IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 9, 2021, accepted January 5, 2022, date of publication January 11, 2022, date of current version January 20, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3142174

A Generative Deep Learning Framework Across Time Series to Optimize the Energy Consumption of Air Conditioning Systems

RAKSHITHA GODAHEWA[1], CHANG DENG[1], ARNAUD PROUZEAU[2], AND CHRISTOPH BERGMEIR[1]

[1]Department of Data Science and AI, Faculty of Information Technology, Monash University, Melbourne, VIC 3800, Australia
[2]Inria & LaBRI (University of Bordeaux, CNRS, Bordeaux-INP), 33000 Bordeaux, France

Corresponding author: Rakshitha Godahewa (rakshitha.godahewa@monash.edu)

This work was supported by Honeywell International Inc.

ABSTRACT Working towards active buildings that fully integrate efficient demand management with renewable energy sources and storage, energy efficiency is an important step, as building inefficiencies cause energy wastage and increase energy-related expenses. Currently, static thermal setpoints are typically used to maintain the inside temperature of a building at a comfortable level irrespective of its occupancy. This paper introduces a deep learning framework that trains across time series to forecast the temperatures of a future period directly where a particular room is unoccupied and optimises the setpoints of the room. To the best of our knowledge, this is the first study to use a state-of-the-art deep learning method trained across series to accurately predict temperatures for the subsequent optimal control of room setpoints. In contrast to traditional forecasting approaches that build isolated models to predict each series, our framework uses global recurrent neural network models that are trained with a set of relatively short temperature series, allowing the models to learn cross-series information. The predicted temperatures were then used to define the optimal thermal setpoints to be used inside the room during the unoccupied periods. We evaluate the prediction accuracy of our deep learning framework against a set of state-of-the-art forecasting models and can outperform those by a large margin. Furthermore, we analyse the usage of our deep learning framework to optimise the energy consumption of an air conditioning system in a real-world scenario using temperature data from a university lecture theatre. Based on simulations, we show that our proposed framework can lead to savings of approximately 20% and 15%, respectively, compared to the traditional temperature control model that does not use optimisation techniques and a programmable thermostat.

INDEX TERMS Deep learning, energy optimisation, generative models, recurrent neural networks, temperature forecasting.

I. INTRODUCTION

A limited number of energy resources, rapid population growth, and increased energy consumption [1] make energy optimisation a pressing need for modern societies. In urban areas, more than 40% of the energy is consumed by buildings [2], where heating, ventilation, and air conditioning (HVAC) systems consume a large portion of energy in many commercial buildings [3].

To solve these issues in our way to a carbon-neutral future, active buildings integrate renewable energy technologies, heating, cooling and other sources related to building energy efficiency. They aim to provide a comfortable environment for occupants while optimally controlling building energy production, storage, and usage.

In this study, we focus on the energy usage component of the problem, particularly for HVAC systems. The purpose of an HVAC system is to maintain appropriate and comfortable thermal conditions inside a building. The common approach of operating an HVAC system is to use static thermal setpoints, where the inside temperature level is always maintained within a predefined temperature limit. However, this strategy is beneficial only during the occupied periods of a building. Maintaining the same temperature level during unoccupied periods may be unnecessary and lead to energy wastage.

The associate editor coordinating the review of this manuscript and approving it for publication was Joanna Kołodziej.

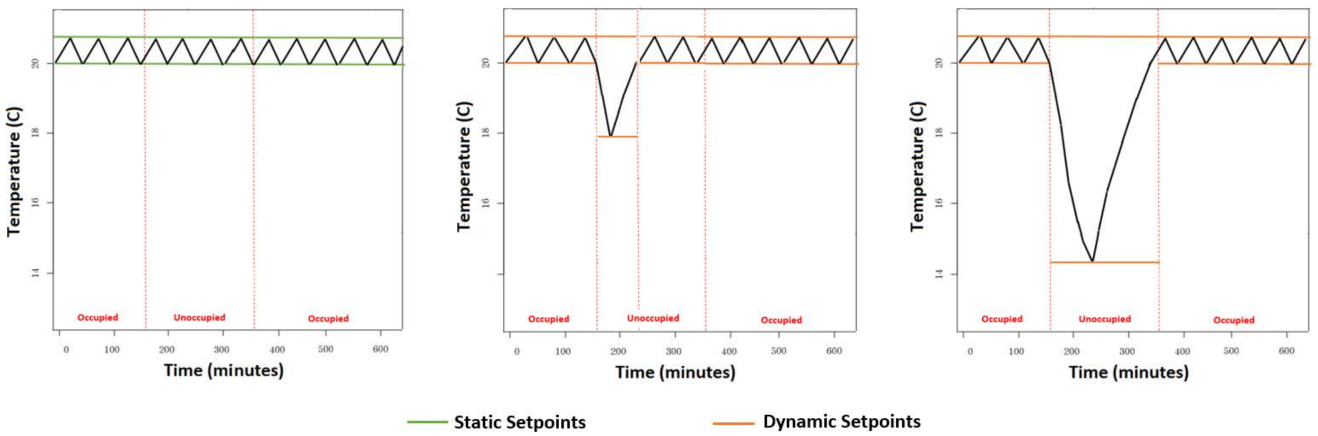


FIGURE 1. (left) Static thermal setpoints. The system always keeps the inside temperature in between two specific temperature levels irrespective of room occupancy. (middle) Dynamic thermal setpoints with a programmable thermostat. The temperature is allowed to drop to a certain level during unoccupied periods. However, the thermostat is always programmed to switch on before a fixed period of time ending the unoccupied period irrespective of the inside and outside temperatures. (right) Dynamic thermal setpoints with an optimal control. The temperature is allowed to drop during unoccupied periods and the HVAC is switched on optimally. As a result, large amounts of energy can be saved during periods where the room is unoccupied.

Dynamic thermal setpoints are a possible approach to control the inside temperature of a building more efficiently than static thermal setpoints. This approach can define different setpoints for the occupied and unoccupied periods, and hence be more energy efficient by allowing the inside temperature to be closer to the outside temperature during unoccupied periods. The concept of dynamic thermal setpoints has been used to optimise the behaviour of HVAC systems. Recent work in this area uses predictions of building management related factors such as occupancy and percentage of dissatisfied building occupants to dynamically change the setpoints of buildings [4], [5]. Adaptive programmable thermostats [6] also use occupation schedules to control HVAC systems; however, they are only programmed to switch on/off an HVAC system before a fixed period of time irrespective of the inside and outside temperatures; in many cases, this time point may not be the optimal time point to switch on/off the HVAC system. Thus, there is a possibility of energy wastage when programmable thermostats are used. Hence, to reduce the energy wastage, the setpoints should be optimally controlled. Fig. 1 illustrates the concepts of the static thermal setpoints, dynamic thermal setpoints with a programmable thermostat and dynamic thermal setpoints with an optimal control.

For the use of dynamic setpoints and for energy efficiency optimisation, it is highly beneficial to know in advance when rooms are occupied, and it is necessary to model and predict inside temperatures so that the HVAC system has sufficient time to return to a comfortable level at the time it is needed. Consequently, a body of literature is available on the forecasting of inside temperatures and energy consumption within buildings. In particular, several machine learning technologies have been used, such as *multiple linear regression (MLR)*, *support vector machines (SVM)*, *random forests (RF)* and *feed-forward neural networks (FFNN)* [7]–[13].

There is a recent trend in the forecasting community towards global forecasting models [14] that build a single model across many series, allowing the model to learn cross-series information. They have shown a huge potential in providing accurate forecasts compared to traditional univariate forecasting models such as the exponential smoothing state space model [ETS, 15] and autoregressive integrated moving average [ARIMA, 16], which build separate models to forecast each series in a more isolated way. In particular, many of the recently held forecasting competitions such as the M4 [17] and M5 competitions [18], have been won by global forecasting models. In our temperature forecasting scenario, the temperature series that show the heating or cooling behaviours inside a particular room are quite short. Thus, in our case, global forecasting models are more suitable for forecasting the inside temperatures as they train across multiple series, whereas they can learn the cross-series information. However, these temperature series have varying lengths. Furthermore, the forecasting model often has to address the cold-start problem, as explained in Section III. This phenomenon limits the use of state-of-the-art deep learning models such as attention-based schemes [19] and transformers [20] as the main building blocks of our framework. This motivates us to use recurrent neural networks [RNN, 21] as the underlying forecasting model of our framework in a generative manner.

An RNN is a special type of NN that is highly suitable for sequence modelling as it can address the temporal order and temporal dependencies of a sequence [22]. RNNs were incorporated into the winning method of the recently held M4 forecasting competition. RNNs are suitable for modelling in our scenario because of their capability to predict the inside temperatures corresponding to an unoccupied period of a room in a sequential manner. The current temperature depends on the previous temperature, and based on this, heating or cooling can occur. Thus, it is important to use a model such as the RNN in our case which can properly address the temporal order of the temperature series. RNNs can also address the cold-start problem, as explained in Section III. They have also been used to address real-world building management related forecasting problems [23]–[25].

Although researchers have considered RNNs to solve many real-world forecasting problems related to building management, to the best of our knowledge, there is no prior work on using global RNNs for indoor temperature forecasting, especially if these forecasts are then used to optimise the energy consumption of HVAC systems. We examine the merits of recent advances in the field of forecasting, namely the good performance of global forecasting models, RNNs and their applicability in such an optimisation system. Our study makes the following contributions.

1) We propose a global deep learning framework based on RNNs that can forecast the future indoor temperatures of a particular building as a function of the current inside and outside temperatures. In contrast to the traditional univariate forecasting models which build isolated models to predict each series, our proposed global deep learning framework trains across a collection of temperature series in a global manner, allowing the model to learn the cross-series information. Furthermore, the deep learning unit we use in our framework, RNNs, can properly handle the temporal order and temporal dependencies of the temperature series while extracting more useful information to predict future temperatures with a limited history. We compare the prediction accuracy of our global deep learning framework with a set of baseline machine learning benchmark models, namely, RF, SVM, MLR, and FFNN. We show that our global RNN-based framework is more accurate in forecasting temperatures than the benchmark models.
2) We then use the global RNN model to simulate the thermal behaviour inside a particular room. Based on this, we propose an optimisation approach to find the optimal time point to switch on the air conditioning system (AC) during an unoccupied period of that room such that it uses a minimal amount of energy to heat or cool the room. As the setpoints depend on the predicted temperatures, it is crucial for the system to have an accurate forecasting engine at its core.

We also present a quantitative comparison of the relative performance of the optimised global RNN model and the second most accurate model, an SVM model, against a traditional AC system and a programmable thermostat in a case study of a university lecture theatre. This lecture theatre does not have rooftop solar panels; therefore, controlling building energy efficiency using renewable energy sources is not applicable. Our proposed framework is suitable for optimising the energy efficiency of any building, particularly for a building, such as this lecture theatre which do not have renewable energy sources. We show that our proposed framework can achieve considerable energy savings in the lecture theatre compared with its traditional AC system as well as the programmable thermostat in terms of the amount of energy usage. In particular, relying on the global RNN model for predictions leads to an estimated 20% and 15% energy savings, respectively, with the traditional AC system and programmable thermostat according to our generative modelling. All implementations related to the framework are publicly available at: https://github.com/rakshitha123/TemperatureForecasting.

The remainder of this paper is organised as follows: Section II reviews related work. Section III describes our proposed temperature forecasting framework based on global RNNs and the setpoint optimisation procedure. Section IV summarises the experimental framework and model evaluation results based on prediction accuracy. Section V presents a case study using our proposed deep learning framework to optimise the energy consumption of an HVAC system in a real-world application. Finally, Section VI concludes the paper and discusses possible future research directions.

II. RELATED WORK

In this section, we review the existing state-of-the-art temperature and energy prediction models and HVAC optimisation techniques. The literature provides many examples of the use of machine learning technologies to forecast internal temperatures and energy consumption within buildings. As stated earlier, the popular machine learning models in this area are MLR, SVM, RF, and FFNN. Krüger and Givoni [26] proposed an MLR model to predict the internal thermal behaviour of three separate rooms. They compared the predicted results with the simulation of COMFIE software [7], where the simulated results were similar to the predictions. Kisi and Sanikhani [27] further proposed a similar MLR model to predict the inside temperatures in summer and winter separately.

Radhika and Shashi [8] used an approach based on SVM and FFNN to predict the maximum atmospheric temperature at a particular location. Paniagua-Tineo *et al.* [9] used an SVM to predict the daily maximum temperature which was then used to forecast the daily maximum energy consumption. Salcedo-Sanz *et al.* [10] established a model based on SVM and FFNN to predict the outside temperature at different locations in Australia and New Zealand. Their results showed that the SVM had better prediction accuracy than the FFNN. Furthermore, Jing *et al.* [28] used SVMs to predict the air pressure to facilitate air balancing in ventilation systems.

Researchers have also used RFs to predict energy expenditure in buildings. Ahmad *et al.* [11] compared FFNNs and RFs in predicting the energy consumption in buildings where both the FFNN and RF demonstrated similar performances. Touzani *et al.* [12] established an RF and a gradient boosted tree algorithm to predict the energy consumption in buildings where the gradient boosted tree algorithm achieves better prediction accuracy. Wang *et al.* [13] used an RF to predict the hourly electricity usage, with good results. Taheri and Razban [29] used FFNN, RF, SVM, and boosting techniques to predict CO_2 concentrations, where FFNN achieved better prediction accuracy. Taheri *et al.* [30] proposed a new stochastic planning framework for energy hubs.

A series of temperatures can also be considered as a sequence or time series. Therefore, it is possible to apply time series forecasting techniques to predict a set of future temperatures. In the forecasting community, there is a recent trend towards global forecasting models [14] that build a single model across many series, in contrast to the traditional univariate forecasting methods such as ETS and ARIMA which build one model per time series. Thus, global models are allowed to learn cross-series information unlike traditional univariate forecasting models. Many recently introduced deep learning models can be trained as global models, such as attention-based schemes [19], transformers [20], and Deep4cast [31], [32]. These novel deep learning methods are faster to train, easier to scale up, and thus more favourable for use than traditional deep learning models such as RNNs. However, in our temperature forecasting scenario, the temperature series are often very short and vary in length. Furthermore, the forecasting model must properly address the cold-start problem (Section III), and it is required to be a generative model. Thus, the applicability of popular deep learning frameworks is very limited in our scenario. On the other hand, RNNs are a particularly promising forecasting model. They recently contributed to the winning solution of the M4 forecasting competition [17]. RNNs have also obtained competitive results in real-world applications. For example, Bandara *et al.* [33] used RNNs for sales demand forecasting and outperformed statistical benchmarks and a production system. RNNs have also been used as generative forecasting models [34] and have provided promising results. Thus, we use RNNs as the underlying forecasting model of our framework.

RNNs have recently been used to address many building management related forecasting problems. Kreider *et al.* [35] used RNNs to predict building energy usage based on hourly data recorded at an engineering centre. Taheri *et al.* [36] proposed a deep RNN to automatically detect faults in HVAC systems, where the proposed method is more accurate in detecting faults compared to RF and gradient boosted trees. Furthermore, Kato *et al.* [23] proposed a heat load prediction approach using RNNs, where the method provided more accurate forecasts than a layered NN. Taheri *et al.* [37] proposed a deep RNN for medium and long-term forecasting of heating and electricity consumption which outperformed SVM and gradient boosted trees in terms of forecasting accuracy. RNNs have also been used for power forecasting [24], [38], [39], building occupancy forecasting [40], and energy consumption forecasting [41]. Although researchers have considered RNNs to solve many real-world forecasting problems related to building management, to the best of our knowledge, this is the first study to use globally trained RNNs for indoor temperature forecasting.

In the past, researchers have used the concept of dynamic thermal setpoints to optimise the behaviour of HVAC systems. Peng *et al.* [5] established an optimisation model to dynamically control the status of an HVAC system by predicting room occupancy. Their results showed that their system could save up to 21% of energy on average. However, the motion sensors used in the study which measured the motions of occupants were only operated in fixed 10 minute intervals where more energy could be saved by considering motion sensors which use other sampling methods and adaptive periods to measure the motions. Furthermore, those authors used only four occupancy patterns to train the occupancy prediction models, where the results could be improved by considering more occupancy patterns. Roussac *et al.* [42] evaluated two strategies for changing the temperature setpoints in office buildings: static intervention and dynamic intervention, where static intervention raises the setpoints by 1C and dynamic intervention involves load shifting. The results show that dynamic intervention has better performance, with a 6.3% reduction in energy consumption compared to static intervention. However, the static method was the most effective during the warmer summer when the dynamic method was not operational. Thus, there are issues with the comparison of static and dynamic intervention methods, where these two methods cannot be directly compared. Furthermore, for the static method, those authors have only considered a threshold of 1C when raising the setpoints where the results could be different for other thresholds, such as 2C or 3C. Similarly, Ward *et al.* [4] proposed an optimisation model to minimise the proposed cost function that can adjust the temperature setpoints in 15-minute intervals. The cost function contains the predicted percentage of dissatisfied building occupants (PDD), HVAC system power, electricity price, and CO_2 index. Their results demonstrated that their proposed optimisation model could reduce the energy consumption of HVAC systems by 10% - 15%. However, the uncertainty of the optimisation process is high, as multiple variables are predicted to optimise the cost function. Furthermore, the predictions are obtained for the next 24 hours in 5-minute intervals and that further increases the uncertainty of the method. The adaptive programmable thermostats [6] also use the concept of dynamic thermal setpoints to control the HVAC systems. The thermostats are programmed to switch on/off the HVAC system before a specific period of time ending an unoccupied period of a particular room, irrespective of the inside or outside temperatures. However, there is a high possibility that this time point may not be the optimal time point to switch on/off the HVAC system. Thus, there is a possibility of energy wastage when programmable thermostats are used.

The success of previous studies in reducing the energy consumption by using dynamic thermal setpoints encouraged us to use such an approach to optimise the HVAC system.

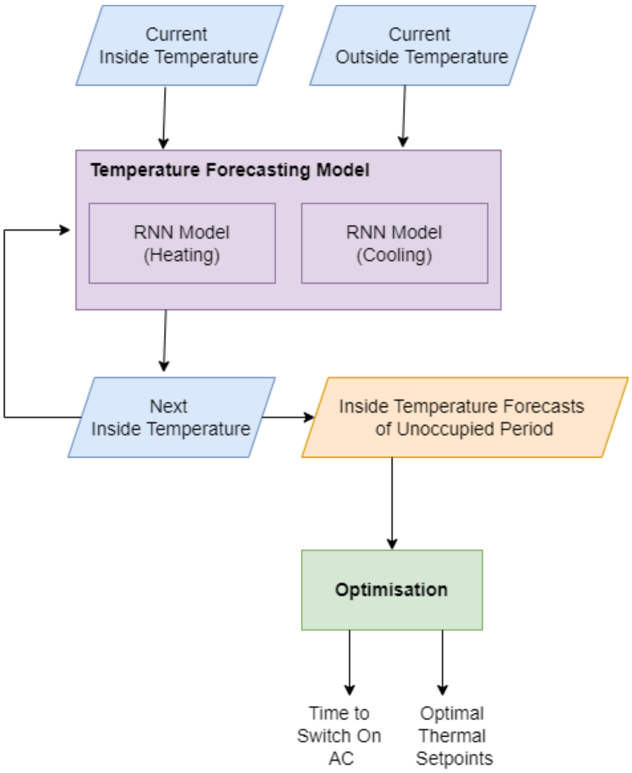


FIGURE 2. Overall process of setpoint optimisation during unoccupied periods.

As the literature lacks approaches to use indoor temperature forecasts to change the setpoints dynamically, we propose a global deep learning temperature forecasting model based on RNNs, where the temperature predictions subsequently enabled us to dynamically change the setpoints of particular rooms, particularly during unoccupied periods.

III. METHODOLOGY

The main goal of our research is to optimise the HVAC system by determining the optimal thermal setpoints to be used inside a particular room during unoccupied periods. Fig. 2 shows the overall procedure used to optimise the setpoints during unoccupied periods. Our framework first forecasts the future inside temperatures of a room during its unoccupied periods as a function of the current inside and outside temperatures using global RNNs, where separate RNNs are used to model the heating and cooling behaviours of the room. The predicted inside temperatures are fed back to the RNNs to predict the next inside temperature in a sequential manner, where this process is repeated until it finds the inside temperatures corresponding to the full unoccupied period. Our framework then uses these temperature forecasts to determine the optimal thermal setpoints and optimal time point to switch on the AC during the unoccupied period.

The details of our proposed prediction model and its optimisation procedure are described in the following.

A. PREDICTION MODEL

The overall system has four possible states: active heating or cooling, and passive heating or cooling. In winter, when the outside temperature is lower than the inside temperature, the room switches between the active heating and passive cooling states when the HVAC is switched on and off, respectively. In summer, the system switches between active cooling and passive heating.

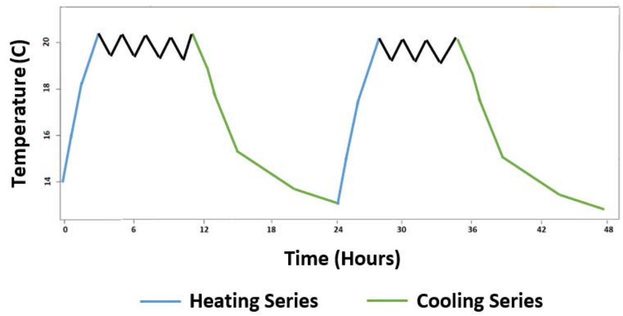


FIGURE 3. Heating and cooling parts of a temperature series. As these parts show a quite different behaviour, we use different models to fit them. The models need to be able to train on many relatively short heating or cooling series, respectively.

As the underlying mechanisms and the dynamics of the measured temperatures are quite different, we use a separate RNN model to forecast the inside temperatures for each of the four states. Each model is trained only with parts of the time series that correspond to their respective states. For example, the model for active heating is trained only with periods of active heating. For an illustration, see Fig. 3. Thus, the model trains effectively with many (relatively short) time series. Therefore, we train our models as global forecasting models that can learn across multiple time series during the training process [14]. In addition to the current inside and outside temperatures, we also provide future outside temperatures obtained from weather forecasts as inputs to the models.

Traditional univariate forecasting models would be of limited use in our setup, as we deal in many situations with a so-called cold-start problem, that is, predicting series with very short history or no history at all, which corresponds to predict directly after, for example, heating has been switched on in our application. In particular, our global RNN models predict one future inside temperature at a time, and feed the prediction back into the model to predict the next inside temperature. RNNs are especially suitable for modelling this situation, as they memorise the previous outputs using their feedback loops and use them when providing a new forecast. In this way, RNNs properly address the temporal order and temporal dependencies of inside temperatures; in this case, it is very important that the inside temperatures of an unoccupied period be predicted in a sequential manner. Other machine learning models such as FFNNs and (causal) convolutional networks require a potentially large number of inputs to produce accurate forecasts; hence, they are of limited use in this situation.

Note that our approach can also be considered a clustering approach, where two clusters of time series with different characteristics are used to train separate global RNN models, as presented by Bandara *et al.* [43].

1) RECURRENT NEURAL NETWORKS WITH LONG SHORT-TERM MEMORY CELLS

The Elman RNN unit [ERNN, 22], long short-term memory cell [LSTM, 44], and gated recurrent unit [GRU, 45] are some of the commonly used RNN units for time series forecasting. Out of those, we use LSTM in our work, due to its capability of capturing long-term dependencies while addressing vanishing/exploding gradient problems [46], and based on the recommendation given by Hewamalage *et al.* [21]. An LSTM cell contains two memory components, a short-term and a long-term memory component, which correspond to its two states: the hidden state and the cell state. Furthermore, an LSTM cell contains three gates: input, forget, and output gates. The input and forget gates determine the amount of past information to be saved in the current cell state, and the amount of information to be passed forward to future time points. In our model, we use an LSTM cell with peephole connections [47] which considers the previous cell state in the updating process of the input and forget gates, and is usually more accurate than the vanilla LSTM. For further details on LSTMs for forecasting, we refer to Hewamalage *et al.* [21].

2) RNN MODEL ARCHITECTURE

We use the stacked architecture for the RNN models, as suggested by Hewamalage *et al.* [21] and Bandara *et al.* [43]. Fig. 4 shows the unfolded version of an RNN over time with one hidden layer. In this case, each time step corresponds to an LSTM cell. In Fig. 4, we denote X_t as the input to the LSTM cell at time step t, Y_t as the output of the dense layer corresponding to each LSTM cell at time step t, and h_t and C_t as the hidden and cell states of the LSTM cell at time step t, respectively. Note that X_t and Y_t are vectors that can contain multiple data points. The feedback loops between LSTM cells support the model in carrying the states of the cells, namely hidden states and cell states, to future time points. In the stacked architecture, multiple layers can be placed on top of each other. The inputs are taken at the bottom layer, and the corresponding output is propagated to the next layer. The cell functions calculate the outputs based on current weights. There can be several hidden layers exist within the network. To convert the dimension of the cell output to the number of forecasts needed for the selected forecast horizon, the cell outputs are fed into a dense layer, a neural layer trained together with the RNN. The output of the final LSTM cell Y_n contains the expected forecasts of a particular time series. The cell dimension and number of hidden layers are externally tuned hyperparameters, as described in Section III-A3.

The model training process uses the errors of all the time steps until the end of the sequence to calculate the accumulated training error. Equations (1) and (2) represent the formulas for calculating the error per time step (e_t) and the final accumulated error (E), respectively, where Z_t is the actual output vector at time step t. The calculated accumulated error is then used with the backpropagation through time (BPTT) process to update the weights of the RNN cells.

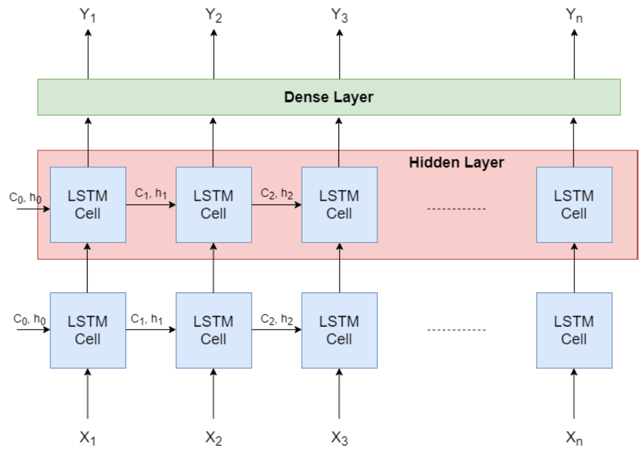


FIGURE 4. Stacked architecture used in our RNN model.

$$e_t = Z_t - Y_t \tag{1}$$

$$E = \sum_{t=1}^{T} e_t \tag{2}$$

Recent literature suggests the application of a moving window scheme to split the time series into a set of input and output vectors and use them to train the RNN model [21], [33]. However, as stated before, in our particular application we are frequently in the situation of a cold-start problem after switching states, so that we can effectively use only the current inside and outside temperatures as direct inputs into our model and not a window of lagged inputs. Therefore, instead of the input and output windows, we predict one future inside temperature at a time and feed the predictions back into the model iteratively until we have predicted the full unoccupied period.

3) HYPERPARAMETER TUNING

Each RNN model contains a set of hyperparameters that need to be tuned. For this purpose, we use the last value of each temperature series as a validation set. Then, we use the sequential model based algorithm configuration (SMAC) optimisation method [48] to automatically identify the optimal values for the hyperparameters given predefined ranges for hyperparameter selection. Table 1 lists the hyperparameters and the initial ranges used in our experiments.

4) LEARNING ALGORITHM

We use the continuous coin betting [COCOB, 49] algorithm as the learning algorithm. Unlike other algorithms, including Adam [50] and Adagrad [51], COCOB does not require the learning rate as a hyperparameter to train the models, because it automatically chooses an optimal learning rate. A recent

TABLE 1. Initial parameter ranges used with hyperparameter tuning of RNN model.

Parameter	Min. Value	Max. Value
Cell Dimension	10	15
Maximum Number of Epochs	2	25
Maximum Epoch Size	2	10
Mini-batch Size	1	15
Number of Hidden Layers	1	2
L2-Regularization Weights	0.0001	0.0008
Std. of Random Normal Initializer	0.0001	0.0008
Std. of Gaussian Noise	0.0001	0.0008

study by Hewamalage *et al.* [21] proposes using COCOB as the learning algorithm for RNN training processes in time series forecasting, based on experiments on six benchmark datasets.

This approach is inspired by a coin-betting scheme in which the outcome of a coin toss decides the amount of money required to maximise total wealth. COCOB applies the same idea to optimise a loss function where total wealth, coin toss, and bet correspond to the optimum point of the function, negative subgradient of the function at the bet point, and size of the step taken along the axis of the independent variable.

B. OPTIMISATION METHOD

We assume that the passive behaviour of a room is that the inside temperature strives towards the outside temperature in an exponentially decaying manner. That is, the closer the inside temperature is to the outside temperature, the longer it takes for the inside temperature to change more towards the outside temperature.

This assumption allows us to implement a straightforward optimisation procedure as follows: The best way to minimise the energy consumption of the HVAC system is to leave it in a switched-off state as long as possible until the inside temperature approaches the outside temperature. Only when the room is occupied again, there should be a comfortable inside temperature level for the occupants. Therefore, we need to be able to predict the passive change in the inside temperature and the time it takes for the HVAC system to return the temperature to a comfortable level before the occupation starts. In particular, we proceed as follows: As the first step, we predict the temperatures for the entire unoccupied period, assuming passive temperature behaviour during this period. We then predict the active temperature behaviour of the room using the respective prediction engine from different starting points. We start from the last time point of the unoccupied period and then produce a prediction successively going back one point at a time, until we reach a point from which the predicted temperature at the end of the unoccupied period is within the setpoints, if the HVAC is switched on at this point in time. The procedure is illustrated in Fig. 5 considering the passive cooling and active heating behaviours. Fig. 6 illustrates the overall setpoint optimisation procedure used with the temperature prediction models.

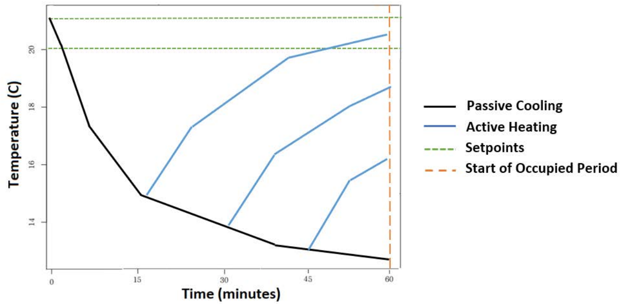


FIGURE 5. HVAC optimisation procedure for a 60 minutes unoccupied period considering passive cooling and active heating behaviours. According to the predictions, the HVAC system should be switched on 45 minutes before occupying the room to bring the inside temperature to a comfortable level.

IV. EXPERIMENTAL FRAMEWORK AND RESULTS

In this section, we present our experimental setup and the results on a real-world temperature dataset. This section describes the dataset, data preprocessing techniques, and benchmarks used for model evaluation.

A. DATASET

We use the temperature readings from a university lecture theatre in our experiments. The dataset contains 15-minutely readings between 27/06/2017 and 02/10/2017, 9408 rows in total; hence, a slight limitation is that we are only able to consider winter data for our experiments, even though our proposed framework is applicable to optimise HVAC systems during any season. Each row contains a timestamp, inside temperature, outside temperature (both measured in degrees Celsius), AC status (on/off), and setpoint. As an example, Fig. 7 shows the temperature behaviour on two selected winter days (Australian winter) along with the active (heating) and passive (cooling) periods between these days.

B. DATA PREPROCESSING

Our dataset contains an indicator variable that indicates when the AC system is switched on and off. However, in the exploratory data analysis, we determined that this indicator variable is relatively unreliable in the historical data. Therefore, we opted for a (semi-)manual preprocessing step to extract the active heating and passive cooling periods/series from the original temperature series. Fig. 8 shows an example of this (semi-)manual detection of active heating and passive cooling periods between June 28 and June 29 in the dataset. The black points in Fig. 8 indicate the starting and finishing points of the cooling and heating periods. For the heating section, we only extract the heating period when the temperature almost reaches the lower setpoint which is 20C in the example. For the cooling section, we select the entire period during which the AC system is completely switched off until it is switched on the next day.

We extract 68 cooling and 112 heating series from the lecture theatre temperature data, as described above. The corresponding outside temperature series are also extracted,

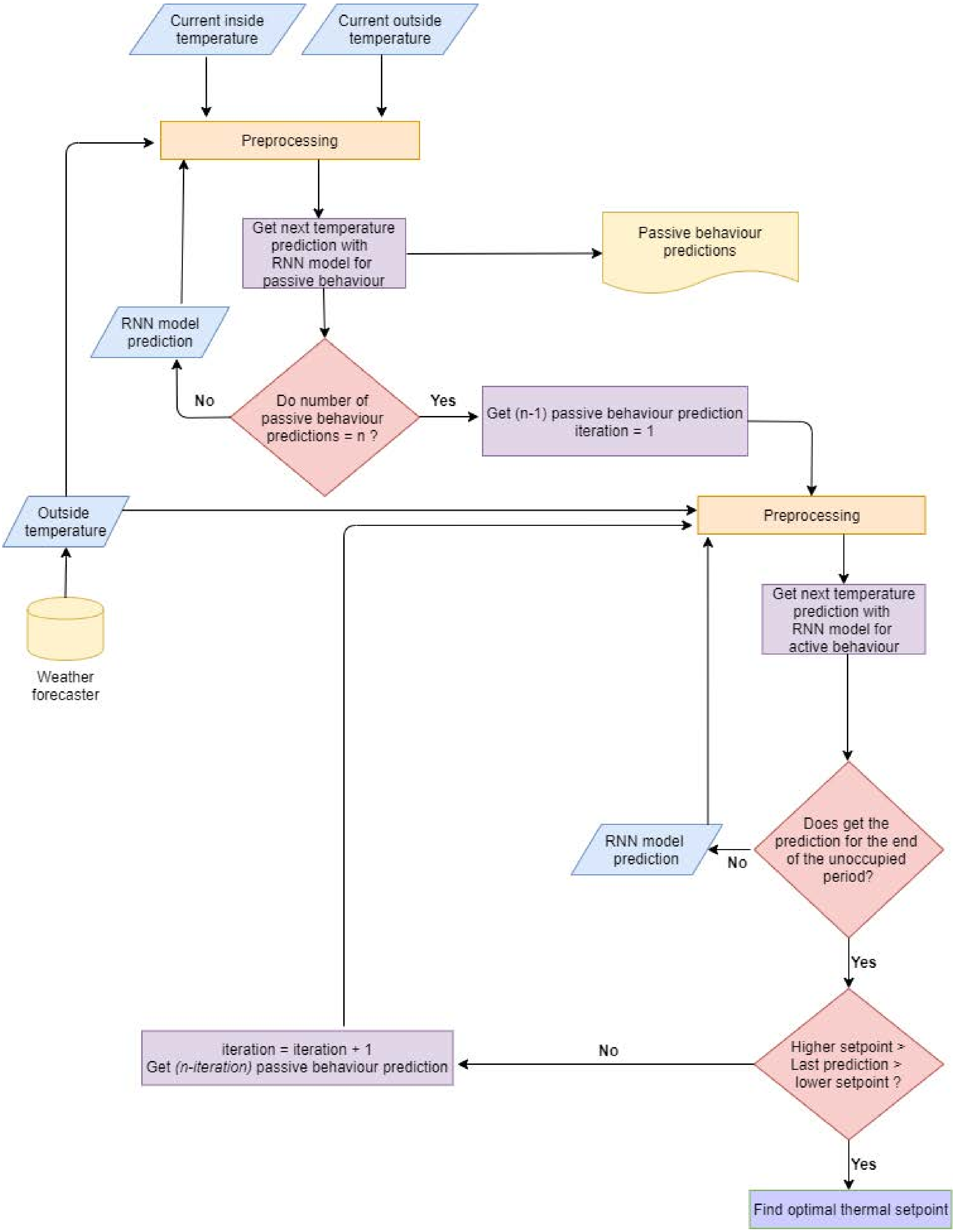


FIGURE 6. **Setpoint optimisation procedure using the prediction models. Here, *n* is the total predictions fit with the unoccupied period.**

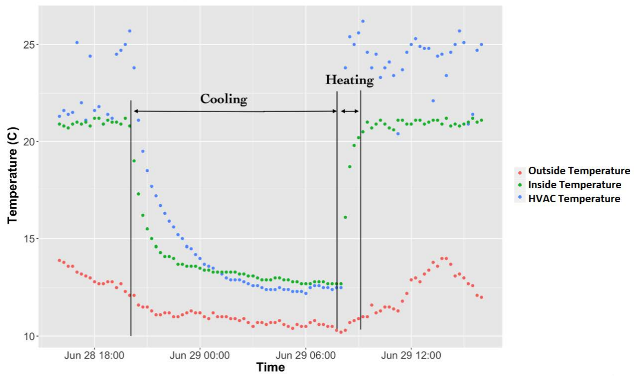


FIGURE 7. Passive cooling and active heating periods between June 28 and June 29. The lecture theatre cools down during the night when the HVAC is switched off, and then heats up in the morning when the HVAC is switched on.

TABLE 2. Heating and cooling series information.

Type	No: of Series	Min. Length	Max. Length
Cooling	68	21	51
Heating	112	6	6

as both the inside and outside temperatures are used to train the RNN models. The extracted heating and cooling series range from 8 to 22, and in order to bring all the temperatures to the same scale, mean scale normalisation is applied, where we normalise the data by dividing them using the average temperature during the daytime, which is 20 in our case. The purpose of data normalisation is to avoid the placement of RNN outputs within the saturated range [52].

Due to the iterative procedure and generative modelling throughout the entire unoccupied period, the models require the corresponding future outside temperatures when predicting the future inside temperatures. These future outside temperatures can be retrieved using a weather forecasting platform. For simplicity, in our experiments, we approximate these weather forecasts using the actual outside temperatures in the dataset. We note that this information is fed into all algorithms compared with each other in the same way, so that it does not alter their relative performance or the conclusions of our work.

We build separate global RNN models to model heating and cooling using the preprocessed heating and cooling series. Table 2 represents the number of series used to train each model, along with the maximum and minimum lengths of the extracted heating and cooling series. Table 3 lists the optimal parameter values we use to train the global RNNs that are obtained by the SMAC algorithm.

We use TensorFlow [53], an open-source deep learning platform, to implement all RNN models.

C. ERROR METRIC

In time series forecasting, many error metrics are commonly used, such as the symmetric mean absolute percentage error (SMAPE) and mean absolute scaled error [MASE, 54].

TABLE 3. Optimal parameters used to train the RNN models.

Parameter	Cooling RNN	Heating RNN
Cell Dimension	13	14
Maximum Number of Epochs	20	8
Maximum Epoch Size	8	6
Mini-batch Size	9	10
Number of Hidden Layers	1	1
L2-Regularization Weights	0.0006	0.0002
Std. of Random Normal Initializer	0.0002	0.0001
Std. of Gaussian Noise	0.0004	0.0004

However, the primary goal of most of these measures is to evaluate forecasts across series on different scales. However, in our case, all series are on the same scale, namely temperature in degrees Celsius, and hence, we refrain from using specialised forecasting error metrics. Instead, we use the root mean squared error (RMSE) and mean absolute error (MAE), as defined in (3) and (4), respectively.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(a_i - f_i)^2}{N}} \tag{3}$$

$$MAE = \frac{\sum_{i=1}^{N}|a_i - f_i|}{N} \tag{4}$$

In (3) and (4), N is the total number of temperature measurements, and a_i and f_i are the ith actual and predicted temperatures, respectively.

D. BENCHMARK MODELS

We use four state-of-the-art temperature prediction models, SVM, RF, MLR, and FFNN, as benchmarks against the RNN to predict the inside temperatures. The current inside and outside temperatures are used as inputs to these models to predict future inside temperatures in an iterative manner.

A grid-search approach is used to tune the hyperparameters of the models. We use 10-fold cross-validation with the training dataset to tune the hyperparameters. For the SVM, we choose a radial kernel function. The parameters *gamma* and C (cost) are varied from 1 to 5 with steps of 0.1, by cross-validation. In the heating section, the best value for C is 0.2, and the chosen value for *gamma* is 0.1. In the cooling section, the best values for C and *gamma* are 2 and 0.2, respectively. For the RF, we choose 500 for *ntree*, 30 for *ntime* and 3 for *mtry* in both the cooling and heating sections and change the node size from 1 to 20. For heating and cooling, the node sizes of 2 and 10 are selected, respectively, using the quantile regression computation algorithm. For the FFNN, the same normalisation method as for the RNN model, that is, mean scale normalisation is used. For the FFNN, we choose 0.1 for the threshold, 0.001 for the learning rate, and the learning algorithm is back-propagation in both the heating and cooling sections. We try different structures of layers and units with (0), (1), (2), (3), (4), (1,1), (1,2), (1,3), (2,3), (2,4), (1,1,2) and (1,2,2). The activation functions that we compare are sigmoid, tanh, and logistic. For heating, the best structure is 2-layers structure with two and three units, respectively, and

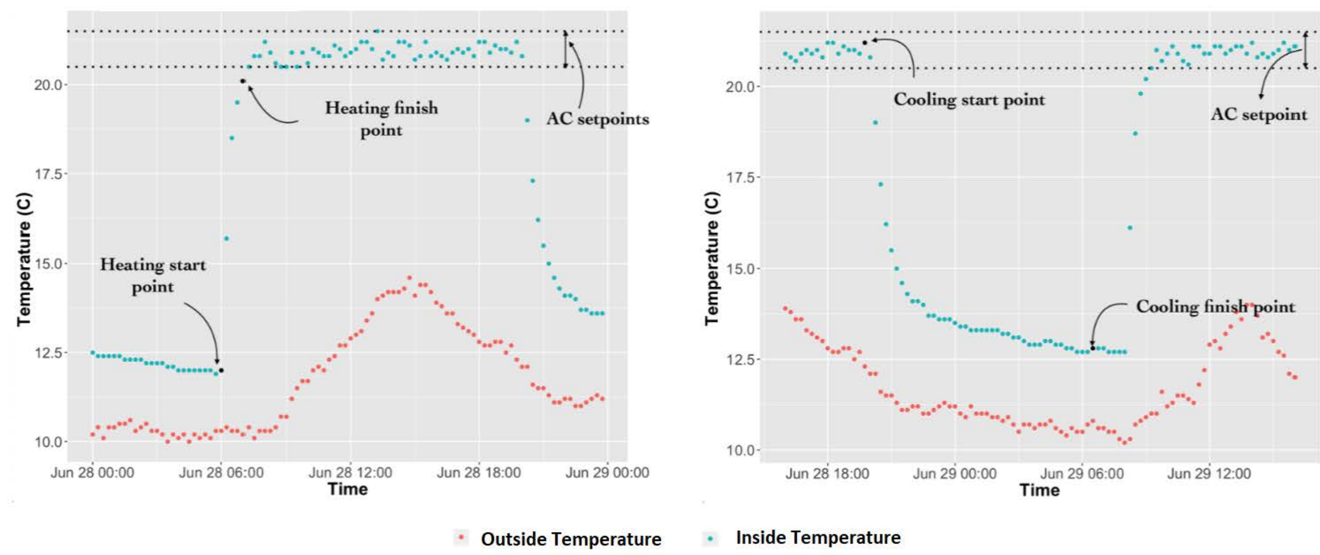


FIGURE 8. Detection of heating (left) and cooling (right) periods.

the activation function is tanh. For cooling, the best model has one hidden layer with the tanh activation function.

The setpoint optimisation is conducted for all benchmark models in the same way as the RNN model, as described in Section III-B, and the benchmark models also use the corresponding outside temperatures for training.

E. EVALUATION OF PREDICTION ACCURACY

We evaluate our global RNN model against the benchmark models based on the prediction accuracy. For this, we divide both the heating and cooling series into training and test sets, such that 80% of the series are used for training and 20% for testing. All the prediction models are then trained using the series in the training set. The forecasts are obtained from the trained models for each series in the test set by providing the start inside temperature and the corresponding outside temperatures of the series. Finally, the forecasts provided by the separate models are compared with the actual temperatures in the test set, and the RMSE and MAE are calculated for the entire test set.

Table 4 presents the RMSE and MAE values for the test set based on the predictions for both cooling and heating series with all considered models. From Table 4, it is clear that our proposed global RNN model is more accurate in forecasting indoor temperatures than the benchmark models in terms of both RMSE and MAE.

We further perform a nonparametric Friedman rank-sum test to evaluate the statistical significance of our results in the RMSE values. We then use Hochberg's post-hoc procedure [55] to characterise these differences, compared to the best-performing model. Table 5 shows the results of the statistical testing with the adjusted p-values calculated from the Friedman test with Hochberg's post-hoc procedure considering a significance level $\alpha = 0.05$. The p-value for the overall Friedman rank-sum test is 5.17×10^{-11}, that is highly significant. The proposed global RNN performs the best; hence, it is used as the control method, as mentioned in the first row. A horizontal line is used to separate the models that perform significantly worse than the RNN. All benchmark models report p_{Hoch} values less than α indicating that their performances are significantly worse than that of the RNN; hence, they are listed below the horizontal line.

TABLE 4. Comparison of RNN Model with SVM, RF, MLR and FFNN based on RMSE and MAE.

Model	RMSE	MAE
SVM	0.72	0.53
RF	0.75	0.52
MLR	1.08	0.77
FFNN	1.89	1.67
RNN	**0.68**	**0.48**

TABLE 5. Results of statistical testing.

Model	p_{Hoch}
RNN	-
FFNN	4.47×10^{-18}
MLR	1.09×10^{-5}
SVM	0.025
RF	0.039

V. CASE STUDY: USAGE OF OUR DEEP LEARNING FRAMEWORK TO OPTIMISE THE ENERGY CONSUMPTION OF HVAC SYSTEMS

In this section, we analyse the performance of our proposed global RNN-based framework in a real-world scenario of energy optimisation in an university lecture theatre. This lecture theatre does not have any rooftop solar panels, and thus, optimising the energy usage of its HVAC system using our proposed approach is more beneficial. We determine the extent to which the duration of heating periods can be reduced, as a proxy for energy savings, compared with a baseline of static thermal setpoints, a programmable thermostat, and the best-performing benchmark from Section IV-E, the SVM model.

A. BASELINE MODEL WITH STATIC THERMAL SETPOINTS

As a baseline model, we assume that the AC system of the lecture theatre uses static thermal setpoints to maintain the inside temperature at a comfortable level. Fig. 9 shows the generative modelling process with static thermal setpoints of 19C and 20C, where the model maintains the inside temperature between the two setpoints irrespective of room occupancy.

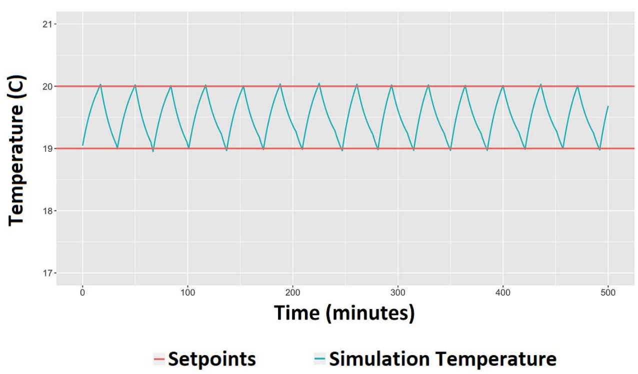


FIGURE 9. Generative modelling with static thermal setpoints.

B. PROGRAMMABLE THERMOSTAT

The thermostats are always programmed to be switched on the AC system before a fixed period of time occupying the room. With the generative modelling of the RNN and SVM models, we find the maximum period of time it takes to heat the room to the comfortable level after switching on the AC system. This time period is considered as the programmed time period of the thermostat for calculating the energy consumption savings.

C. OCCUPATION SCHEDULE

Fig. 10 shows the weekly schedule of the lecture theatre. Although the schedule may change from week to week, the chosen week has a typical occupation pattern, and we deem our conclusions drawn from this week as representative for a semester. We see from the schedule that there are unoccupied periods in the room on Wednesday and Thursday, as well as cases where occupation starts later or finishes earlier than usual. We can use this information in our models by reducing the heating time during an unoccupied period, by determining the optimal time points to switch on the AC in the morning, and by switching off the AC in the evening after occupation ends. We optimise our global RNN model and the most accurate benchmark, the SVM model as described in Section III-B.

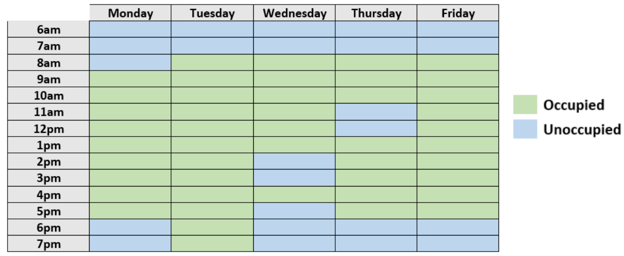


FIGURE 10. Weekly schedule of the lecture theatre in consideration.

D. COMPARISON OF RNN MODEL WITH SVM MODEL

We compare the models by calculating the reduction percentage of the heating time for all days in the schedule. Table 6 represents the total minutes (total operation time of the HVAC system) required by the models to heat the room with the baseline model using static thermal setpoints, the programmable thermostat, and optimised models according to the generative modelling of SVM and RNN, along with the percentages of the reduction of heating time given by the optimised models. According to Table 6, the SVM and RNN models predict an average of 12.79% and 20.26% heating reduction for the baseline with static thermal setpoints, and an average of 8.00% and 14.97% heating reduction for the programmable thermostat, respectively, over the week. As we know from Section IV, the RNN is significantly more accurate than the benchmarks, and its predicted savings are much higher than the savings from the benchmark model and programmable thermostat, overall this shows that using the RNN model is superior to using the benchmark and programmable thermostat.

Fig. 11 and 12 show the generative modelling of the optimised RNN and SVM models and the static baseline, as an example, for Wednesdays. We obtain the optimised curves of both models according to the procedure described in Section III-B. The behaviour of the optimised models within setpoints depends on their predictions. According to the generative modelling, the RNN and SVM models respectively allow the room to cool down until 13.81C and 16.33C during the unoccupied period and these temperatures can be identified as the optimal lower setpoints to be used during that period with respect to the optimised models.

For all days of the week, the generative modelling of both the SVM and RNN models predicts a considerable amount of energy saving in terms of heating time reduction compared to the generative modelling with the baseline model using static thermal setpoints of the lecture theatre in the morning as well as during the unoccupied periods in the afternoon. The generative modelling of the RNN predicts a higher amount of energy savings than the generative modelling of the SVM. The RNN model allows the room to cool down to a lower temperature during the unoccupied periods of the room and as a result, it predicts a higher energy saving compared to the SVM model on Wednesdays and Thursdays, according to our generative modelling. Furthermore, the generative modelling of the RNN model allows the room to start heating much

TABLE 6. Comparison of RNN model and SVM model with the baseline model and a programmable thermostat using current static thermal setpoints of the lecture theatre.

Day	SVM Model					RNN Model				
	Baseline Model	Thermostat	Optimised Model	Heating Reduction wrt Baseline	Heating Reduction wrt Thermostat	Baseline Model	Thermostat	Optimised Model	Heating Reduction wrt Baseline	Heating Reduction wrt Thermostat
Monday	359 mins	321 mins	283 mins	**21.17%**	**11.84%**	252 mins	234 mins	202 mins	**19.84%**	**13.68%**
Tuesday	434 mins	428 mins	394 mins	**9.22%**	**7.94%**	322 mins	307 mins	265 mins	**17.70%**	**13.68%**
Wednesday	563 mins	509 mins	464 mins	**17.58%**	**8.84%**	396 mins	365 mins	284 mins	**28.28%**	**22.19%**
Thursday	652 mins	635 mins	570 mins	**12.58%**	**10.24%**	467 mins	443 mins	380 mins	**18.63%**	**14.22%**
Friday	548 mins	524 mins	518 mins	**5.47%**	**1.15%**	365 mins	341 mins	306 mins	**16.16%**	**10.26%**
Average	**511.2 mins**	**483.4 mins**	**445.8 mins**	**12.79%**	**8.00%**	**360.4 mins**	**338 mins**	**287.4 mins**	**20.26%**	**14.97%**

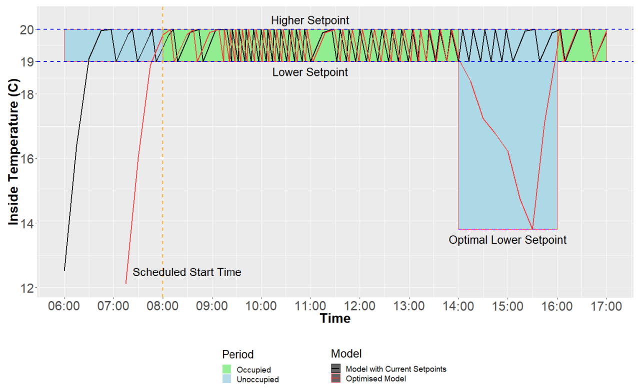


FIGURE 11. Generative modelling of the RNN on wednesday comparing the baseline and optimised models.

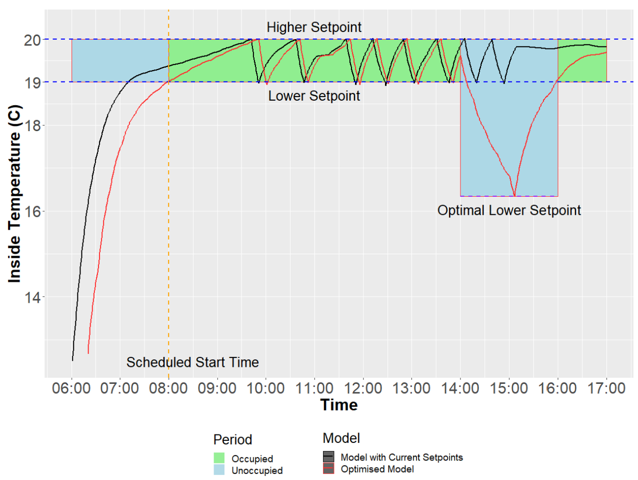


FIGURE 12. Generative modelling of the SVM on wednesday comparing the baseline and optimised models.

closer to the scheduled start time on each day compared to the generative modelling of the SVM model; and therefore, it predicts an energy saving of more than 16% on Monday, Tuesday, and Friday, even though the room does not contain any unoccupied periods in the afternoon.

These observations are very similar to those of the programmable thermostat. We consider that the thermostat is programmed to switch on the AC before 105 minutes occupying the lecture theatre for the generative modelling of SVM and RNN. This programmed time period is the maximum amount of time that the AC takes to heat the lecture theatre to the comfortable level during any day of the week according to the generative modelling of SVM and RNN. Thus, this maximum heating period is considered as the programmed time of the thermostat to ensure that the lecture theatre will be always at a comfortable level when it is occupied. Both SVM and RNN optimised models show a considerable amount of energy saving compared to the programmable thermostat. The HVAC operation time is minimal with our proposed RNN optimised model compared with both the static baseline and the programmable thermostat; thus, it is clear that our proposed framework is able to save more energy compared to them.

E. COMPUTATIONAL PERFORMANCE OF RNN MODEL

The simulations are run on an Intel(R) Core(TM) i7-8850 processor (2.6GHz) and 24GB of main memory.

Our proposed framework takes one minute to complete the training of RNN models, forecasting the inside temperatures, and optimising the setpoints for the unoccupied periods on Wednesday (2pm to 4pm) and Thursday (11am to 1pm). For temperature forecasting and optimisation, it only takes a few seconds and thus, in general, it is able to return the optimal thermal setpoints and the time that the AC should be switched on during a 2 hour unoccupied period within a few seconds.

VI. CONCLUSION AND FUTURE RESEARCH

Building energy efficiency is an important part of the work towards active buildings that fully integrate efficient demand management with renewable energy sources and storage. Building inefficiencies cause energy wastage and increase energy-related expenses. Currently, many buildings use static thermal setpoints to maintain their inside temperatures at a comfortable level, irrespective of building occupancy.

In this paper, we have proposed a global deep learning framework based on RNNs to predict the inside temperatures of a particular room. The predicted temperatures are then used to optimally control the room setpoints. To the best of our knowledge, we are the first to use this kind of approach for controlling setpoints. In contrast to the traditional univariate forecasting models that build isolated models to predict each series, our framework uses globally trained RNNs which are trained across a set of temperature series, allowing the models to learn cross-series information. Global RNNs are particularly useful in our setup compared to the popular deep learning models, such as transformer and attention-based schemes, due to their capability to address the cold-start problem and act as a generative model. Our global RNNs can learn across many relatively short varied-length time series, which allows us to train separate models on particular operation modes. We have compared the prediction accuracy of our global RNN-based framework with four state-of-the-art temperature prediction models based on SVM, RF, MLR, and FFNN. We have shown that our proposed framework is significantly more accurate in predicting indoor temperatures than the benchmark models. We have analysed the usage of our proposed global RNN temperature prediction model and the best-performing benchmark method in a real-world scenario by using them to optimise the energy consumption of the HVAC system of a lecture theatre. We have evaluated these models against the generative modelling of the current AC system of the lecture theatre and a programmable thermostat where our proposed model and the benchmark model lead to energy savings of approximately 20% and 13%, respectively, compared with the current AC system of the lecture theatre and 15% and 8%, respectively, compared with the programmable thermostat. Thus, our proposed model is more capable of saving energy by reducing the time required to heat the lecture theatre.

As our global RNN model provides more accurate temperature forecasts and predicts more energy savings, it can be identified as an appropriate method for predicting temperatures.

The success of this approach encourages as future work to build a global temperature prediction model which can predict future temperatures related to any room type. Different rooms have different characteristics such as size, occupancy, and number of windows. Therefore, developing a prediction model that can predict the temperatures belonging to any room type will be useful. Developing a temperature prediction model using ensemble mechanisms is another possible approach that would further increase the performance of our optimised global RNN model. Furthermore, our research is also applicable to similar room settings outside the university space, such as hotel conference rooms, libraries, and theatres, which highlights the potential commercial benefits of our research. Our model is designed for very broad applicability with the use of only the most standard data resources. As such, we are currently not considering data such as occupancy patterns, electricity prices and CO_2 measurements, that have been considered in the literature. As a future work, it will be worthwhile to acquire and use such data to compare these approaches with our proposed framework in different room settings.

ACKNOWLEDGMENT

The authors would like to thank Honeywell International Inc., for supporting this research, and the Buildings and Property Division, Monash University, Australia, for providing the temperature data for conducting the experimental study.

REFERENCES

[1] F. Mateo, J. J. Carrasco, A. Sellami, M. Millán-Giraldo, M. Domínguez, and E. Soria-Olivas, "Machine learning methods to forecast temperature in buildings," *Expert Syst. Appl.*, vol. 40, no. 4, pp. 1061–1068, Mar. 2013.
[2] A. M. Omer, "Energy, environment and sustainable development," *Renew. Sustain. Energy Rev.*, vol. 12, no. 9, pp. 2265–2300, 2008.
[3] A. Allouhi, Y. El Fouih, T. Kousksou, A. Jamil, Y. Zeraouli, and Y. Mourad, "Energy consumption and efficiency in buildings: Current status and future trends," *J. Cleaner Prod.*, vol. 109, pp. 118–130, Dec. 2015.
[4] J. K. Ward, J. Wall, S. West, and R. De Dear, "Beyond comfort managing the impact of HVAC control on the outside world," in *Proc. Windsor Conf.*, U.K., 2008.
[5] Y. Peng, A. Rysanek, Z. Nagy, and A. Schlüter, "Using machine learning techniques for occupancy-prediction-based cooling control in office buildings," *Appl. Energy*, vol. 211, pp. 1343–1358, Feb. 2018.
[6] J. Woolley, "Occupancy sensing adaptive thermostat controls–A market review and observations from multiple field installations in University residence halls," in *Proc. ACEEE Summer Study Energy Efficiency Buildings*, 2012, pp. 1–13.
[7] B. Peuportier and I. Blanc, "COMFIE: A software for passive solar design," in *Proc. IBPSA*, 1991, pp. 521–527.
[8] Y. Radhika and M. Shashi, "Atmospheric temperature prediction using support vector machines," *Int. J. Comput. Theory Eng.*, vol. 1, no. 1, pp. 55–58, 2009.
[9] A. Paniagua-Tineo, S. Salcedo-Sanz, C. Casanova-Mateo, E. G. Ortiz-García, M. A. Cony, and E. Hernández-Martín, "Prediction of daily maximum temperature using a support vector regression algorithm," *Renew. Energy*, vol. 36, no. 11, pp. 3054–3060, Nov. 2011.
[10] S. Salcedo-Sanz, R. C. Deo, L. Carro-Calvo, and B. Saavedra-Moreno, "Monthly prediction of air temperature in Australia and New Zealand with machine learning algorithms," *Theor. Appl. Climatol.*, vol. 125, nos. 1–2, pp. 13–25, Jul. 2016.
[11] M. W. Ahmad, M. Mourshed, and Y. Rezgui, "Trees vs neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption," *Energy Buildings*, vol. 147, pp. 77–89, Jul. 2017.
[12] S. Touzani, J. Granderson, and S. Fernandes, "Gradient boosting machine for modeling the energy consumption of commercial buildings," *Energy Buildings*, vol. 158, pp. 1533–1543, Jan. 2018.
[13] Z. Wang, Y. Wang, R. Zeng, R. S. Srinivasan, and S. Ahrentzen, "Random forest based hourly building energy prediction," *Energy Buildings*, vol. 171, pp. 11–25, Jul. 2018.
[14] T. Januschowski, J. Gasthaus, Y. Wang, D. Salinas, V. Flunkert, M. Bohlke-Schneider, and L. Callot, "Criteria for classifying forecasting methods," *Int. J. Forecasting*, vol. 36, no. 1, pp. 167–177, Jan. 2020.
[15] R. J. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting with Exponential Smoothing: The State Space Approach*. Berlin, Germany: Springer, 2008.
[16] G. E. P. Box and G. Jenkins, *Time Series Analysis, Forecasting and Control*. San Francisco, CA, USA: HoldenDay, 1990.
[17] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "The M4 competition: Results, findings, conclusion and way forward," *Int. J. Forecasting*, vol. 34, no. 4, pp. 802–808, Oct. 2018.
[18] S. Makridakis, E. Spiliotis, and V. Assimakopoulo, "The M5 accuracy competition: Results, findings and conclusions," *Int. J. Forecasting*, Jan. 2022.
[19] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, 2015, pp. 1–15.
[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, vol. 30, 2017, pp. 5998–6008.
[21] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *Int. J. Forecasting*, vol. 37, no. 1, pp. 388–427, 2021.
[22] J. L. Elman, "Finding structure in time," *Cognit. Sci.*, vol. 14, no. 2, pp. 179–211, Mar. 1990.
[23] K. Kato, M. Sakawa, K. Ishimaru, S. Ushiro, and T. Shibano, "Heat load prediction through recurrent neural network in district heating and cooling systems," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2008, pp. 1401–1406.
[24] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Trans. Energy Convers.*, vol. 21, no. 1, pp. 273–284, Mar. 2006.

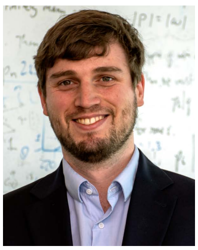
[25] X. Liu, H. Zhang, Y. Niu, D. Zeng, J. Liu, X. Kong, and K. Y. Lee, ''Modeling of an ultra-supercritical boiler-turbine system with stacked denoising auto-encoder and long short-term memory network,'' *Inf. Sci.*, vol. 525, pp. 134–152, Jul. 2020.
[26] E. Krüger and B. Givoni, ''Predicting thermal performance in occupied dwellings,'' *Energy Buildings*, vol. 36, no. 3, pp. 301–307, Mar. 2004.
[27] O. Kisi and H. Sanikhani, ''Modelling long-term monthly temperatures by several data-driven methods using geographical inputs,'' *Int. J. Climatol.*, vol. 35, no. 13, pp. 3834–3846, Nov. 2015.
[28] G. Jing, W. Cai, H. Chen, D. Zhai, C. Cui, and X. Yin, ''An air balancing method using support vector machine for a ventilation system,'' *Building Environ.*, vol. 143, pp. 487–495, Oct. 2018.
[29] S. Taheri and A. Razban, ''Learning-based CO_2 concentration prediction: Application to indoor air quality control using demand-controlled ventilation,'' *Building Environ.*, vol. 205, Nov. 2021, Art. no. 108164.
[30] S. Taheri, R. Ghoraani, A. Pasban, M. Moeini-Aghtaie, and A. Safdarian, ''Stochastic framework for planning studies of energy systems: A case of EHs,'' *IET Renew. Power Gener.*, vol. 14, no. 3, pp. 435–444, Feb. 2020.
[31] T. Bischoff and A. Gross, ''Wavenet+Dropout: An efficient setup for competitive forecasts at scale,'' in *Proc. ISF*, 2019.
[32] T. Bischoff, A. Gross, and K. Tran. (2019). *Deep4cast: Forecasting for Decision Making Under Uncertainty*. [Online]. Available: https://github.com/MSRDL/Deep4Cast
[33] K. Bandara, P. Shi, C. Bergmeir, H. Hewamalage, Q. Tran, and B. Seaman, ''Sales demand forecast in e-commerce using a long short-term memory neural network methodology,'' in *Proc. ICONIP*, 2019, pp. 462–474.
[34] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, ''DeepAR: Probabilistic forecasting with autoregressive recurrent networks,'' *Int. J. Forecasting*, vol. 36, no. 3, pp. 1181–1191, Jul. 2020.
[35] J. F. Kreider, D. E. Claridge, P. Curtiss, R. Dodier, J. S. Haberl, and M. Krarti, ''Building energy use prediction and system identification using recurrent neural networks,'' *J. Sol. Energy Eng.*, vol. 117, no. 3, pp. 161–166, Aug. 1995.
[36] S. Taheri, A. Ahmadi, B. Mohammadi-Ivatloo, and S. Asadi, ''Fault detection diagnostic for HVAC systems via deep learning algorithms,'' *Energy Buildings*, vol. 250, Nov. 2021, Art. no. 111275.
[37] S. Taheri, M. Jooshaki, and M. Moeini-Aghtaie, ''Long-term planning of integrated local energy systems using deep learning algorithms,'' *Int. J. Electr. Power Energy Syst.*, vol. 129, Jul. 2021, Art. no. 106855.
[38] S. M. Kelo and S. V. Dudul, ''Short-term Maharashtra state electrical power load prediction with special emphasis on seasonal changes using a novel focused time lagged recurrent neural network based on time delay neural network model,'' *Expert Syst. Appl.*, vol. 38, no. 3, pp. 1554–1564, Mar. 2011.
[39] M. Zhou, B. Wang, S. Guo, and J. Watada, ''Multi-objective prediction intervals for wind power forecast based on deep neural networks,'' *Inf. Sci.*, vol. 550, pp. 207–220, Mar. 2021.
[40] W. Wang, J. Chen, T. Hong, and N. Zhu, ''Occupancy prediction through Markov based feedback recurrent neural network (M-FRNN) algorithm with WiFi probe technology,'' *Building Environ.*, vol. 138, pp. 160–170, Jun. 2018.
[41] L. G. B. Ruiz, R. Rueda, M. P. Cuéllar, and M. C. Pegalajar, ''Energy consumption forecasting based on Elman neural networks with evolutive optimization,'' *Expert Syst. Appl.*, vol. 92, pp. 380–389, Feb. 2018.
[42] A. C. Roussac, J. Steinfeld, and R. de Dear, ''A preliminary evaluation of two strategies for raising indoor air temperature setpoints in office buildings,'' *Architectural Sci. Rev.*, vol. 54, no. 2, pp. 148–156, May 2011.
[43] K. Bandara, C. Bergmeir, and S. Smyl, ''Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach,'' *Expert Syst. Appl.*, vol. 140, Feb. 2020, Art. no. 112896.
[44] S. Hochreiter and J. Schmidhuber, ''Long short-term memory,'' *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[45] K. Cho, B. V. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, ''Learning phrase representations using RNN encoder-decoder for statistical machine translation,'' in *Proc. EMNLP*, 2014, pp. 1724–1734.
[46] Y. Bengio, P. Simard, and P. Frasconi, ''Learning long-term dependencies with gradient descent is difficult,'' *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
[47] F. A. Gers, J. Schmidhuber, and F. Cummins, ''Learning to forget: Continual prediction with LSTM,'' *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000.
[48] F. Hutter, H. H. Hoos, and K. Leyton-Brown, ''Sequential model-based optimization for general algorithm configuration,'' in *Proc. LION*, 2011, pp. 507–523.
[49] F. Orabona and T. Tommasi, ''Training deep networks without learning rates through coin betting,'' in *Proc. NeurIPS*, vol. 30, 2017, pp. 2160–2170.
[50] D. P. Kingma and J. Ba, ''Adam: A method for stochastic optimization,'' in *Proc. ICLR*, 2015, pp. 1–15.
[51] J. Duchi, E. Hazan, and Y. Singer, ''Adaptive subgradient methods for online learning and stochastic optimization,'' *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Feb. 2011.
[52] S. Smyl and K. Kuber, ''Data preprocessing and augmentation for multiple short time series forecasting with recurrent neural networks,'' in *Proc. ISF*, 2016, pp. 1–13.
[53] M. Abadi. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [Online]. Available: https://www.tensorflow.org/
[54] R. J. Hyndman and A. B. Koehler, ''Another look at measures of forecast accuracy,'' *Int. J. Forecasting*, vol. 22, no. 4, pp. 679–688, Oct. 2006.
[55] S. García, A. Fernández, J. Luengo, and F. Herrera, ''Advanced non-parametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power,'' *Inf. Sci.*, vol. 180, no. 10, pp. 2044–2064, May 2010.

RAKSHITHA GODAHEWA received the B.Sc. degree in computer science and engineering from the University of Moratuwa, Sri Lanka, in 2016. She is currently pursuing the Ph.D. degree with the Faculty of Information Technology, Monash University, Australia.

Her research interests include global time series forecasting models, machine learning, and deep learning.

CHANG DENG received the B.Sc. degree in electrical engineering from the University of Technology Sydney, Australia, in 2017, and the M.Sc. degree in data science from Monash University, Australia, in 2020.

His research interests include machine learning, deep learning, and energy optimization.



ARNAUD PROUZEAU received the M.Sc. degree in human–computer interaction from University Paul Sabatier, France, in 2014, and the Ph.D. degree in human–computer interaction from Paris-Sud University, France, in 2017.

From 2018 to 2020, he was a Postdoctoral Researcher with the Human–Computer Interaction Group, Faculty of Information Technology, Monash University, Australia. He is currently a Researcher in human–computer interaction and visualization at Inria, Bordeaux, France. His research interests include interaction, immersive technologies, and visualization.

CHRISTOPH BERGMEIR received the M.Sc. degree in soft computing and intelligent systems and the Ph.D. degree in computer science and information technology from the University of Granada, Spain, in 2009 and 2013, respectively.

He is currently an ARC DECRA Fellow with the Faculty of Information Technology, Monash University, Australia. His research interests include time series forecasting, recurrent neural networks, feature extraction, and ensembling.