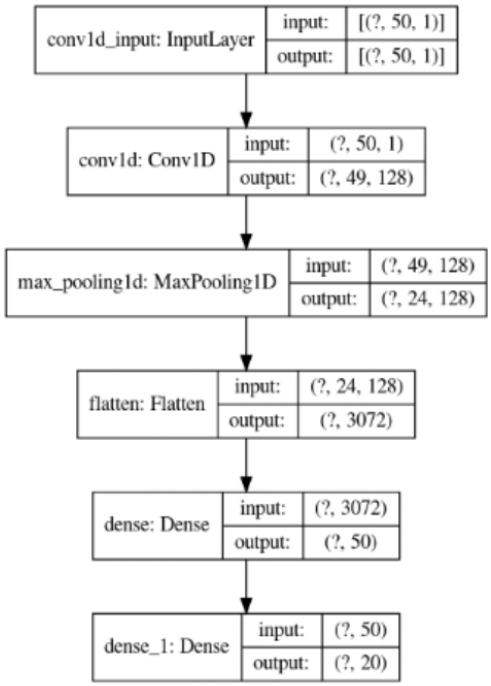


Figure 8: Architecture of the One-Dimensional Convolutional Neural Network

Furthermore, Early Stopping technique with a patience of *8 epochs* and a delta of *0.003* is used to avoid overfitting. The activation functions chosen are ReLU for the Convolutional and Dense layers, while linear for the Output.

During the tuning hyperparameters phase, the best set of parameters for the neural network with this dataset and purpose are searched for. The tuning of such hyperparameters is carried out following a trial and error strategy, but with the parameters being tuned in the following order: prediction lag (inputs), number of hidden neurons of the dense layer, batch size, learning rate, number of filters (filter optimization). Once the set of parameters is chosen, the model is evaluated on the test set for predictions until 1 day (96 steps) ahead.

The number of regressors is the first parameter to be optimized in this neural network. The neural network is evaluated with 16, 32, 64, 96 and 128 timesteps. In order to evaluate the performance on this parameter, the other parameters

are kept constant with the following values: (i) batch size: 128; (ii) filters: 64; (iii) forecasting horizon: 7,5 hours; (iv) hidden neurons: 100. The results are shown in Figure 9. The accuracy of the prediction is evaluated, together with the risk of overfitting, but also network complexity and training time are considered. The objective is to identify the tradeoff between network accuracy and training time. Of course the larger the input size, the more accurate is the prediction. However, an upper limit is set at 120 inputs (30 hours before), since an even bigger value would add too much complexity to the network, increasing both the training time and the risk of overfitting. It is interesting to highlight how even though 120 neurons guarantee a greater accuracy, the improvement compared to the previous input size is negligible, especially compared to the increase in training time. Following these considerations and the presented results, all environments except for West Room are assigned an input size of 96, whereas West Room is assigned an input size of 64.

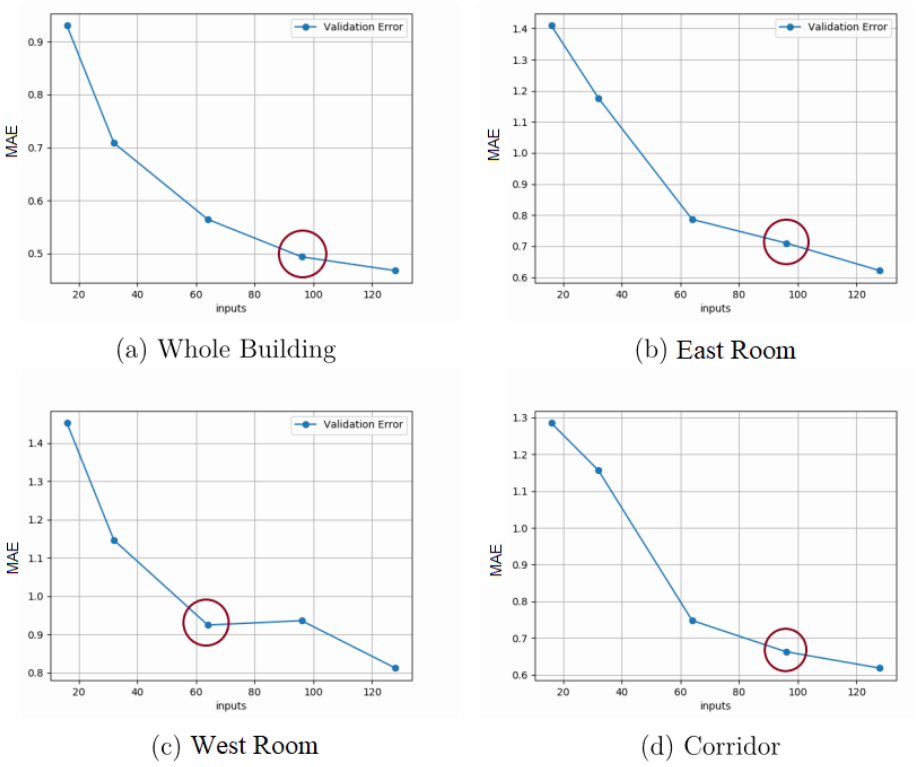


Figure 9: 1D-CNN, MAE variation against number of regressors

The second parameter to be optimized in this neural network is the number of hidden neurons in the first Dense layer. During this parameters optimization, the other parameters set during the optimization of the inputs remain unchanged. The results are presented in Figure 10. Again, the tradeoff between accuracy and training time is a key element in determining the value of this parameter. The number of neurons for the different environments is kept between 50 and 100, since the results showed how increasing the number of neurons does not produce a significant increase in accuracy, as shown by the value of the error which is fluctuating within the very small range of 0.03 °C.

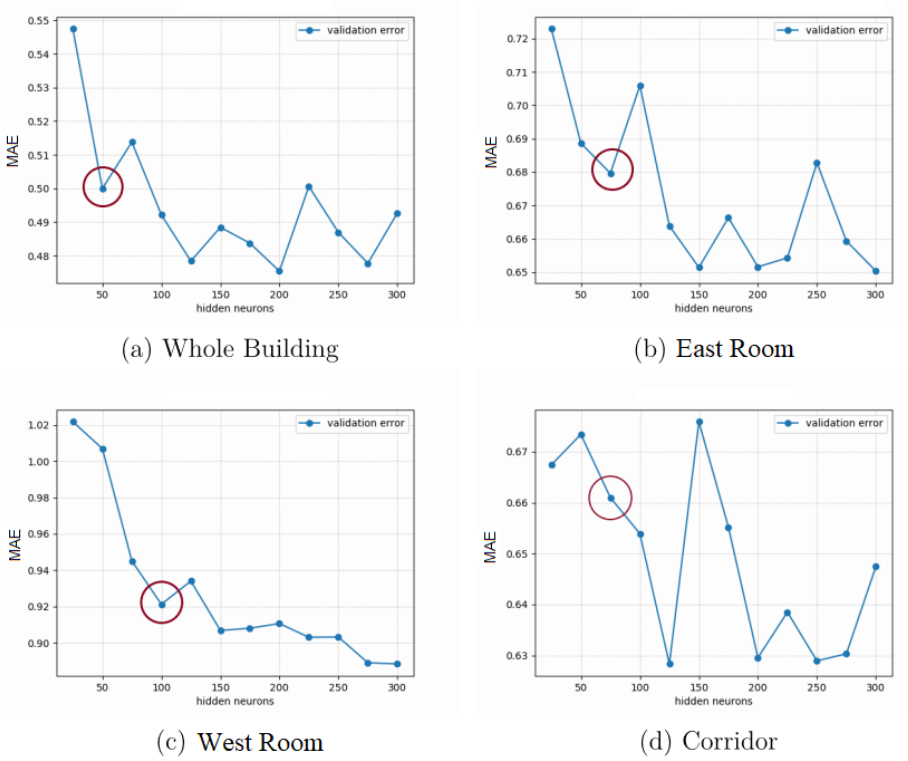


Figure 10: 1D-CNN, MAE variation against number of neurons

The next step is the optimization of two hyperparameters together, batch and learning rate. Batch size, more specifically, is one of the most important hyperparameters to tune in modern deep learning systems. The tradeoff between large batch size, which generates high speed, accuracy, and the ability of the model to generalize are always key points to consider and, when training neural networks, one must consider the behaviour of such parameters in non-convex optimization. In this investigation, batch and learning rate are acted upon together because of the close link between them: batch size affects the number of iterations, while learning rate affects the "speed" of the convergence. One would therefore expect a small batch size with a small learning rate to achieve similar results compared to a large batch and a large learning rate. Also, the benefits of smaller batch sizes are also expected to include better generalization,

together with higher speed. This decrease in batch size cannot however be excessive, since if the batch size is too low (for example 1) it will be unable to generalize and will risk falling into over-fitting. Figure 11 highlights the results. In this case, accuracy is the driving factor in selecting the best hyperparameters. Once comparable accuracy values are identified, the second choice factor is taken into consideration, training time. The parameters allowing for the simpler (and faster to train) model are therefore selected, so the largest possible batch size and learning rate allowing to maintain the same level of accuracy.

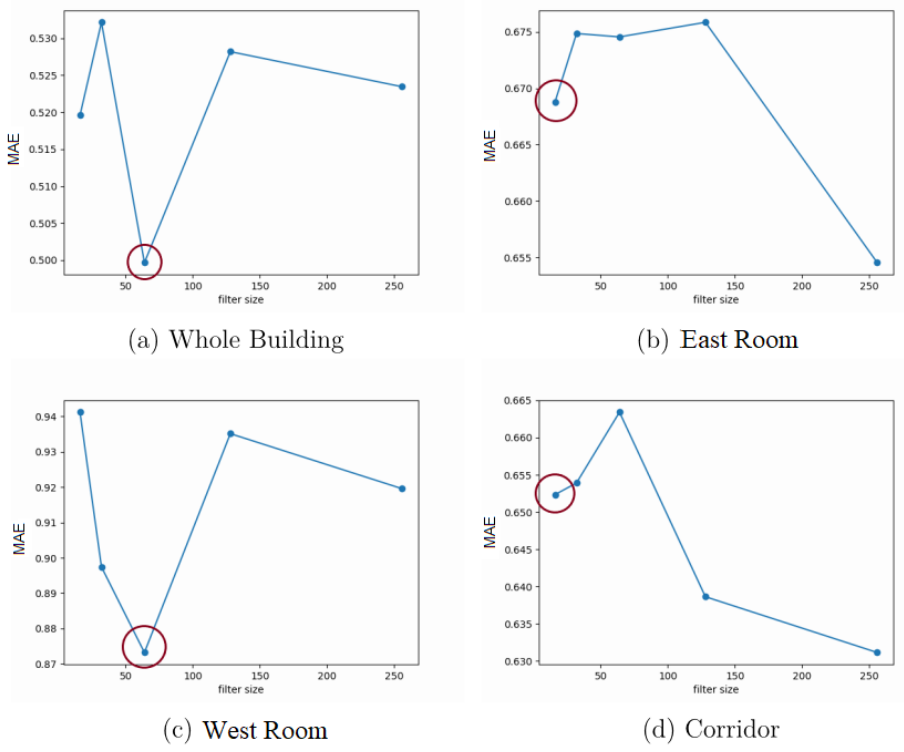
(a) Whole Building (b) East Room

(c) West room (d) Corridor

Figure 11: 1D-CNN, MAE variation against batch and learning rate

The last step in the optimization of the hyperparameters for this neural network is the optimization of the number of filters used in the convolution. All

the other parameters are already identified, so the performance of the neural network is evaluated with different number of filters, as shown by Figure 12. Like in the optimizations of the previous parameters, the accuracy performance of the neural network is considered together with network complexity and consequent training time. It is interesting to highlight how for some environments, West Room and Whole Building, an increase in the number of filters actually degraded the performance, possibly as a consequence of overfitting.



(a) Whole Building (b) East Room

(c) West Room (d) Corridor

Figure 12: 1D-CNN, MAE variation against number of filters

Table 2 resumes the parameters for this neural network chosen for the 4 environments.

Table 2: 1D-CNN parameters

Zone	Regressors	Neurons	Batch	Learning rate	Filters
Whole building	96	50	64	0,001	16
East Room	96	75	32	0,001	16
West Room	94	100	32	0,001	16
Corridor	96	75	256	0,001	16

2.4.4. Long Short Term Memory neural network

The architecture chosen for this neural network, represented in Figure 13, is the stateful vanilla model, which consists of a single hidden LSTM layer. The following layers make up the network:

- Input: It accepts a shape of *(batch size, lags, number of features)*.
- LSTM: Composed of LSTM cells, whose number is decided in the tuning phase.
- Output: The dimensionality of the LSTM layer matches the length of the forecasting horizons. Its activation function is the linear function.

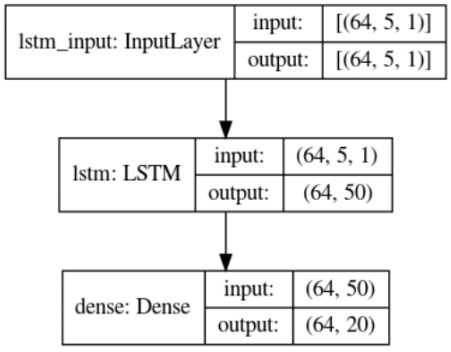


Figure 13: Architecture of the Long Short Term Memory neural network

The parameters being tuned for this neural network architecture are batch size, number of LSTM cells, number of regressors and the learning rate. Like for the other neural network, also the tuning of these hyperparameters follows a trial and error approach, with parameters being tuned in a specific order. The

starting set of chosen parameters are: (i) batch size: 256; (ii) forecasting horizon: 7,5 hours; (iii) training epochs: 100; (iv) number of LSTM cells: 50. This is the starting configuration, after which different inputs are tried to find the most performing ones. First the number of regressors (inputs), then the number of epochs are optimized, analysing the behaviour of loss during training and stopping the training when there is no more significant improvement. Finally, the batch size and the learning rate are optimized together, because of their relationship, by creating a grid with their values coupled *(learning rate, batch size)*.

This section now presents the methodology followed to optimize the hyper-parameters of the LSTM neural network. The same considerations discussed during the 1D-CNN optimization related to the selection of the most suitable inputs, batches and learning rate, remain valid also for LSTM.

The first parameter to be optimized is the number of regressors used by the neural network. The LSTM structure is characterized by a cell mechanism capable of "memorizing" the state of the previous batches, so the number of inputs for this neural network can be smaller than the one in 1D-CNN networks. The network was therefore evaluated for 1, 3, 5, 7, 9 and 11 timesteps. In order to evaluate the performance on this parameter, the other parameters are kept constant with the following values: (i) batch size: 256; (ii) forecasting horizon: 7,5 hours; (iii) training epochs: 100; (iv) number of LSTM cells: 50. This initial configuration is replicated for all environments, then the neural network performance in terms of prediction accuracy is evaluated with different input size, as shown by the results presented in Figure 14.

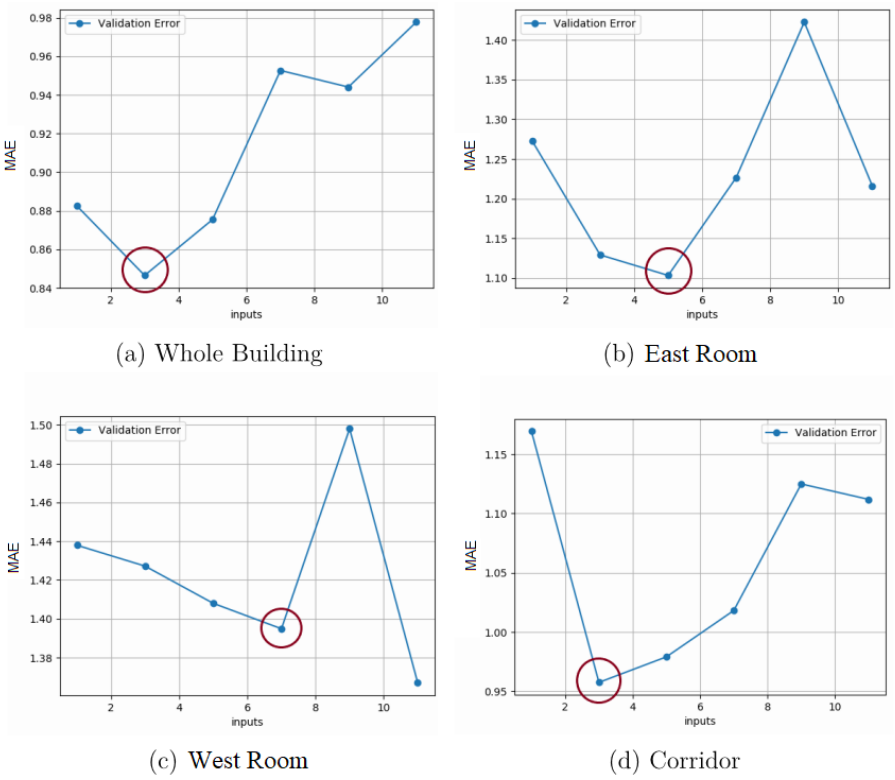
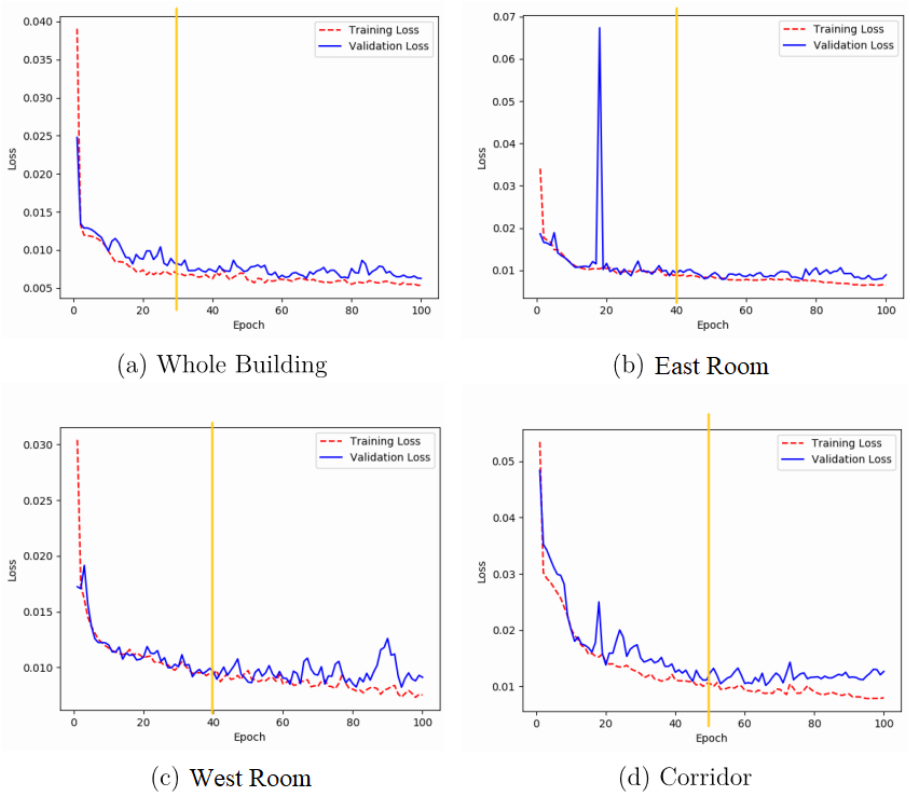


Figure 14: LSTM, MAE variation against number of regressors

When determining the best value for the epochs parameter, the risk of overfitting and the improvement in network learning are taken into consideration. The loss at each epoch is therefore considered, and when an epoch value is reached such that the network learning capacity becomes negligible, the training can be stopped and the number of epochs can be determined, as shown in Figure 15. This optimization is carried out with a batch size of 256, which has been set as the maximum batch size in order to allow a conservative stopping point, since a smaller batch size would allow a greater number of iterations.



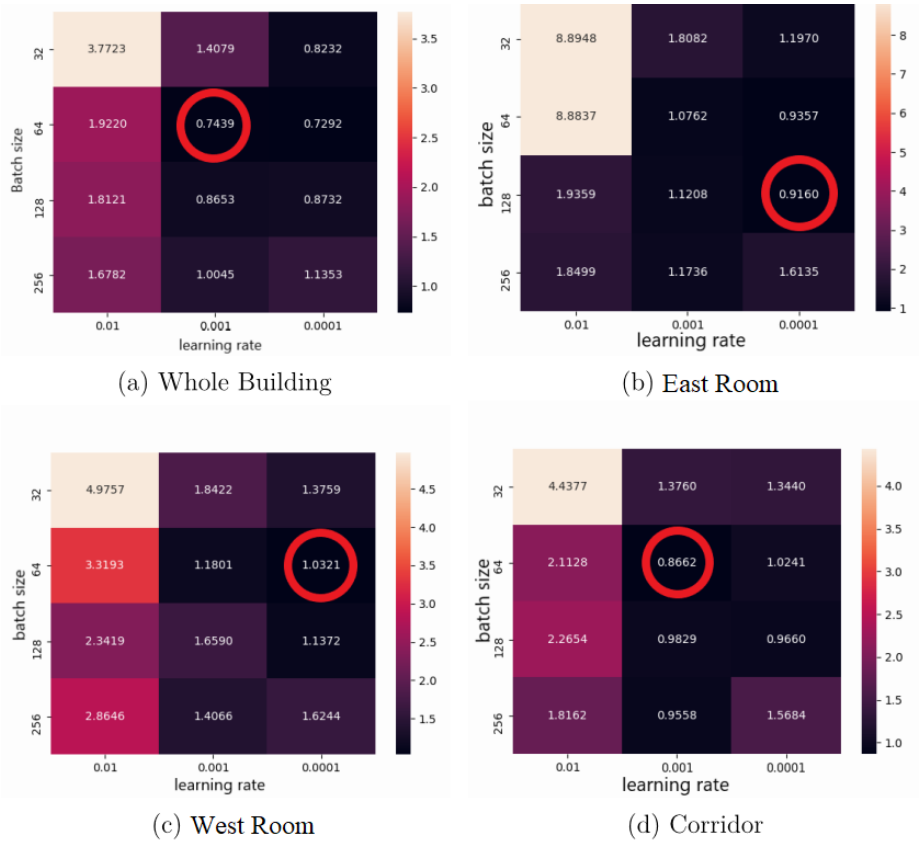
(a) Whole Building (b) East Room

(c) West Room (d) Corridor

Figure 15: LSTM, epochs loss

The next phase in the optimization of the hyperparameters is the optimization of batch size and learning rate, which are again considered jointly because of their impact on, respectively, number of iterations and "speed" of the convergence. As previously stated, the optimization of batch size is a particularly important step because of its impact on speed, accuracy and the ability of the model to generalize. Again, similar results are expected between small batch size with a small learning rate and large batch and a large learning rate, better generalization is expected by smaller batch sizes and attention is paid to avoid excessive decrease in batch size, in order to maintain the ability of the model to generalize and to avoid over-fitting. The batch size values being tested are 32, 64, 128 and 256, while 0.01, 0.001, 0.0001 are the ones for the learning rate. The results are presented in Figure 16, which shows that there is a high variability in

the accuracy. In this case, the best accuracy performance is therefore achieved with small learning rates combined with small batch sizes.



(a) Whole Building (b) East Room

(c) West Room (d) Corridor

Figure 16: LSTM, MAE varition against batch and learning rate

The final phase of the optimization of the hyperparameters is the optimization of the neurons, or number of LSTM cell units. All the other parameters are already identified, so the performance of the neural network is evaluated with different number of units: 50, 100, 150, 200, 250, 300. The results are shown in Figure 17: it is interesting to highlight how as the number of LSTM units increase beyond a certain value, the error also increases, probably because of overfitting.

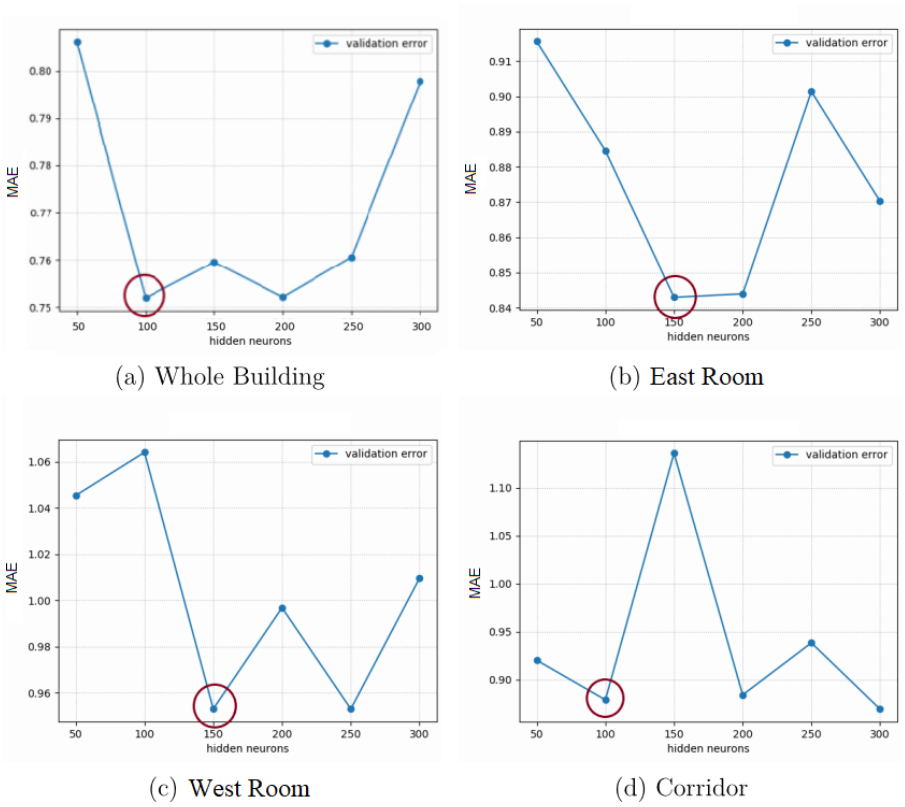


Figure 17: LSTM, MAE variation against cell number

Table 3 resumes the parameters for this neural network chosen for the 4 environments.

Table 3: LSTM parameters

Zone	Regressors	Neurons	Batch	Learning rate	Filters
Whole building	96	50	64	0,001	16
East Room	96	75	32	0,001	16
West Room	94	100	32	0,001	16
Corridor	96	75	256	0,001	16

2.4.5. The multivariate approach

The methodology presented previously to be applied to the selected neural networks is being implemented by considering indoor air-temperature as the

only input feature. In this section the investigation wants to introduce another input feature, being the day of the week.

Previous works, such as the one carried out by Deihimi et a. [4], show how if the day is also added as input variable to the neural networks, it can result in greater prediction accuracy. As highlighted previously, after observing the indoor air-temperature's behaviour (see Figure 5), a clear difference in behaviour between weekdays and weekends can be perceived. The objective is to try to "help" the neural network to capture this pattern, adding a different label to identify each day of the week. This application has been executed with both the 1D-CNN and LSTM architectures, while the rest of the parameters defined in the original neural networks remain unchanged. This is because the purpose of this application is to understand the impact of the second variable, therefore the exact same architectures and parameters must be used. The neural networks applied on datasets which do not consider the day of the week are called "univariate" cases, whereas the ones applied on datasets which include a different label to identify each day of the week are called "multivariate" cases.

2.4.6. Fine Tuning using real data

Starting from the models trained according to the procedure explained in the previous paragraphs, the model is then retrained through transfer learning in three different ways: i) retrain all the layers; ii) retrain only the input layer; iii) retrain only the output layer. As discussed in [55], and in [56], the different transfer learning retraining approaches impact mainly accuracy and computation time. If none of the layers of the models are frozen, the accuracy of the trained model will be greater, but will also require greater computational time. On the other hand, if all layers but the last one are frozen, the model required backpropagating and updating the weights only of the unfrozen layer, which brings a significant decrease in computational time. The different solutions are therefore investigated.

For the purpose of this investigation, the accuracy of the different transfer learning approaches is the only aspect which has been considered. At the

beginning of the investigation, it was observed that there was no appreciable difference in time between the fine tuning of the neural networks with the different transfer learning techniques. What has been noted however is the difference in time required to train the original neural networks, compared to their fine tuning using transfer learning. Whereas the time required for the initial training remained within the order of magnitude of numerous months, the fine tuning through transfer learning could be completed with a few hours only. This allowed, in the same unit of time, to launch a greater number of transfer learning simulations with different approaches, and to collect a greater number of results on the effects of different transfer learning methods. However, the true purpose of the investigation was to verify if the fine tuning, through the application of transfer learning on a much smaller dataset (2 years, compared to the 6 years for the initial training), was capable of generating a model with an acceptable level of accuracy.

3. Results and discussions

The first step in evaluating the investigation's results is to verify the performance of the initial neural networks, trained on the simulated datasets, before the application of transfer learning with the real datasets. Both the 1D-CNN and the LSTM neural networks are trained on each individual environment: Whole Building, East Room, West Room and Corridor. Furthermore, each neural network in each environment is also applied on both the univariate and the multivariate case, with the latter including the label of the day of the week as further input. As described in Section 2, following Fanger's studies [41] and the current standards to evaluate thermal comfort based on such studies [42] [43], the MAE and PMV thresholds for acceptable prediction accuracy and comfort levels are set at 2 °C (MAE) and 0.5 (PMV). The forecast horizon (in hours), that each neural network is able to achieve while maintaining within such thresholds before the application of transfer learning, is presented in Table 4.

Table 4: Performance of original neural networks before the application of transfer learning: the best performance for each environment is highlighted in grey

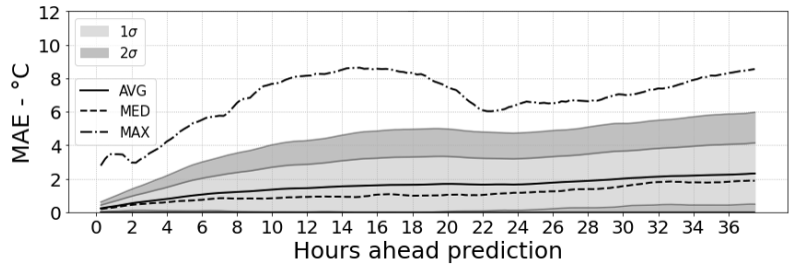
			ORIGINAL NEURAL NETWORKS	
Environment	**Case**	**Network**	**Max forecast horizon (h) with MAE <= 2°C**	**Max forecast horizon (h) with PMV <= 0.5**
Whole building	Univariate	1DCNN	32	32
		LSTM	35	36
	Multivariate	1DCNN	25	10
		LSTM	11	17
East Room	Univariate	1DCNN	36	34
		LSTM	27	36
	Multivariate	1DCNN	24	36
		LSTM	29	36
West Room	Univariate	1DCNN	28	36
		LSTM	29	36
	Multivariate	1DCNN	7	32
		LSTM	5	36
Corridor	Univariate	1DCNN	30	36
		LSTM	31	36
	Multivariate	1DCNN	25	35
		LSTM	28	36

The above results provide some first interesting insight on the performance of the neural networks. First of all, the forecast horizons for the MAE and for the PMV are not the same. This difference can however be expected, considering the different nature of the two metrics: the MAE evaluates the accuracy of the neural network (the difference between the $\hat{y}_t$ predicted values and the y_t real values), while the PMV evaluates the estimated state of well-being of a potential group of individuals given the $\hat{y}_t$ predicted values. The actual forecast horizon of each neural network application can therefore be considered to be the lowest between the two forecast horizons. The above results show that the PMV forecast horizon appears to be, on average, greater than the MAE forecast horizon. As far as comparing the performance of the univariate and multivariate cases, the latter clearly underperforms against the former in all environments. As far as the neural network type is concerned, the LSTM neural networks

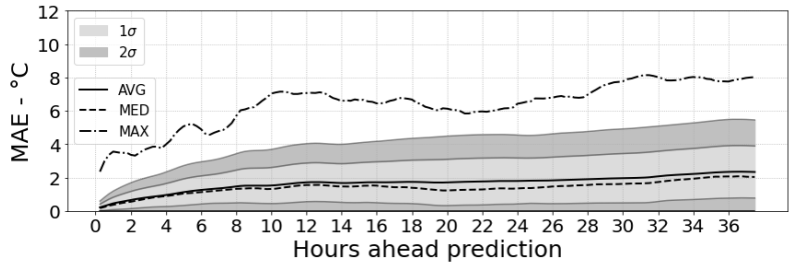
appears to be better performing compared to the 1D-CNN ones, achieving the longest forecast horizon in all environments apart for the East Room.

However, apart for observing the maximum forecast horizon achievable by the different neural network applications, it is also important to observe the shape of the graphs resulting from plotting the MAE and PMV metrics against the forecast horizon. Such graphs provide further interesting insight in understanding the variability in performance of the neural networks, since they present the mean, median, max, 1 standard deviation and 2 standard deviations of the MAE and PMV metrics.

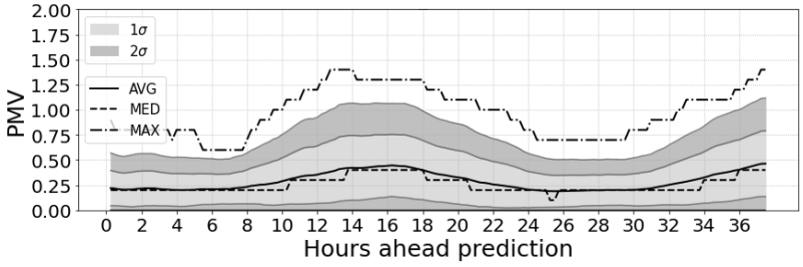
For example, Figure 18 presents the graph of the metrics for the neural networks of the Corridor environment, univariate case. Whereas the MAE graphs for both the 1D-CNN and LSTM neural networks appear to be quite similar in shape, the PMV plots for the two neural networks present very different behaviours. The 1D-CNN shows a sinusoidal trend, with increasing divergence and therefore degrading performance as the forecasting horizon increases, whereas the LSTM neural network shows a converging trend, with its performance appearing to improve as the forecasting horizon increases. These trends are even more significantly portrayed in Figure 19, which zooms under the MAE and PMV thresholds. Finally, it is important to notice the behaviour not only of the mean, but also of the median and the 1 standard deviation. In both the 1D-CNN and the LSTM graphs, the 1 standard deviation limits appear to be evenly distributed around the mean line, showing a consistent distribution in the prediction of the different outcomes for each forecasting horizon. Furthermore, the median line is consistently lower than the average line, especially for the MAE graphs, thus showing how the distribution of the prediction of the different outcomes for each forecasting horizon tends to be skewed towards lower, acceptable values. If the median line, rather than the mean line, was considered to evaluate the forecasting horizon achievable by the neural network, such horizon would therefore further increase.



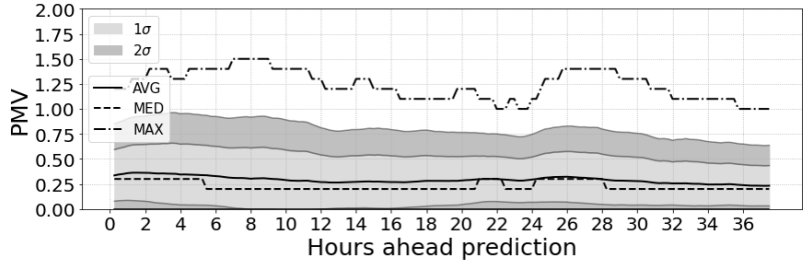
i) Corridor temperature predicted by 1D-CNN univariate original network with n_out = 150



ii) Corridor temperature predicted by LSTM univariate original network with n_out = 150

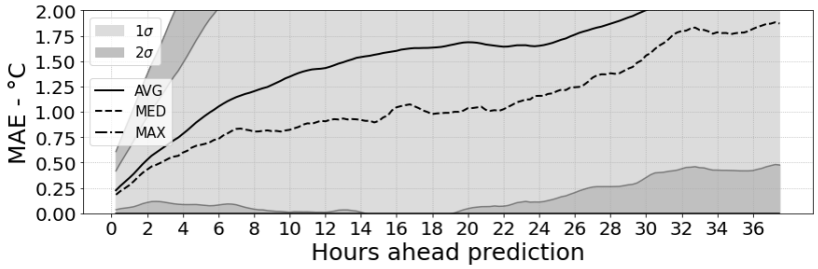


iii) PMV for Corridor temperature predicted by 1D-CNN univariate original network with n_out = 150

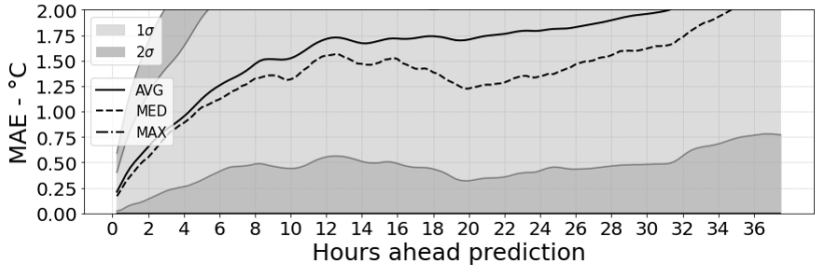


iv) PMV for Corridor temperature predicted by LSTM univariate original network with n_out = 150

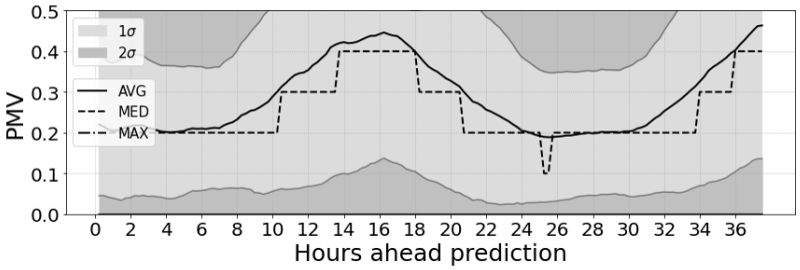
Figure 18: Performance of the Corridor univariate original neural networks before the application of transfer learning



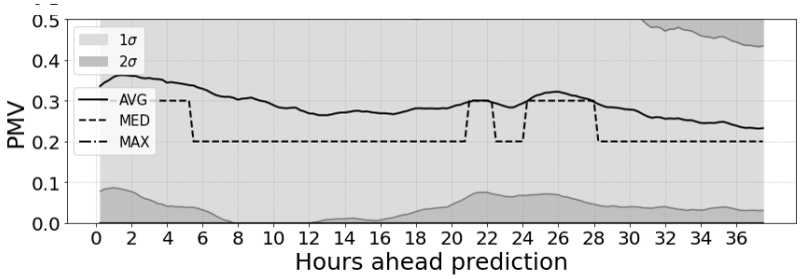
i) Corridor temperature predicted by 1D-CNN univariate original network with n_out = 150



ii) Corridor temperature predicted by LSTM univariate original network with n_out = 150



iii) PMV for Corridor temperature predicted by 1D-CNN univariate original network with n_out = 150



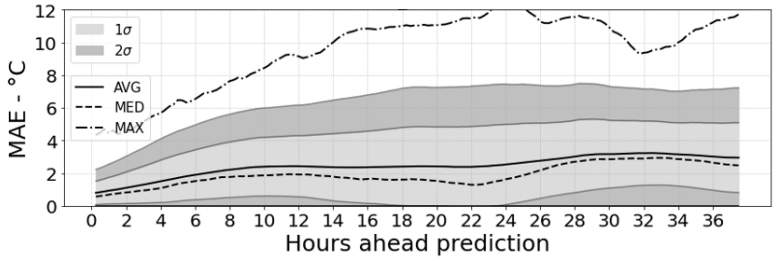
iv) PMV for Corridor temperature predicted by LSTM univariate original network with n_out = 150

Figure 19: Performance of the Corridor univariate original neural networks before the application of transfer learning - zoom on MAE and PMV thresholds

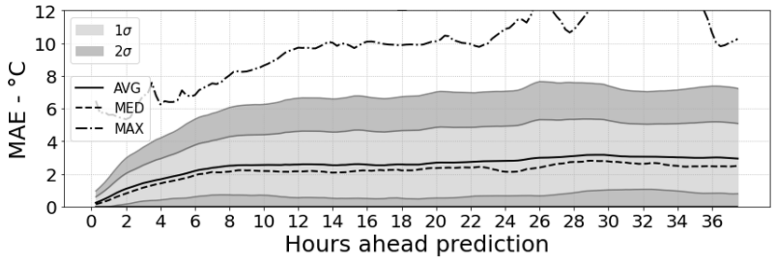
As shown in Table 4, not all neural networks performed as expected. Both the neural networks of the West Room environment multivariate case, for example, although they achieved over 30h of forecast horizon while maintaining the PMV value below the desired threshold, are not able to go beyond 10h when considering the MAE. When observing the shape of the graphs resulting from plotting the MAE and PMV metrics against the forecast horizon for the West Room multivariate case (Figures 20 and 21), the overall shapes of the MAE graphs are not too different from the previously analyzed Corridor univariate case. The median line also remains below the average line, showing how also the distribution of the prediction of the different outcomes for each forecasting horizon for these neural networks tends to be skewed towards lower, acceptable values, while the 1 standard deviation limits are also evenly distributed around the mean line, showing a consistent distribution in the prediction of the different outcomes for each forecasting horizon. However, these West Room multivariate graphs are much wider than the Corridor univariate ones, showing a greater variability in the difference between the $\hat{y}_t$ predicted values and the y_t real values. Furthermore, when zooming under the MAE acceptable threshold for the West Room multivariate case (Figure 21), and comparing the graphs to the ones of the Corridor univariate case (Figure 19), the graphs clearly show how the West Room multivariate MAE quickly grows beyond the set threshold, therefore indicating the neural networks' lack of acceptable accuracy.

Similar interesting considerations can be made by analyzing the graphs of the PMV values resulting from the West Room multivariate neural networks' applications. The West Room's multivariate 1D-CNN case shows a similar sinusoidal trend compared to the same neural network in the Corridor's univariate case. Unlike the Corridor's univariate case however, the West Room's multivariate 1D-CNN case does not appear to be diverging as the forecast horizon increases, while the amplitude of its oscillations are greater. The LSTM neural network for the West Room's multivariate, on the other hand, also shows a sinusoidal behaviour, unlike what shown by the same network type in the Corridor's univariate case. When zooming into the PMV's graphs and show their behaviour

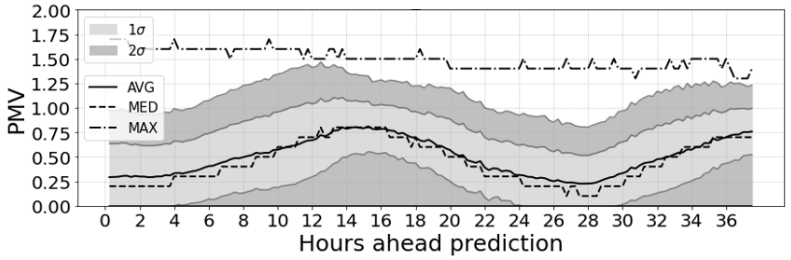
below the set PMV threshold, it is interesting to see how the 1D-CNN does indeed manage to remain below the set threshold for 32h of forecasting horizon, however it does not do so consistently. It actually grows above the set limit after 8h, similarly to when its MAE grows beyond its threshold, and remains above such threshold for 12h, before returning below the set 0.5 PMV at 20h and remaining there for another 12h until 32h of forecast horizon. The LSTM PMV value for the West Room multivariate case, on the other hand, shows a more linear behaviour, slightly decreasing as forecasting horizon increases, just like for the same network type in the Corridor univariate case, although with higher values.



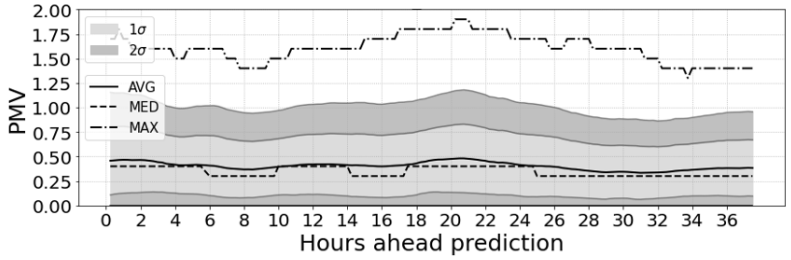
i) West Room temperature predicted by 1D-CNN multivariate original network with n_out = 150



ii) West Room temperature predicted by LSTM multivariate original network with n_out = 150

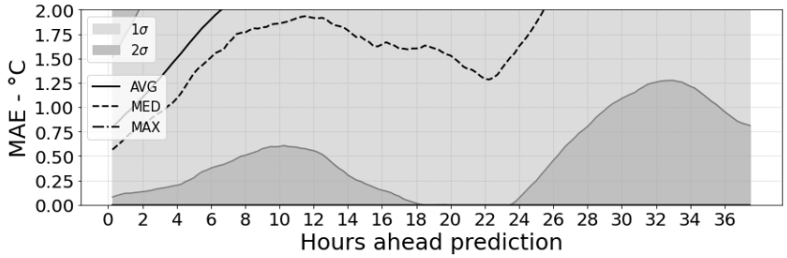


iii) PMV for West Room temperature predicted by 1D-CNN multivariate original network with n_out = 150

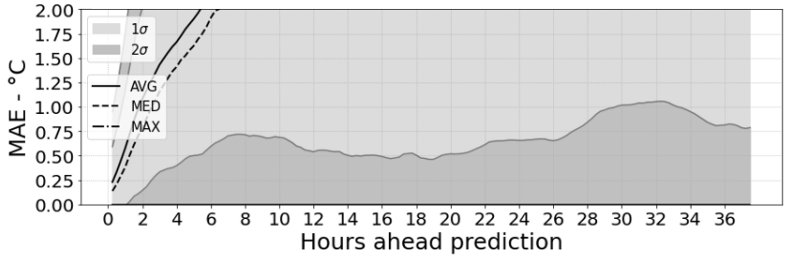


iv) PMV for West Room temperature predicted by LSTM multivariate original network with n_out = 150

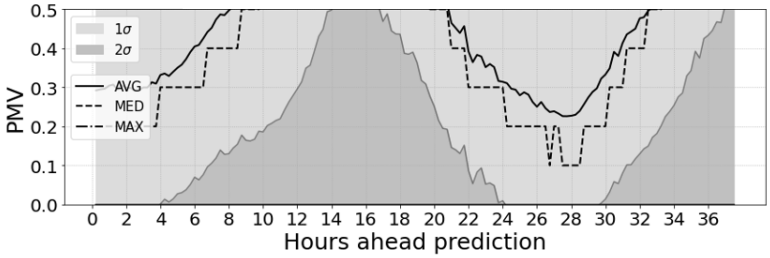
Figure 20: Performance of the West Room multivariate original neural networks before the application of transfer learning



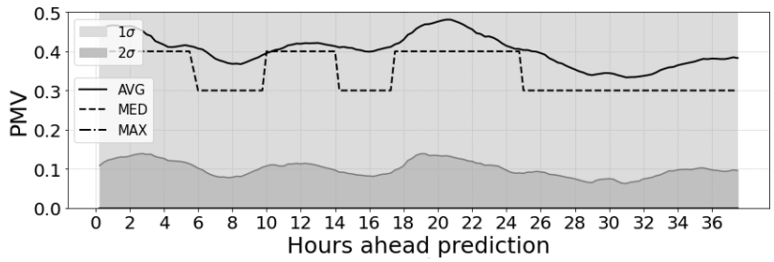
i) West Room temperature predicted by 1D-CNN multivariate original network with n_out = 150



ii) West Room temperature predicted by LSTM multivariate original network with n_out = 150



iii) PMV for West Room temperature predicted by 1D-CNN multivariate original network with n_out = 150



iv) PMV for West Room temperature predicted by LSTM multivariate original network with n_out = 150

Figure 21: Performance of the West Room multivariate original neural networks before the application of transfer learning - zoom on MAE and PMV thresholds

After having verified and analyzed the performance of the initial neural networks trained on the simulated datasets, the different transfer learning techniques are applied and their results further compared and analyzed. The results can be seen in Table 5.

Table 5: Performance of neural networks after the application of transfer learning

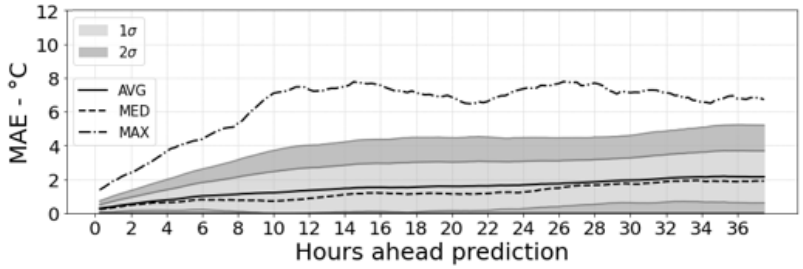
			ORIGINAL NEURAL NETWORKS		TRANSFER LEARNING WITH LAST LAYER RE-TRAINED		TRANSFER LEARNING WITH FIRST LAYER RE-TRAINED		TRANSFER LEARNING WITH ALL LAYERS RE-TRAINED	
Environment	Case	Network	Max forecast horizon (h) with MAE <= 2°C	Max forecast horizon (h) with PMV <= 0.5	Max forecast horizon (h) with MAE <= 2°C	Max forecast horizon (h) with PMV <= 0.5	Max forecast horizon (h) with MAE <= 2°C	Max forecast horizon (h) with PMV <= 0.5	Max forecast horizon (h) with MAE <= 2°C	Max forecast horizon (h) with PMV <= 0.5
Whole building	Univariate	1DCNN	32	32	36	36	36	34	23	36
		LSTM	35	36	36	25	36	25	36	32
	Multivariate	1DCNN	25	10	36	36	36	35	36	16
		LSTM	11	17	36	8	36	13	36	8
East Room	Univariate	1DCNN	36	34	36	34	36	34	36	36
		LSTM	27	36	36	20	36	18	36	22
	Multivariate	1DCNN	24	36	36	36	36	35	36	34
		LSTM	29	36	36	3	36	3	36	5
West Room	Univariate	1DCNN	28	36	33	34	28	36	14	36
		LSTM	29	36	34	36	36	36	36	36
	Multivariate	1DCNN	7	32	36	36	36	36	36	36
		LSTM	5	36	36	29	36	14	36	8
Corridor	Univariate	1DCNN	30	36	36	36	23	36	15	36
		LSTM	31	36	36	29	33	36	23	36
	Multivariate	1DCNN	25	35	19	36	36	36	36	36
		LSTM	28	36	36	25	36	23	36	24

The above results show how the application of transfer learning does improve the performance of the neural networks, but in a heterogeneous way. More specifically, the MAE sees an increase in forecast horizon while remaining within the set acceptable threshold in 79% of the cases, with an average increase in forecast horizon of 13.4 hours. The PMV, on the other hand, sees an increase in forecast horizon while remaining within the set acceptable threshold in 27% of the cases, with an average increase in forecast horizon of 8.6 hours. When considering each individual neural network, 10 out of the 16 original neural networks see an increase in forecast horizon for both MAE and PMV after the application of one of the transfer learning approaches. However, such increases are heterogeneous: there is no specific transfer learning approach which con-

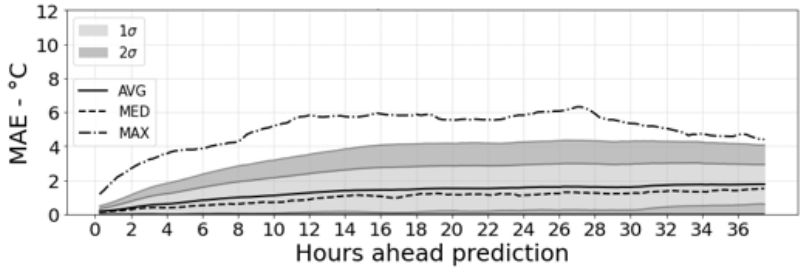
sistently improves the performance of all neural networks. Furthermore, there does not appear to be a pattern between the transfer learning approaches, the environment/case or network of the original neural network, and the increase in performance.

As in the analysis of the original neural networks before the application of transfer learning, in order to evaluate the effectiveness of the methodology the shape of the graphs resulting from plotting the MAE and PMV metrics against the forecast horizon must also be observed. Figures 22, 23, 24 and 25 show a positive example coming from the application of transfer learning on the 1D-CNN univariate neural networks for the Whole Building environment. As far as the MAE is concerned, the results show how the application of transfer learning allows the shape of the curve to compress, thus reducing the MAE and indicating an increase in the accuracy of the network, but only for 2 out of the 3 transfer learning techniques, when either the last layer or the first layer are retrained. When all layers are retrained, the MAE graph actually shows to worsen. This is further shown in Figures 24 and 25, which zooms under the MAE e PMV thresholds, and shows how the neural networks which apply transfer learning and retrain only the last or the first layer see an increase in the forecast horizon from 32h to 36h, whereas when all layers are retrained, the forecast horizon is reduced from 32h to 22h. The PMV graphs then provide other interesting insights. As in the analysis of the original neural networks before the application of transfer learning, also these PMV graphs show a sinusoidal pattern. However, whereas all three neural networks which applied transfer learning see a positive compression in the shape of the PMV graph, unlike in the MAE graphs the best improvement can be seen for the neural network which retrained all layers. This neural network sees a much lower amplitude in the sinusoidal pattern of the PMV graph, which remains on average at half the threshold set value and never passes it. Furthermore, also the 1 standard deviation shadow is shown to have compressed significantly, remaining almost completely below the threshold value too. As far as the graphs concerning the neural networks which apply the other transfer learning techniques, one can see how when only the first layer is

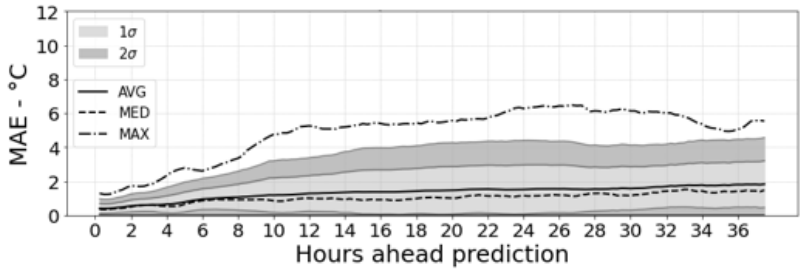
retrained the graph shows some improvement but still slightly surpasses the set threshold. On the other hand, when only the last layer is retrained the graph improves significantly, maintaining the PMV value well below threshold until the 36h of frecast horizon and also most of the 1 standard deviation area. The amplitude of the sinusoidal pattern in this neural network with the last layer retrained is wider compared to the one with all layers retrained, but it proves to be the most effective if one considers both PMV and MAE improvements.



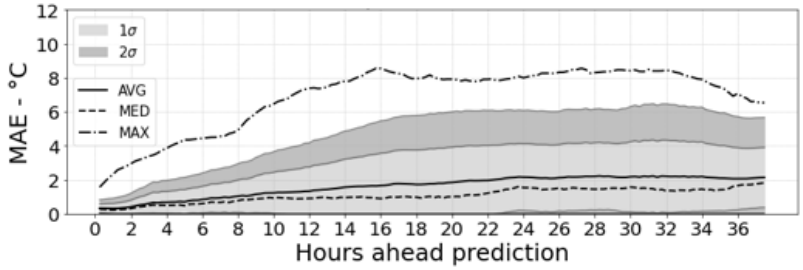
i) Building Average temperature predicted by 1D-CNN univariate original network with n_out = 150



ii) Building Average temperature predicted by 1D-CNN univariate new network with TL and last layer retrained with n_out = 150

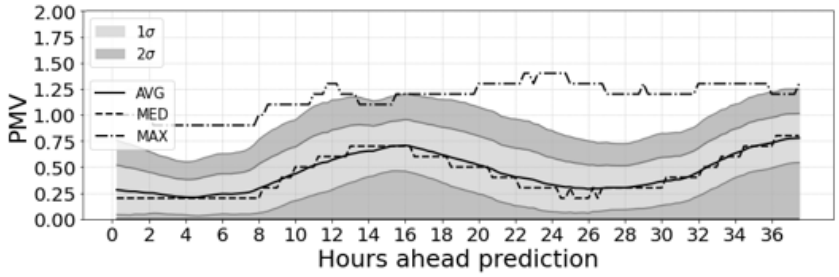


iii) Building Average temperature predicted by 1D-CNN univariate new network with TL and first layer retrained with n_out = 150

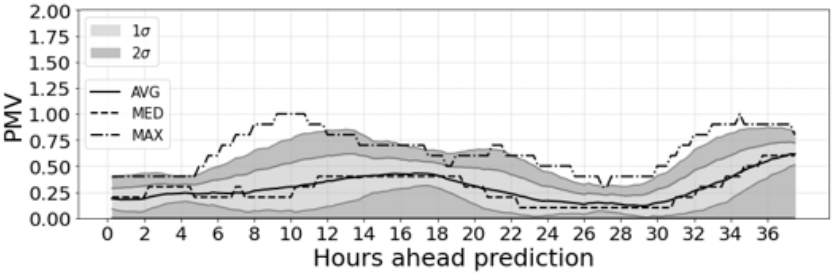


iv) Building Average temperature predicted by 1D-CNN univariate new network with TL and all layers retrained with n_out = 150

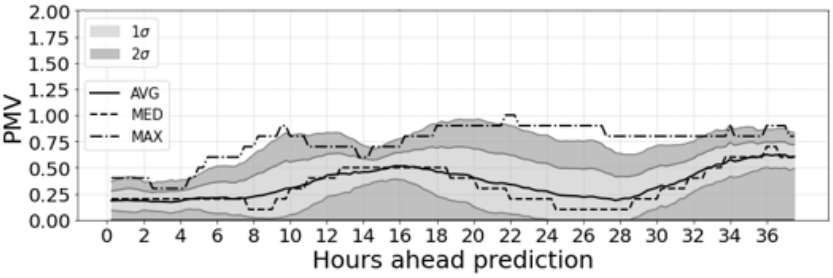
Figure 22: Comparison of the different 1D-CNN univariate neural networks for the Whole Building environment - MAE



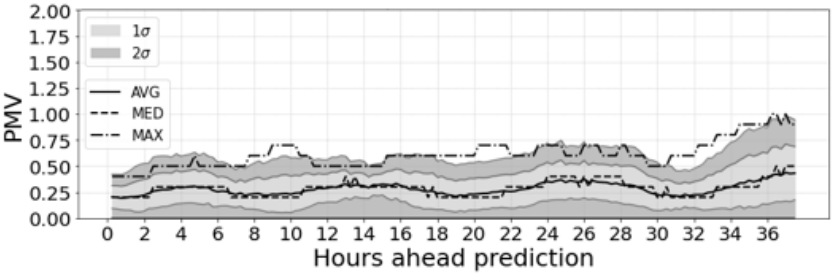
i) PMV for Building Average temperature predicted by 1D-CNN univariate original network with n_out = 150



ii) PMV for Building Average temperature predicted by 1D-CNN univariate new network with TL and last layer retrained with n_out = 150



iii) PMV for Building Average temperature predicted by 1D-CNN univariate new network with TL and first layer retrained with n_out = 150



iv) PMV for Building Average temperature predicted by 1D-CNN univariate new network with TL and all layers retrained with n_out = 150

Figure 23: Comparison of the different 1D-CNN univariate neural networks for the Whole Building environment - PMV

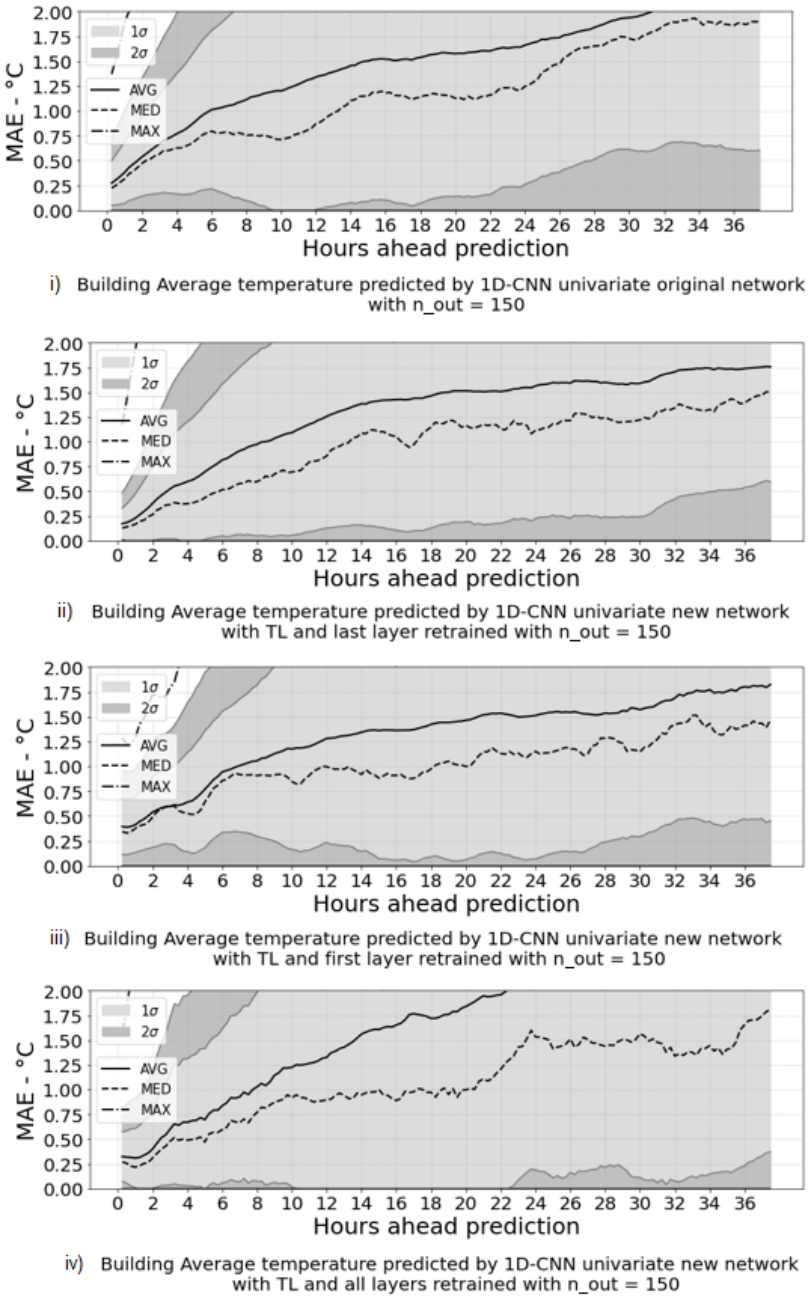
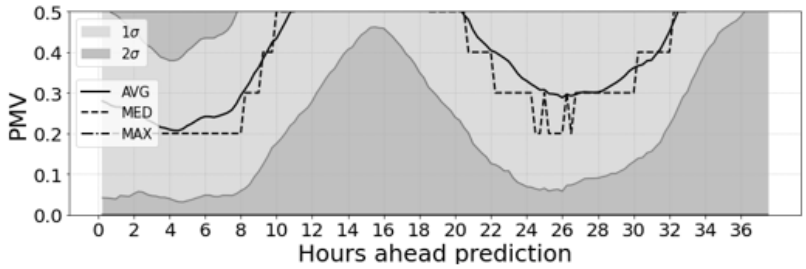
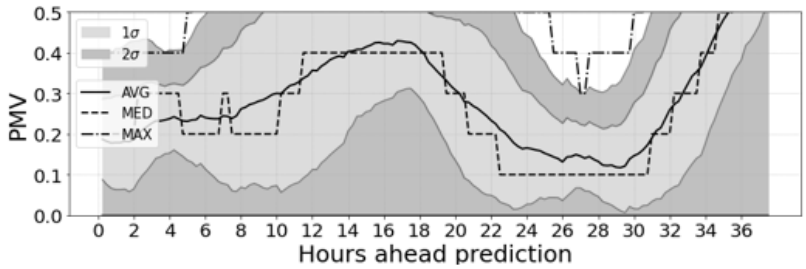


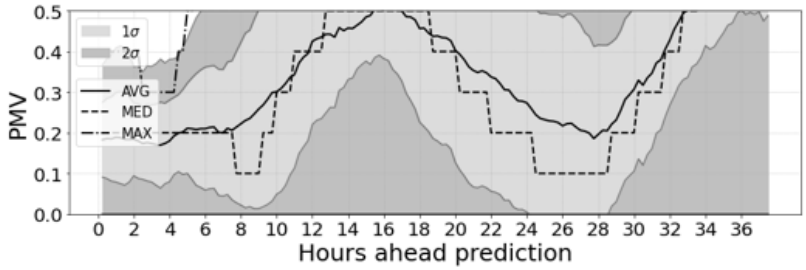
Figure 24: Comparison of the different 1D-CNN univariate neural networks for the Whole Building environment - zoom on MAE thresholds



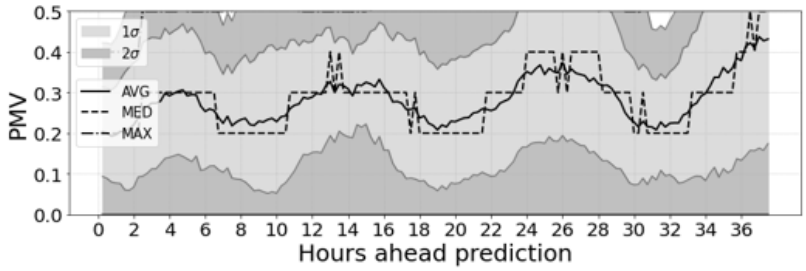
i) PMV for Building Average temperature predicted by 1D-CNN univariate original network with n_out = 150



ii) PMV for Building Average temperature predicted by 1D-CNN univariate new network with TL and last layer retrained with n_out = 150



iii) PMV for Building Average temperature predicted by 1D-CNN univariate new network with TL and first layer retrained with n_out = 150



iv) PMV for Building Average temperature predicted by 1D-CNN univariate new network with TL and all layers retrained with n_out = 150

Figure 25: Comparison of the different 1D-CNN univariate neural networks for the Whole Building environment - zoom on PMV thresholds

The results of the investigation therefore show how, following the transfer learning, the neural networks significantly increase their performance. In the less interesting cases, the forecast horizon capable of remaining within the MAE threshold increases to 36 hours or more. Most interestingly though, numerous cases showed how the MAE values tend to level out well below the MAE threshold. This allows for a greater forecasting horizon, while maintaining an acceptable indoor air-temperature prediction accuracy. However, such improvements were heterogeneous: there is no specific transfer learning approach which consistently improves the performance of all neural networks. Also, there is no apparent pattern between the transfer learning approaches, the environment/case or network of the original neural network, and the increase in performance.

4. Conclusions

The investigation's results show how 1D-CNN and LSTM can successfully be used for indoor air-temperature predictions even if trained on simulated datasets, and that their performance further improves through the application of transfer learning on real, but limited, datasets. More specifically, the MAE sees an increase in forecast horizon while remaining within the set acceptable threshold in 79% of the cases, with an average increase in forecast horizon of 13.4 hours. The PMV, on the other hand, sees an increase in forecast horizon while remaining within the set acceptable threshold in 27% of the cases, with an average increase in forecast horizon of 8.6 hours. When considering each individual neural network, 10 out of the 16 original neural networks see an increase in forecast horizon for both MAE and PMV after the application of one of the transfer learning approaches. However, such increases are heterogeneous: there is no specific transfer learning approach which consistently improves the performance of all neural networks. Furthermore, there does not appear to be a pattern between the transfer learning approaches, the environment/case or network of the original neural network, and the increase in performance.

As future work for this investigation, we plan to research possible appli-

cations in similar environments. The current investigation was carried out on a building, which for its structural characteristics can be a public or residential building. Future investigations could research the replicability of such a methodology on industrial buildings, where the structural characteristics and also the environmental factors are significantly different. Another future investigation could involve its applicability in a Smart Building context. The proposed solution can be tested in a Smart Building environment to evaluate possible integration in real-world building management systems, thus testing Demand/Response control strategies.

As discussed by Yu in [57], efficiency requirements of the digital age are one of many challenges encountered by power grids in recent years, and an increasing number of artificial intelligence-related and big-data-driven solutions are emerging to push forward smart-grid development. The proposed solution can thus provide an important contribution to the overall effort of reducing energy consumption, which is a hot topic in our contemporary society.

Abbreviations

The following abbreviations are used in this manuscript:

1D-CNN	One-Dimensional Convolutional Neural Network
ANN	Artificial Neural Networks
BIM	Building Information Modelling
CNN	Convolutional Neural Network
DR	Demand/Response
DSM	Demand Side Management
EU	European Union
IoT	Internet of Things
HVAC	Heating, Ventilation and Air Conditioning
LSTM	Long Short Term Memory
MAE	Mean Average Error

MLP-NARX	Multilayer Perceptron with Non-linear Autoregressive Exogenous
MLR	Multiple Linear Regression
NAR	Non-linear Autoregressive neural network
PMV	Predicted Mean Vote
PPD	Percentage of Person Dissatisfied
RNN	Recurrent Neural Networks
SG	Smart Grid
TMY	Typical Meteorological Year
VLSI	Very Large Scale Integration

References

[1] European Parliament, Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings, `https://eur-lex.europa.eu/eli/dir/2010/31/oj`, 2010. [Online; accessed 12-October-2020].

[2] E. Patti, A. Acquaviva, M. Jahn, F. Pramudianto, R. Tomasi, D. Rabourdin, J. Virgone, E. Macii, Event-driven user-centric middleware for energy-efficient buildings and public spaces, IEEE Systems Journal 10 (2016) 1137–1146.

[3] Z. Afroz, G. Shafiullah, T. Urmee, G. Higgins, Modeling techniques used in building hvac control systems: A review, Renewable and Sustainable Energy Reviews 83 (2018) 64–84.

[4] A. Deihimi, H. Showkati, Application of echo state networks in short-term electric load forecasting, Energy 39 (2012) 27–340.

[5] P. Siano, Demand response and smart grids—a survey, Renewable and Sustainable Energy Reviews 30 (2014) 461–478.

[6] J. L. Cremer, M. Pau, F. Ponci, A. Monti, Optimal scheduling of

heat pumps for power peak shaving and customers thermal comfort, in: SMARTGREENS, 2017, pp. 23–34. doi:10.5220/0006305800230034.

[7] M. Veselỳ, W. Zeiler, Personalized conditioning and its impact on thermal comfort and energy performance–a review, Renewable and Sustainable Energy Reviews 34 (2014) 401–408.

[8] F. G. Brundu, E. Patti, A. Osello, M. Del Giudice, N. Rapetti, A. Krylovskiy, M. Jahn, V. Verda, E. Guelpa, L. Rietto, A. Acquaviva, Iot software infrastructure for energy management and simulation in smart cities, IEEE Transactions on Industrial Informatics 13 (2017) 832–840.

[9] G. Bianchini, M. Casini, A. Vicino, D. Zarrilli, Demand-response in building heating systems: A model predictive control approach, Applied Energy 168 (2016) 159–170.

[10] A. Aliberti, L. Bottaccioli, G. Cirrincione, E. Macii, A. Acquaviva, E. Patti, Forecasting short-term solar radiation for photovoltaic energy predictions, in: SMARTGREENS, 2018, pp. 44–53. doi:10.5220/0006683600440053.

[11] J. A. Clarke, J. Hensen, Integrated building performance simulation: Progress, prospects and requirements, Building and Environment 91 (2015) 294–306.

[12] G. Marques, R. Pitarma, A cost-effective air quality supervision solution for enhanced living environments through the internet of things, Electronics 8 (2019) 170.

[13] H. Gao, C. Koch, Y. Wu, Building information modelling based building energy modelling: A review, Applied Energy 238 (2019) 320–343.

[14] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, Y. J. Huang, C. O. Pedersen, R. K. Strand, R. J. Liesen, D. E. Fisher, M. J. Witte, J. Glazer, Energyplus: creating a new-generation building energy simulation program, Energy and buildings 33 (2001) 319–331.

[15] S. Klein, W. Beckman, J. Mitchell, J. Duffie, N. Duffie, T. Freeman, J. Mitchell, J. Braun, B. Evans, J. Kummer, et al., Trnsys 16: A transient system simulation program: mathematical reference, Trnsys 5 (2007) 389–396. [DOI not available. Material available on numerous online sources, including the following link: http://web.mit.edu/parmstr/Public/Documentation/05-MathematicalReference.pdf. Accessed 03-October-2020.].

[16] U.S. Department of Energy's (DOE) Building Technologies Office (BTO), Energyplus: Energy simulation software for buildings, `https://ipo.lbl.gov/lbnl2118/`, 2020. [Online; accessed 15-September-2020].

[17] U.S. Department of Energy's (DOE) Building Technologies Office (BTO), Energyplus, `https://energyplus.net/`, 2020. [Online; accessed 15-September-2020].

[18] A. Castell, C. Solé, Design of latent heat storage systems using phase change materials (pcms), in: Advances in Thermal Energy Storage Systems, Elsevier, 2015, pp. 285–305. doi:`10.1533/9781782420965.2.285`.

[19] L. Bottaccioli, A. Aliberti, F. Ugliotti, E. Patti, A. Osello, E. Macii, A. Acquaviva, Building energy modelling and monitoring by integration of iot devices and building information models, in: 2017 IEEE 41st annual computer software and applications conference (COMPSAC), volume 1, IEEE, 2017, pp. 914–922. doi:`10.1109/COMPSAC.2017.75`.

[20] W. J. Cole, K. M. Powell, E. T. Hale, T. F. Edgar, Reduced-order residential home modeling for model predictive control, Energy and Buildings 74 (2014) 69–77.

[21] C. F. Reinhart, C. C. Davila, Urban building energy modeling–a review of a nascent field, Building and Environment 97 (2016) 196–202.

[22] M. Massano, E. Macii, E. Patti, A. Acquaviva, L. Bottaccioli, A grey-box model based on unscented kalman filter to estimate thermal dynamics in

buildings, in: 2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), IEEE, 2019, pp. 1–6. doi:10.1109/EEEIC.2019.8783974.

[23] P. Bacher, H. Madsen, Identifying suitable models for the heat dynamics of buildings, Energy and Buildings 43 (2011) 1511–1522.

[24] Y. Chen, Z. Tong, Y. Zheng, H. Samuelson, L. Norford, Transfer learning with deep neural networks for model predictive control of heating, cooling, and natural ventilation in smart buildings, Journal of Cleaner Production 254 (2020) 119866.

[25] D. Li, S. X.-D. Tan, E. H. Pacheco, M. Tirumala, Parameterized architecture-level dynamic thermal models for multicore microprocessors, ACM Transactions on Design Automation of Electronic Systems (TODAES) 15 (2010) 1–226.

[26] T. J. Eguia, S. X.-D. Tan, R. Shen, D. Li, E. H. Pacheco, M. Tirumala, L. Wang, General parameterized thermal modeling for high-performance microprocessor design, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 20 (2012) 211–224.

[27] A. E. Ruano, E. M. Crispim, E. Z. Conceiçao, M. M. J. Lúcio, Prediction of building's temperature using neural networks models, Energy and Buildings 38 (2006) 682–694.

[28] F. Mateo, J. J. Carrasco, A. Sellami, M. Millán-Giraldo, M. Domínguez, E. Olivas, Machine learning methods to forecast temperature in buildings, Expert Systems with Applications 40 (2013) 1061—1068.

[29] A. Aliberti, F. M. Ugliotti, L. Bottaccioli, G. Cirrincione, A. Osello, E. Macii, E. Patti, A. Acquaviva, Indoor air-temperature forecast for energy-efficient management in smart buildings, in: 2018 IEEE International Conference on Environment and Electrical Engineering and 2018

IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), IEEE, 2018, pp. 1–6. doi:10.1109/EEEIC.2018.8494382.

[30] C. Xu, H. Chen, J. Wang, Y. Guo, Y. Yuan, Improving prediction performance for indoor temperature in public buildings based on a novel deep learning method, Building and Environment 148 (2019) 128–135.

[31] E. Kamel, A. Javan-Khoshkholgh, N. Abumahfouz, S. Huang, X. Huang, A. Farajidavar, Y. Qiu, A case study of using multi-functional sensors to predict the indoor air temperature in classrooms, ASHRAE Transactions 126 (2020) 3–11. [DOI not available. Article available at the following link: https://www.thefreelibrary.com/A+Case+Study+of+Using+Multi-Functional+Sensors+to+Predict+the+Indoor...-a0627513870. Accessed 27-August-2020.].

[32] A. Aliberti, L. Bottaccioli, E. Macii, S. Di Cataldo, A. Acquaviva, E. Patti, A non-linear autoregressive model for indoor air-temperature predictions in smart buildings, Electronics 8 (2019) 979.

[33] J. Cifuentes Quintero, G. Marulanda, A. Bello, J. Reneses, Air temperature forecasting using machine learning techniques: A review, Energies 13 (2020) 4215.

[34] J. Connor, R. Martin, L. Atlas, Recurrent neural networks and robust time series prediction, IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council 5 (1994) 240–54.

[35] S. Alawadi, D. Mera, M. Fernández-Delgado, F. Alkhabbas, C. M. Olsson, P. Davidsson, A comparison of machine learning algorithms for forecasting indoor temperature in smart buildings, Energy Systems (2020) 1–17.

[36] Z. Jiang, Y. M. Lee, Deep transfer learning for thermal dynamics modeling in smart buildings, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 2033–2037. doi:10.1109/BigData47090.2019.9006306.

[37] N. Gao, W. Shao, M. S. Rahaman, J. Zhai, K. David, F. D. Salim, Transfer learning for thermal comfort prediction in multiple cities, Building and Environment 195 (2021) 107725.

[38] S. Xu, Y. Wang, Y. Wang, Z. O'Neill, Q. Zhu, One for many: Transfer learning for building hvac control, in: Proceedings of the 7th ACM international conference on systems for energy-efficient buildings, cities, and transportation, 2020, pp. 230–239. doi:10.1145/3408308.3427617.

[39] Z. Deng, Q. Chen, Reinforcement learning of occupant behavior model for cross-building transfer learning to various hvac control systems, Energy and Buildings 238 (2021) 110860.

[40] H. Tercan, A. Guajardo, J. Heinisch, T. Thiele, C. Hopmann, T. Meisen, Transfer-learning: Bridging the gap between real and simulation data for machine learning in injection molding, Procedia CIRP 72 (2018) 185–190.

[41] P. O. Fanger, Thermal comfort. Analysis and applications in environmental engineering, Copenhagen: Danish Technical Press., 1970. doi:10.1177/146642407209200337.

[42] M. Owen, R. American Society of Heating, A.-C. Engineers, 2009 ASHRAE Handbook: Fundamentals, 2009 Ashrae Handbook - Fundamentals, American Society of Heating, Refrigeration and Air-Conditioning Engineers, 2009.

[43] B. Olesen, International standards and the ergonomics of the thermal environment, Applied ergonomics 26 (1995) 293–302.

[44] S. Lawrence, C. Giles, A. Tsoi, A. Back, Face recognition: A convolutional neural-network approach, IEEE transactions on neural networks 8 (1997) 98–113.

[45] H. Kagaya, K. Aizawa, M. Ogawa, Food detection and recognition using convolutional neural network, in: Proceedings of the 22nd ACM interna-

tional conference on Multimedia, 2014, pp. 1085–1088. doi:10.13140/2.1.3082.1120.

[46] N. Kalchbrenner, E. Grefenstette, P. Blunsom, A convolutional neural network for modelling sentences, 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference 1 (2014).

[47] Y. Bengio, P. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council 5 (1994) 157–66.

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

[49] M. Del Giudice, A. Osello, A. Fonsati, D. De Luca, A. Musetti, F. Marchi, E. Patti, F. Brundu, A. Acquaviva, District data management, modelling and visualization via interoperability, Building simulation. San Francisco (2017). [DOI not available. Article available at the following link: https://iris.polito.it/handle/11583/2680251. Accessed 07-September-2020.].

[50] M. Grosso, S. Rinaudo, E. Patti, A. Acquaviva, An energy-autonomous wireless sensor network development platform, in: 2018 13th International Conference on Design Technology of Integrated Systems In Nanoscale Era (DTIS), 2018, pp. 1–6. doi:10.1109/DTIS.2018.8368569.

[51] ANSI/ASHRAE Standard 55, Thermal environmental conditions for human occupancy, https://www.techstreet.com/ashrae/standards/ashrae-55-2013?product_id=1868610, 2013. [ASHRAE 55-2013 standard available for purchase on this website. Purchased in March 2015.].

[52] C. C. Aggarwal, Neural networks and deep learning, volume 10, Springer, 2018. doi:10.1007/978-3-319-94463-0.

[53] D. M. Kline, Methods for multi-step time series forecasting neural networks, in: Neural networks in business forecasting, IGI Global, 2004, pp. 226–250. doi:10.4018/978-1-59140-176-6.ch012.

[54] S. Mehrkanoon, Deep shared representation learning for weather elements forecasting, Knowledge-Based Systems 179 (2019) 120–128.

[55] D. Soekhoe, P. Putten, A. Plaat, On the impact of data set size in transfer learning using deep neural networks, in: International symposium on intelligent data analysis, 2016, pp. 50–60. doi:10.1007/978-3-319-46349-0_5.

[56] R. Ribani, M. Marengoni, A survey of transfer learning for convolutional neural networks, in: 2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T), IEEE, 2019, pp. 47–57. doi:10.1109/SIBGRAPI-T.2019.00010.

[57] Y. Yu, Editorial for the special issue on smart grid and energy internet, 2020. doi:10.1016/j.eng.2020.07.008.