https://doi.org/10.1016/j.apenergy.2021.117238

Attention-based Interpretable Neural Network for Building Cooling Load Prediction

Ao Li[1], Fu Xiao[1, 2*], Chong Zhang[1], Cheng Fan[1,3]

[1] Department of Building Services Engineering, The Hong Kong Polytechnic University, Hong Kong, China

[2] Research Institute for Smart Energy, The Hong Kong Polytechnic University

[3] Department of Construction Management and Real Estate, Shenzhen University, Shenzhen, China

Abstract

Machine learning has gained increasing popularity in building energy management due to its powerful capability and flexibility in model development as well as the rich data available in modern buildings. While machine learning is becoming more powerful, the models developed, especially artificial neural networks like Recurrent Neural Networks (RNN), are becoming more complex, resulting in "darker models" with lower model interpretability. The sophisticated inference mechanism behind machine learning prevents ordinary building professionals from understanding the models, thereby lowering trust in the predictions made. To address this, attention mechanisms have been widely implemented to improve the interpretability of deep learning; these mechanisms enable a deep learning-based model to track how different inputs influence outputs at each step of inference.

This paper proposes a novel neural network architecture with an attention mechanism for developing RNN-based building energy prediction, and investigates the effectiveness of this attention mechanism in improving the interpretability of RNN models developed for 24-hour ahead building cooling load prediction. To better understand, explain and evaluate these neural network-based building energy prediction models, the obtained attention vectors (or metric) are used to visualize the influence of different parts of model inputs on the prediction result. This helps the users to understand why predictions are made by the model, as well as how input sequences proportionally influence the output sequences. Further analysis of attention vectors can provide interesting temporal information for understanding building thermal dynamics, like the thermal inertia of the building. The proposed attention-based architecture can be implemented in developing optimal operation control strategies and improving demand and supply management. The model developed based on this architecture is assessed using real building operational data, and shows improved accuracy and interpretability over baseline models (without adopting attention mechanisms). The research results help to bridge the gap between building professionals and advanced machine learning techniques. The insights obtained can be used as guidance for the development, fine-tuning, explanation and debugging of data-driven building energy prediction models.

Keywords: Cooling load prediction; Attention mechanism; Recurrent neural network; Interpretable machine learning; Building energy management

1. Introduction

Building energy management plays an essential role in global sustainable development due to the huge energy consumption of buildings. Building construction and operations take up 36% of global final energy use and 39% of energy-related carbon dioxide (CO_2) emissions in 2017 [1].

Meanwhile, buildings are becoming major users in power grids, and significantly influencing supply-demand balance and grid reliability. In Hong Kong, buildings are responsible for over 93% of total electricity use, over 30% of which is used by air conditioning [2]. Accurate building cooling load and energy consumption prediction is fundamental for reducing building energy consumption and improving the reliability of the building-grid eco-system. Building energy use prediction models are widely used in developing energy-efficient-optimal control and diagnosis methods [3], evaluating building design alternatives [4], and developing the demand and supply management strategies in power grids [5, 6].

Data-driven building energy modeling has attracted increasing interest in recent years as it requires little a *priori* knowledge of buildings and building energy systems which cannot be easily obtained in large modern buildings. Advanced machine learning algorithms are adopted to develop accurate and computationally efficient data-driven models from massive data in building automation systems. Fan et al. [7] exploited the potential of supervised and unsupervised deep learning in predicting the 24-hour ahead cooling load of an educational building. Zhang et al. [8] presented a weighted-hybrid support vector regression model to forecast building energy consumption of an institutional building. Wang et al. [9] compared the building hourly electricity usage prediction performance of random forest, regression tree and support vector regression models on two educational buildings.

While machine learning is becoming more and more powerful with the advancement of AI, the complexity of models developed, especially artificial neural networks like Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN), is dramatically increasing. This results in “darker” models with lower model interpretability, which means what’s happening inside the model is unclear to model users and even model developers. In developing data-driven models,

a choice usually needs to be made between complex but “darker” models such as RNN and CNN, and simple models easier to understand, such as linear regression and decision trees [11]. The latter usually cannot fully capture the coupled non-linear dynamics of building operations, thus cannot achieve desirable accuracy. Rahman et al. [10] adopted deep RNN for hourly electricity consumption prediction of commercial and residential building over a medium-to-long term time horizon (i.e. time horizon of ≥ 1 week). Two 6-layered RNN models were proposed to predict building electricity consumption. The model input was a combination of weather variables (dry-bulb temperature and relative humidity), schedule-related variables (the hour of day, day of week, day in a given month and month number) and frequency-related variables. Kim and Cho [11] proposed a CNN-LSTM neural network to forecast residential electricity consumption. The CNN-LSTM neural network consists of 7 layers, with more than 190,000 parameters to be optimized. Those models are totally incomprehensible to ordinary building professionals. While machine learning models play an increasingly important role in smart building management, building professionals need to roughly understand how the models work and perform before widely embracing the technology. More energy savings are achievable if decision-makers could understand and trust the underlying models [13]. Substantial efforts have been made to improve the interpretability of machine learning [14-16].

A few interpretation methods have been developed to make the process of developing black-box models and the models themselves understandable to humans. One typical method is to approximate a dark black-box model, with an interpretable model (e.g., multivariate linear regression and decision tree). Bastani et al. [14] propose d to construct global explanations of complex black-box models using decision trees to approximate the original models. Another interpretation method provides summary statistics (e.g., feature importance, and pairwise feature

interaction strengths) for each input feature to indicate the impact each feature has on the machine learning model's predictions. For instance, Altmann et al. [15] proposed a permutation feature importance measure to evaluate the predictive value of an input feature for a black-box model, by evaluating how the prediction error increases when a feature is not available. The P-values computed with permutation importance were very helpful in deciding the significance of input variables, and therefore improved model interpretability. Another method is to find out what changes to input can cause a change in output, for example, the widely used counterfactual explanations [16]. To explain the prediction Y made at a data point X, this method aims to find another data point X' which is in some way related (or similar) to the original instance X but leads to a different prediction Y'.The counterfactual explanations method requires that the data points themselves can be interpreted. Therefore, it works well for images and texts, but is less useful in the building energy prediction field with tabular data of numerous features (e.g., cooling load, water flow rate, and outdoor dry-bulb temperature).

Attention mechanisms are a state-of-the-art interpretation method specifically for artificial neural networks, inspired by human recognition, which allow neural networks to pay attention to how different inputs influence outputs at each step of inference in the model development process and explain the influences either quantitatively or graphically. It was originally proposed by Bahdanau et al. [17] in the context of Neural Machine Translation as an enhancement to the RNN with the encoder-decoder architecture. Afterwards, several variants of attention mechanisms have been proposed, and they have advanced the state-of-the-art in machine translation [18], image captioning [19], video captioning [20], visual question answering [21] and generative modeling [22], etc. For instance, Luong et al. [18] proposed two effective implementations of attention mechanisms in neural machine translation: a global implementation attending to all source words

continuously and a local one examining only a subset of source words at a time. The results showed that the local implementation significantly improved performance over non-attentional systems. Xu et al. [19] constructed an attention-based CNN-RNN model for image capturing. The research showed how the model automatically learned to fix its gaze on salient objects while generating the corresponding words in the output sequence. Various researchers have shown the effectiveness of attention mechanisms in improving the interpretability of neural networks in many areas. However, the performance of attention mechanisms in deep learning-based building energy use prediction is not yet clear.

This paper investigates the effectiveness of attention mechanisms in improving the interpretability of RNN models for 24-hour ahead building cooling load prediction. To the best of the authors' knowledge, this is the first study adopting attention mechanism-based neural networks in building energy consumption prediction. Attention metrics are developed to explain and visualize why a certain prediction is made by the model, and how inputs contribute proportionally more to the output. The introduction of attention mechanisms into RNN models proposed in this study also provides a novel approach for understanding building thermal dynamics like the thermal response of the building, by further analyzing the attention vector for discovering crucial temporal information. This approach is a valuable supplementary to existing physics-dominated methods for characterizing the dynamics of large complex buildings. Furthermore, this paper proposes an attention based neural network architecture for building energy use prediction, which is valuable for building professionals to understand and adopt neural network models. A 24-hour ahead building cooling load prediction model is developed based on this architecture and assessed and compared with baseline models using real building operational data.

The remaining part of the paper is organized as follows. Chapter 2 presents an overview of RNN including typical techniques for improving it. Chapter 3 presents the research methodology. The research results are presented and discussed in Chapter 4 and conclusions are drawn in Chapter 5.

2. Overview of RNN

2.1 Basics of recurrent neural networks

Recurrent neural networks (RNNs) were developed for analyzing time-series data and have been successfully used in various fields, such as speech recognition, machine translation and image captioning [23, 24]. RNN deals with input sequence/time-series data by individual vectors at each step and preserves the information it has captured at previous time steps in a hidden state.

Fig. 1 shows the difference between conventional artificial neural network (ANN) and RNN. Fig. 1(a) presents an example of a conventional ANN with one input layer, one hidden layer (i.e. a dense layer), and one output layer. The W_1, W_2, b_1, and b_2 are weights and biases. $Act_1(z)$ and $Act_2(z)$ are the activation functions. The processes of conventional ANN generating each output sample (e.g. Y_A) using the corresponding input sample (e.g. X_A) are the same, and do not affect each other.

Fig. 1(b) presents an example of RNN using the basic recurrent unit. The W_h, W_y, and U_h are matrices containing input weights, output weights and recurrent weights, respectively. $Act_h(z)$ and $Act_y(z)$ are activation functions, and b_h and b_y are biases. A hidden state vector H_t, which has the same length as the input variables, is defined to preserve the information which has been observed. Its values are set to zero at the initial time step. The hidden state vector at time step $T-1$ (H_{T-1}) will be used together with the input data at time step T (X_T) to calculate the hidden state

at time step T (H_T). In principle, RNN is able to preserve information observed in previous time steps, and has higher potential than conventional ANN to learn temporal relationships or dynamics.

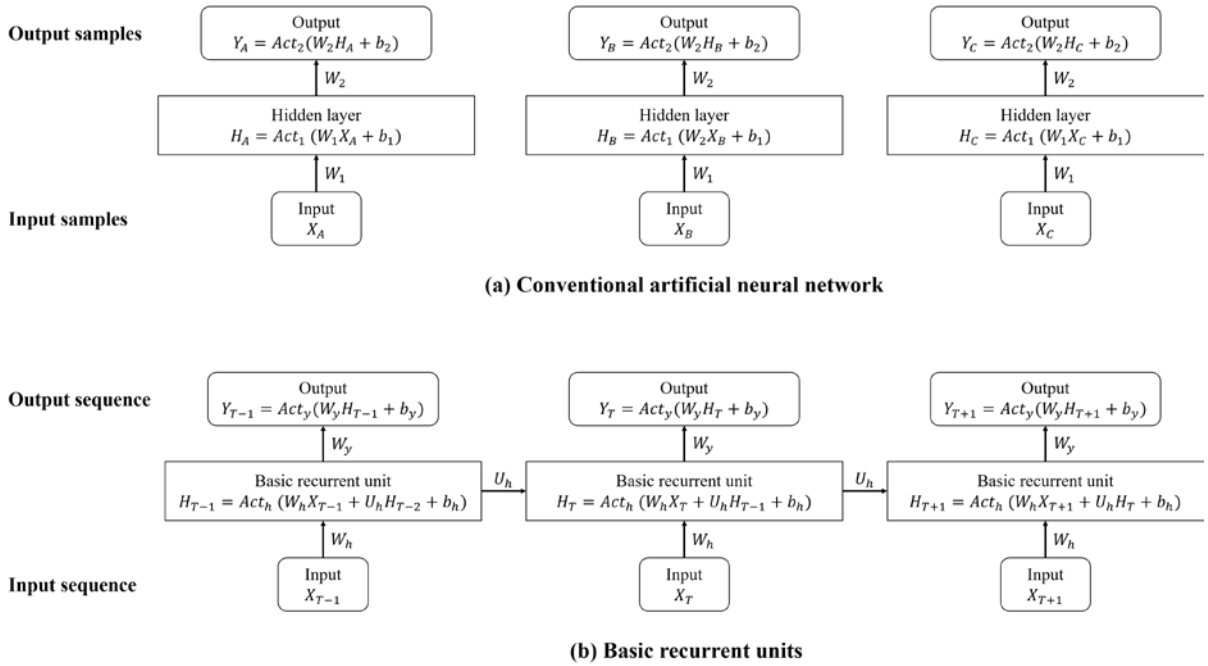


Fig. 1. Schemata of (a) conventional ANN, (b) RNN

However, the capability of a RNN in capturing long-term temporal relationships is usually limited due to the problem of vanishing or exploding gradients, i.e. that the model will become untrainable with an increase in recurrent operations [25]. Over the past couple of decades, several variants of RNN have been proposed to deal with this problem. The most effective and widely-used solutions are the Long Short-Term Memory (LSTM) units [25], and the Gated Recurrent Units (GRU) [26]. LSTM creates an additional cell state for information processing (learning what data in a sequence is important to keep). That is, the output at each time step is calculated based on the input data, the hidden state and the cell state. The LSTM enables the reinjection of past information at a later time by calculating by what percentage will the past information be allowed to affect the present information, thus helping to deal with the problem of vanishing or exploding gradients [24]. The

GRU has been proposed as a simplified alternative to the LSTM, which uses only an update gate and a reset gate and therefore has higher computation efficiency at the expense of slight model accuracy degradation. In this study, the basic recurrent units, LSTM and GRU will be used to develop 24-hour ahead cooling load prediction models and their performance will be compared.

2.2 Strategies for multi-step ahead prediction

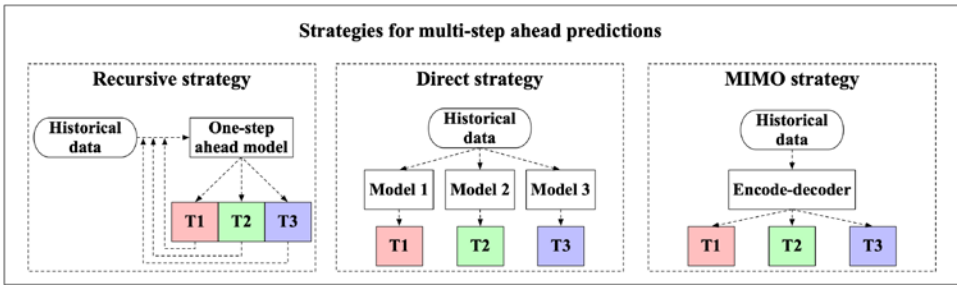


Fig. 2. Strategies for multi-step ahead predictions [27]

Time series predictions can be classified into one-step ahead prediction and multi-step ahead prediction. The former usually needs one prediction model to make prediction at the next time step only, but the latter may use the same or different prediction models at different time steps. The input used for prediction at each step in multi-step ahead prediction may be the same or different. Therefore, the architecture of multi-step ahead prediction is an issue to be considered. There are three main inference strategies for multi-step ahead predictions, i.e., the recursive strategy, the direct strategy, and the multi-input-multi-output (MIMO) strategy as shown in Fig. 2 [27]. T_1, T_2 and T_3 in the figure represent predictions made at consecutive time steps. Historical time series data are used as the input. The recursive strategy is based on one-step ahead prediction for generating multi-step ahead predictions. The prediction at time step T will be used as the model input for the next prediction, and the process continues recursively till the prediction is completed over the entire time horizon defined by the number of steps. On one hand, the recursive strategy is

easy to implement and relies solely on a one-step ahead prediction model. On the other hand, it may suffer from the problem of error accumulation, as the input of predicted values into later prediction causes any error which occurs to accumulate along the prediction time horizon. The resulting prediction accuracy can be very poor, especially when the prediction time horizon is long. By contrast, the direct strategy develops different models for each time step in the prediction time horizon, or n models to predict values at times $T+1, T+2, \ldots, T+n$, respectively. Compared with the recursive strategy, the direct strategy does not suffer from the problem of error accumulation. However, it requires more computational resources as multiple models are needed. In addition, since the predictions at different time steps are generated independently using different models, the prediction profiles may be incoherent and disconnected.

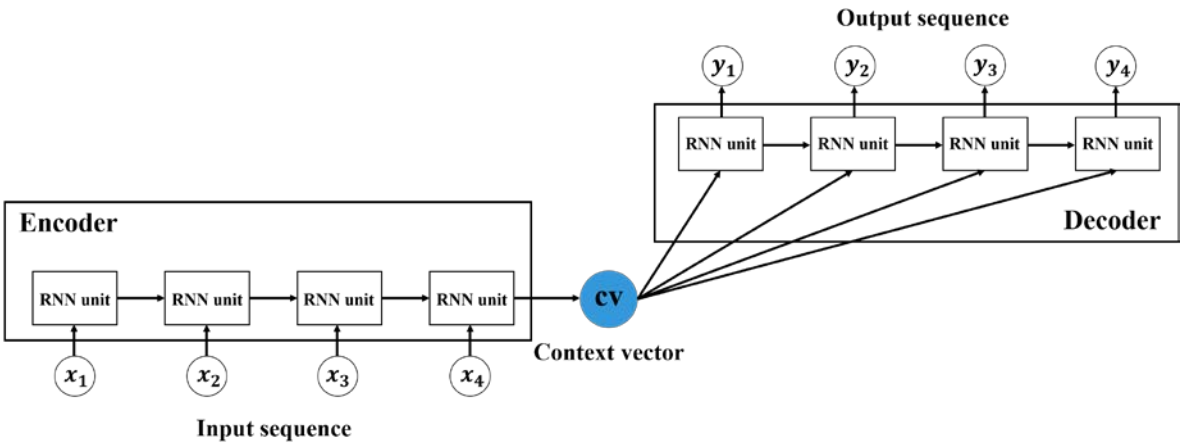


Fig. 3 Diagram of a sequence to sequence architecture

The MIMO strategy, whose inputs and outputs are time series of either a single variable or multiple variables, has been proposed to better capture the stochastic dependencies among future values (input sequence) [28]. In principle, it can avoid the problem of error accumulation associated with the recursive strategy and overcome the problems caused by using different prediction models at different steps in the direct strategy [29]. For multi-step building energy consumption prediction,

the MIMO strategy stands out from recursive and direct strategies in terms of prediction accuracy and flexibility [27].

To implement the MIMO strategy using RNN, a sequence-to-sequence (Seq2Seq) or encoder-decoder architecture, as shown in Fig. 3, can be adopted [30, 31]. The encoder and the decoder are developed based on RNN. The encoder transforms the input sequence into a context vector and stores its hidden states, while the decoder generates the output sequence based on the context vector. Such architecture can provide a great flexibility in practical applications, as it allows the output sequences to have a length different from that of the input sequence. The decoder can use either the context vector encoded as the input for decoding at every time step or the encoder's hidden states as its initial state and generates successive predictions based on previous decoding output iteratively. In this study, the latter is used for developing a Seq2Seq model as it is easier to implement.

2.3 Introduction of Attention mechanism into RNN

An inherent problem with seq2seq models is that the context vector produced by the encoder is of a fixed-length. When applied to modeling over long prediction horizons, the length of the context vector doesn't increase which may result in information loss. When the length of an input or output sequence increases, it is difficult to encode all information contained in a long sequence into a single context vector. Consequently the decoder cannot produce an accurate prediction based on that vector [33]. The accuracy of seq2seq models degrades significantly with increases in the length of input and output sequences, which is a persistent problem in Neural Machine Translation [34].

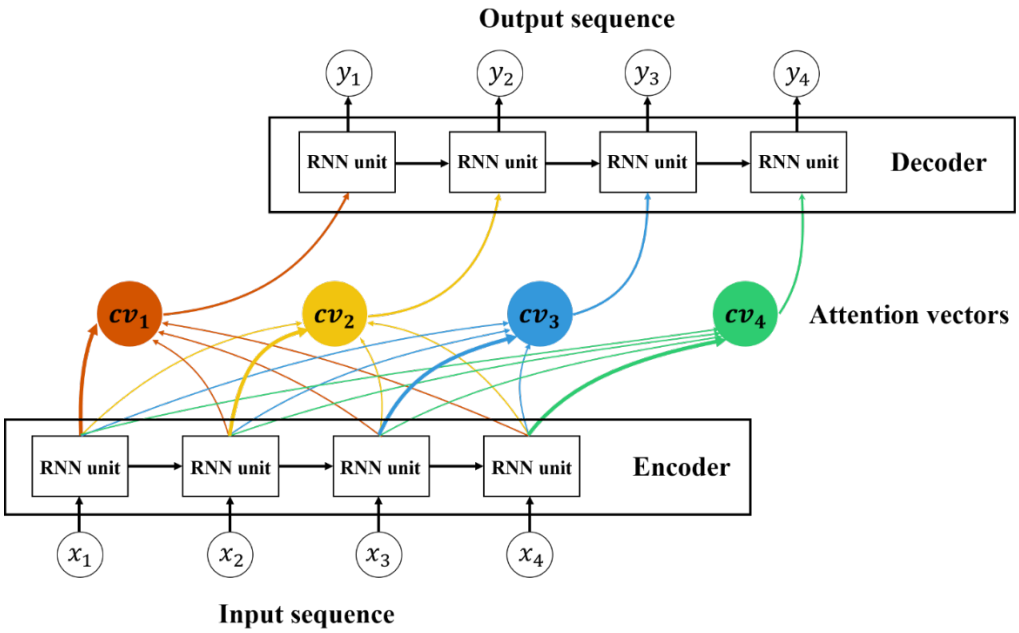


Fig. 4. Diagram of an encoder-decoder model with an attention mechanism

Attention mechanisms, first proposed by Bahdanau et al. [17], address the aforementioned problem by assigning high attention weights to parts of long input sequences of high relevance to the output sequences, and enabling neural networks to focus more on those parts. A diagram of an encoder-decoder model with an attention mechanism is shown in Figure 4. Such models iteratively process their input sequence by selecting content with high relevance to the output at every step.

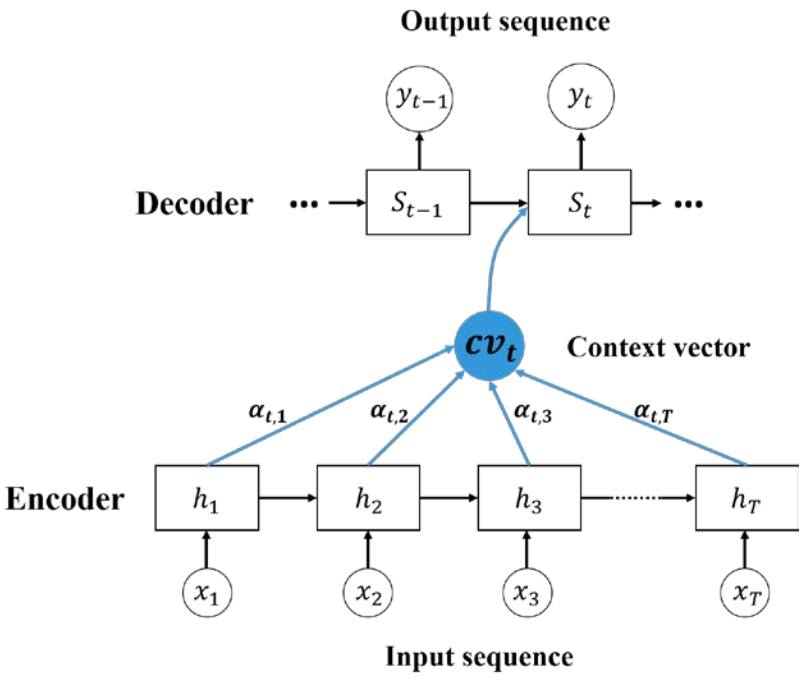


Fig. 5. The graphical illustration of the Attention-based model trying to generate the t-th output time step y_t given a input sequence $[x_1, \dots x_T]$ [17]

As shown in Fig. 5, given the input sequence $[x_1, \dots, x_T]$ and output sequence $[y_1, \dots, y_t]$ (T and t do not need to be the same), the Bahdanau attention mechanism works as follows: instead of the context vector being passed only once at the start of decoding, a unique context vector cv_i (also called attention vector) is calculated for each output time step y_i. The encoder used here generates a sequence of annotations $[h_1, \dots, h_T]$ for each vectors (i.e. x_j) in the input sequence. Using the softmax function, the context vector cv_i is then computed as a weighted sum of these annotations with Eq. (1). The weight α_{ij} of each annotation h_j is computed using Eq. (2), where $e_{ij} = a(s_{i-1}, h_j)$ is an alignment function which scores how well the inputs around position j match the output at position i. This score is based on the RNN hidden state s_{i-1} (just before emitting y_i) and the j-th annotation h_j of the input sequence. At last, the decoder generates an output for the i-th

time step by using the i-th context vector cv_i and the previous hidden outputs s_{t-1}. The alignment model can be parametrized as a feedforward neural network which is jointly trained with all the other components of the proposed system.

$$cv_i = \sum_{j=1}^{T} \alpha_{ij} h_j \tag{1}$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T} \exp(e_{ik})} \tag{2}$$

Instead of encoding all information in a long sequence into a single and identical context vector for all output time steps, attention mechanisms enable the model to calculate a unique context vector for each output time step. In other words, when generating outputs at different time steps, the model does not rely on all inputs equally, but focuses on the most relevant parts of the input sequence. In doing so, the encoding and decoding processes become more accurate, effective, and reliable. The attention mechanism has proved to be effective in improving the performance (including prediction accuracy) of neural networks [18-22].

Attention mechanisms can enhance model interpretability. Take university ranking as an example. The task is to predict ranking indicators (e.g. reputation, citations, and faculty/student ratio) based on all the related information of a university (e.g. location, staff list, research funds etc.). Based on all the related information of a university (e.g. location, staff list, research funds etc.), a traditional neural network (without adopting attention mechanism) can generate the predictions of ranking indicators (e.g., reputation, citations, faculty/student ratio). Still, it cannot explain to you how different inputs influence these indicators. However, an attention-based neural network can not only make the same predictions, but also elucidate how different ranking indicators are generated. For instance, predicting the reputation indicator relies more on the survey results of

academics and employers, while the citations of all research publications and the number of faculty members play a significant role in the citation indicator.

Kelvin Xu et al. [35] utilized an attention mechanism to find out the corresponding regions (of each word) when generating the description (i.e. sentence) of figures. Two kinds of attention mechanisms are adopted, i.e., soft attention and hard attention. The aforementioned attention mechanism belongs to soft attention. It computes a weight α_i for each x_i and uses it to calculates a weighted average for x_i as the decoder input. On account of its admissibility, soft attention can be directly inserted into models for training. The gradients can be back-propagated through an attention mechanism module to other parts of the model. In contrast to soft attention which adopts a deterministic method, hard attention is a rather stochastic procedure. Instead of a weighted average, hard attention selects only part of the hidden state of encoder based on the probability S_i. In order to calculate the gradient descent correctly in the back-propagation, the Monte Carlo sampling method is needed to estimate the gradient of each module. In general, hard attention requires less computation and memory (as the entire input is not being stored or operated over usually) but cannot be easily trained as the objective is non-differentiable. In this research, the soft attention mechanism is adopted due to its easier implementation with standard backpropagation methods.

3. Research Methodology

3.1. Research outline

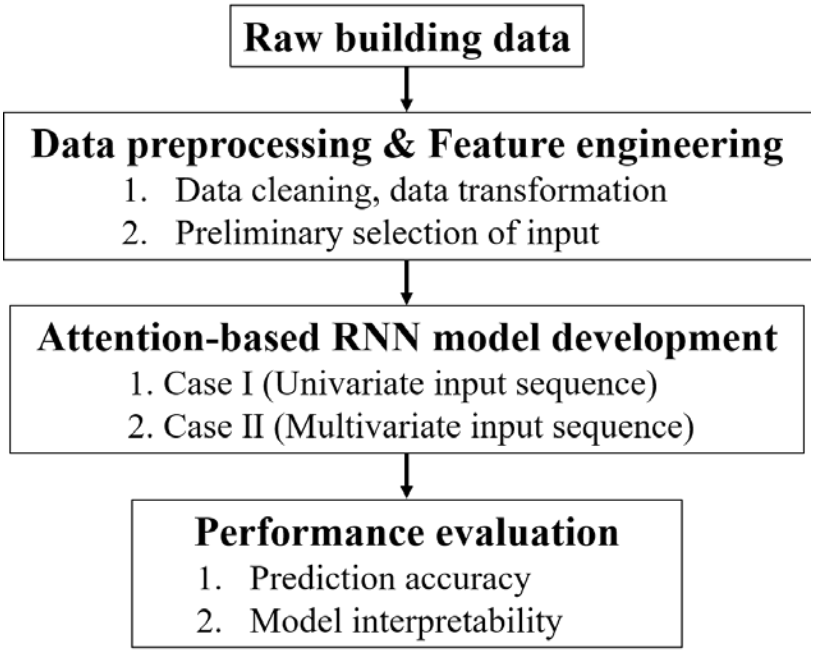


Fig. 6. Research outline

Fig. 6 presents the general research outline of this study. Data preprocessing is first carried out to enhance the data quality by filling in missing values, removing outliers, and providing required data attributes for further analysis. Feature engineering preliminarily selects the most influential variables as inputs of the prediction model. Afterwards, two seq2seq attention-based RNN models are developed for 24-hour ahead cooling load prediction based on previous 24-hour data in a one-hour time interval. Case I uses a univariate input sequence, while Case II adopts a multivariate input sequence. The output is the sequence of the next 24 hours' cooling load. The prediction accuracies are compared with baseline RNN models without adopting an attention mechanism. The obtained attention vectors (cv_i in Eq. 1) can visualize why a particular prediction is made by the model, and which part of the input sequence contributed more to the prediction. A novel

approach based on further analysis of the attention vectors is proposed to understand the building thermal dynamics.

3.2. Attention-based Seq2Seq architecture

Fig. 7 presents the proposed attention-based Seq2Seq architecture for 24-hour ahead building cooling load prediction with a one-hour time interval. The architectures of Case Ⅰ and Case Ⅱ are nearly identical, with the only differences lying in the model input and input layer. As mentioned in Section 2.2, the input and output are both time sequences in the Seq2Seq architecture. The inputs for Case I and Case II are shown in Figure 8. The input sequence lengths in Case I and Case II are both 24, allowing the direct use of previous 24-hour data to make predictions. Case I adopts previous 24-h building cooling load data with a one-hour time interval as its model input sequence. Case II adopts a multi-variable input sequence, as shown in Fig. 8, including cooling loads, time-related variables, and outdoor climate variables. Outdoor variables consist of solar radiation, relative humidity (RH), and outdoor dry-bulb temperature (dry-bulb T). Time-related variables describe the day type (i.e. weekday=0 or weekend=1) and 24-h time (Time, which is transformed to a binary value through one-hot encoding in the data preprocessing procedure).

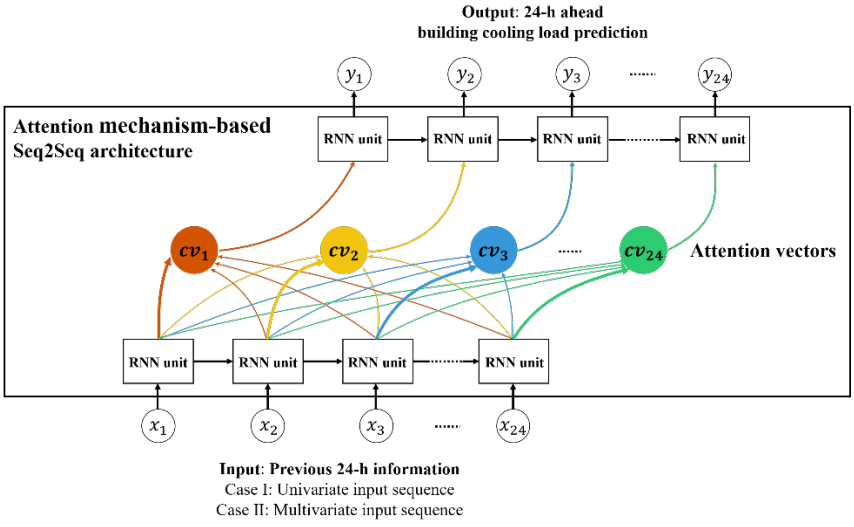


Fig. 7. Proposed attention-based Seq2Seq architecture

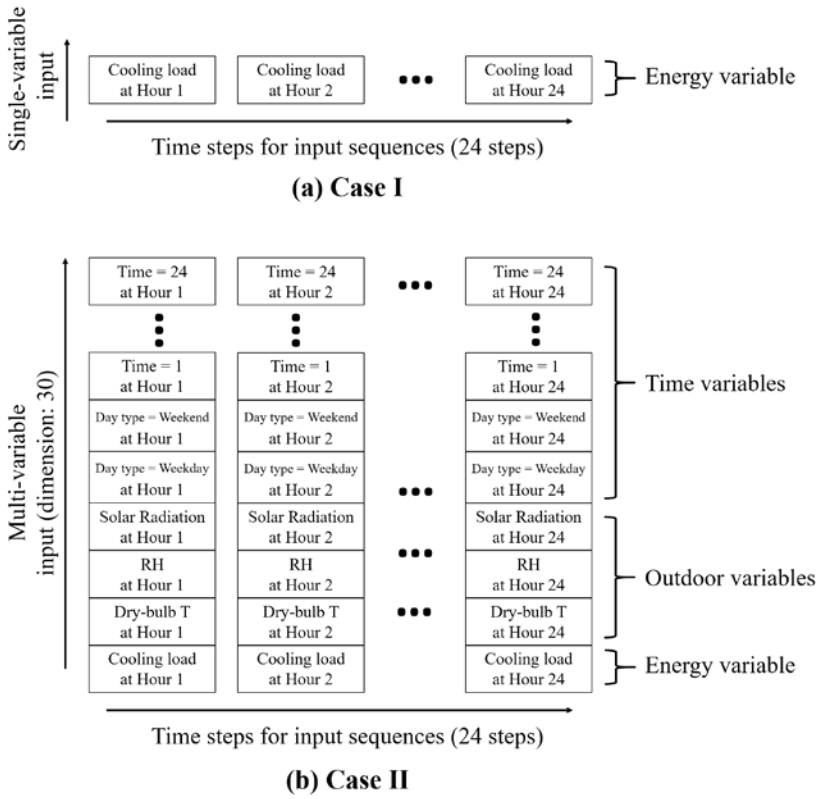


Fig. 8. Input for Case I and Case II

The bidirectional structure is adopted in both the encoder and decoder to obtain forward and backward relationships in the input sequence. The Seq2Seq structure with an attention mechanism is implemented in this research to address the gradient exploding and vanishing problems, as well as to improve model accuracy and interpretability. 24 attention vectors are generated corresponding to the 24 steps of the output sequence, which are the predicted cooling loads for the future 24 hours. The length of each attention vector is 24, corresponding to the input sequence length. Together, these vectors form a 24×24 attention matrix. The encoder first encodes the information contained in the input sequence into each attention vector. The decoder generates the predicted values in the output sequence one-by-one by decoding the corresponding attention vector.

Two baseline Seq2Seq RNN models without attention mechanisms are also constructed for performance comparison.

In this research, all the models were constructed and tested using the Python programming language and the Keras package [36].

3.3. Techniques adopted to improve RNN for building cooling load prediction

RNN provides an elegant way of dealing with sequential/time-series data that embodies correlations between data points that are close in the sequence [37]. Building operations are highly dynamic and involve complex interactions between building envelope, indoor and outdoor environment, building energy systems, occupants and various automatic controllers and the power grid. As a result, the building cooling load and energy use is highly dynamic and non-linear. This makes RNN a promising modeling method for the dynamics of these variables and general prediction-makings [27, 39]. The cooling load at time step T may be affected by cooling loads at previous time steps, owing to building thermal mass and periodicity in indoor and outdoor conditions and working schedules. Standard unidirectional RNNs are trained according to the normal time order of past information. However, this kind of method cannot fully use all available input information. It is possible to use two separate networks (one for positive time direction, the other for negative time direction) and then merge the results. Based on this idea, Schuster and Paliwal [37] proposed a bidirectional recurrent neural network (BRNN), which can be trained using all available information in the past and future of a specific time frame. Two recurrent blocks are connected with the same output, one moving forward and the other moving backward through the timeline. As a result, the model outputs obtained at time T are representations of both past and future information. Bidirectional operations have had great success in analyzing various types of

sequential data, especially in machine translation. Considering that building operation is intrinsically periodic, BRNN potentially has use in building energy prediction [26].

Recurrent models are typically of high complexity and are therefore vulnerable to the problem of overfitting [33]. Dropout is a popular technique in the field of deep learning to fight against over-fitting. It refers to the process of randomly setting parts of the neural network to zero during model training, with the remaining weights trained by backpropagation [40]. The dropout technique prevents overfitting by forcing neurons to be robust and rely on population behavior, rather than on the activity of specific units.

Previous studies have shown that conventional dropout operations are of little use to RNN models, as the random noise introduced may be amplified during recurrent operations. Gal and Ghahramani proposed a proper method to perform regularizations for recurrent models: rather than using different dropout masks at different time steps, the same dropout mask is used for both the input, output and recurrent layers [35]. The resulting dropout operations are controlled by two parameters, dropout and recurrent dropout, both ranging from zero to one. In this study, the dropout technique is implemented in the training process of the proposed attention-based RNN to prevent overfitting, and the influence of different dropout parameters will be studied (in model optimization).

An early-stopping training scheme is also adopted in this research to prevent overfitting, i.e., terminating the training process when the resulting accuracy of validation data stops increasing after a certain number of iterations, as shown in Fig. 9. The early-stopping scheme can also accelerate the training process and improve the efficiency of parameter adjustment. $Patience$ is the tolerated maximum epoch number within which there is no improvement of model performance. Setting $patience$ to zero means that the training is terminated as soon as the performance measure begins to worsen. The setting of $patience$ is actually a tradeoff. If

patience is set too high, then the final accuracy rate may be slightly lower than optimal. If the *patience* is set too low, the model is likely to fluctuate in its early stages and stop at the stage of global search, and will have a generally poor accuracy rate. In this research, *patience* is set to 3 based on preliminary training results shown in the Appendix.

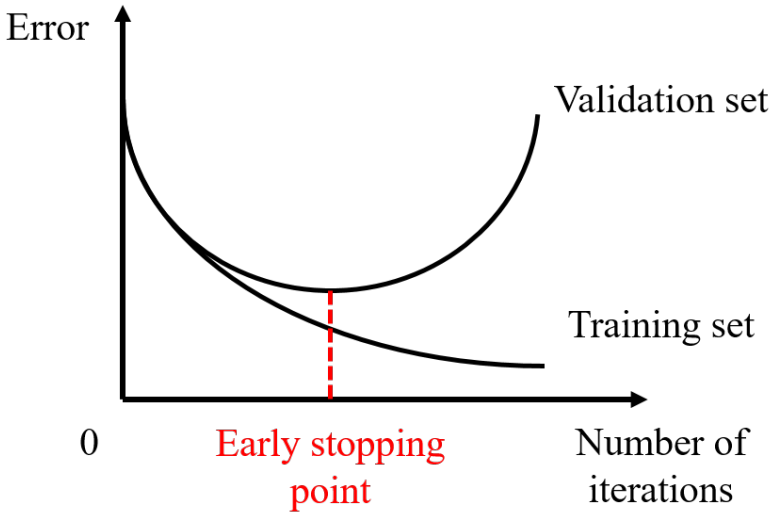


Fig. 9. Early stopping scheme to prevent overfitting

3.4. Performance indexes for model evaluation

The performance indexes used in this research include the root mean square error (RMSE), the mean absolute error (MAE) and the coefficient of variation of the root mean square error (CV-RMSE). They are calculated based on Equations. (3) – (5), respectively.

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}} \tag{3}$$

$$\mathrm{MAE} = \frac{|\hat{y}_i - y_i|}{n} \tag{4}$$

$$\mathrm{CV-RMSE} = \frac{\sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}}}{\frac{\sum_{i=1}^{n} y_i}{n}} \quad (5)$$

Where y_i is the actual energy consumption, $\hat{y}_i$ is the predicted energy consumption, and N is the number of observations. Among these three indicators, RMSE and MAE are scale-dependent, while CV-RMSE is scale-independent which allows us to express the error of the model in a percentage.

In addition, the training time per epoch is recorded to reflect the computation efficiency of each model. One epoch is defined as a full pass through the training set. The training time per epoch is a neutral index to evaluate the computation time of machine learning algorithms, as different researchers may adopt different stopping criteria in model development. It should be noted that once a model is trained, the computing time for generating predictions is negligible using modern computing devices [27].

4. Results and Discussions

4.1. Description of the raw BAS data

The data used in case studies in this research were retrieved from the building automation system (BAS) of the tallest building in Hong Kong, the International Commerce Centre (ICC). This building is about 490 m high with a total floor area of approximately 321,000 m^2, consisting of a basement of four floors, a block building of six floors and a tower building of 98 floors. The building is served by a central chilling system consisting of six identical high-voltage centrifugal chillers which provide chilled water for air handling units. The rated cooling capacity and power consumption of each chiller are 7230 kW and 1270 kW, respectively. A total of 463 days (from January 2017 to August 2018) of building operational data were retrieved for analysis. The daily

cooling load profile of the ICC during this period can be seen in Figure 10. The time interval of data collection is 10 minutes. The climate data in the same period, including outdoor dry-bulb temperature, relative humidity, and solar radiation, were obtained from the Hong Kong Observatory.

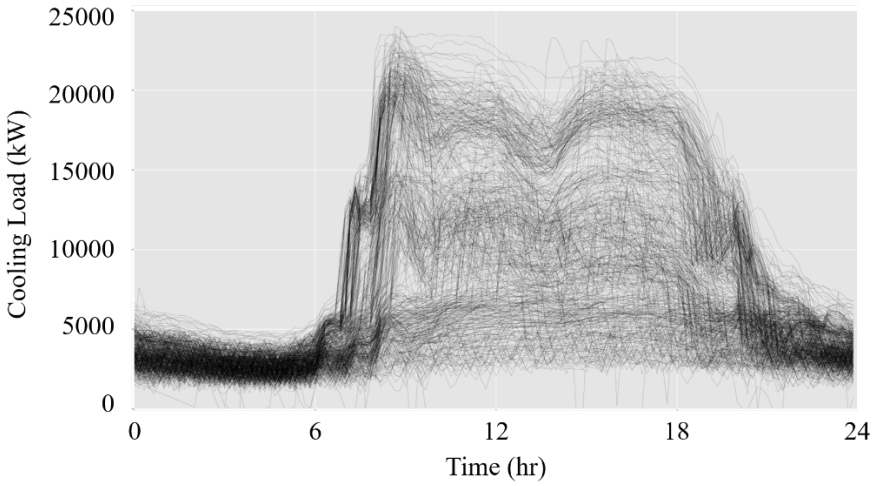


Fig. 10. Daily cooling load profile of ICC

As the data quality of BAS data is usually low due to measurement noise, sensor faults, transmission problems, and other factors. A data preprocessing procedure is used in this research to enhance data quality. The missing values are filled in using moving average method, while the outliers are identified with domain expertise. Afterwards, min-max normalization is adopted to transform the data into a suitable scale for further analysis. As artificial neural networks cannot directly operate on categorical data, they require all input and output variables to be numeric. One-hot encoding is adopted in this research to transform some categorical variables (e.g., the day of the week and hour of the day) into a numerical form. The training and testing dataset take up 70% and 30% of the whole dataset, respectively. During the model development process, 10% of the training data are randomly selected as validation data.

4.2. Performance evaluation

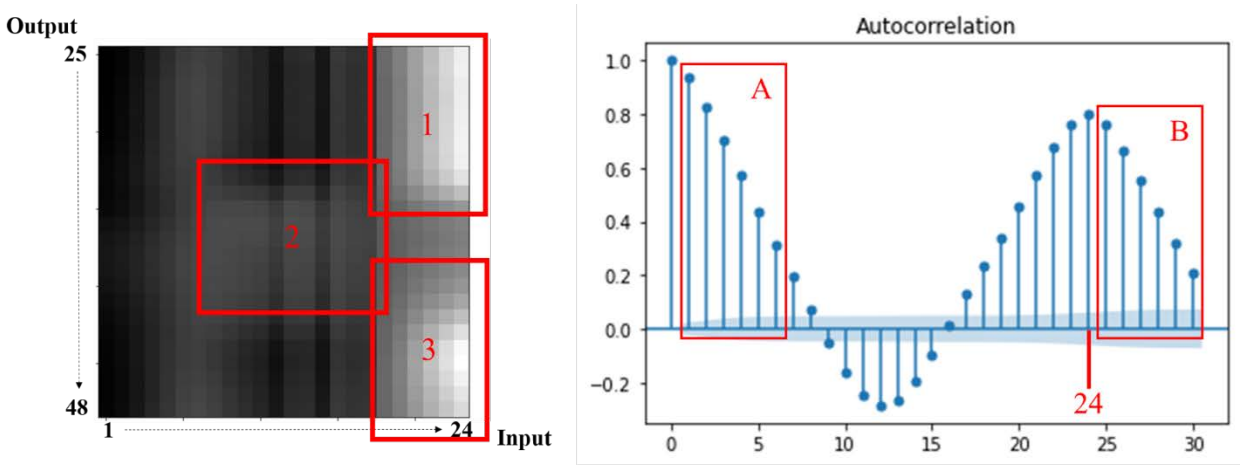
Two prediction models are developed based on the proposed attention mechanism based Seq2Seq architecture, both to fulfill the 24-h ahead building cooling load prediction task. Case I selects the previous 24-h energy consumption data as inputs to predict the 24-h ahead building energy consumption, while the input variables for Case II include energy variables, time variables and outdoor variables. For further elaboration, the index of inputs and outputs is set to 1-24 and 25-48, respectively.

Table 1. The prediction accuracies using different prediction approaches (Case I)

Attention mechanism	RNN units	RMSE	MAE	CV-RMSE	Training time per epoch
With attention mechanism	LSTM	914	661	0.328	5.5s
	GRU	885	644	0.315	5.2S
Without attention mechanism	LSTM	933	671	0.352	2.9s
	GRU	1086	781	0.39	2.6s

Table. 1 presents the prediction accuracies of the models using different prediction approaches for Case I. The accuracies reported were calculated for the 24-step ahead predictions in the testing data set. It can be seen that the attention mechanism leads to a clear improvement in accuracy for both kinds of RNN units. The best model performance is obtained by the attention-based RNN with GRU architecture, with a CV-RMSE of 0.315. Several researchers have reported that the prediction model with a CV-RMSE around 30% or less is acceptable for engineering purposes when using hourly data [39]. In terms of the recurrent unit types, as expected GRU uses the least

computation time both with and without using an attention mechanism. However, LSTM seems to show better collaboration with the attention mechanism compared with GRU.



(a) Attention matrix (b) Autocorrelation analysis

Fig. 11. (a) Color-map of Attention matrix (white: high weight; black: low weight), (b) The autocorrelation of cooling load time series

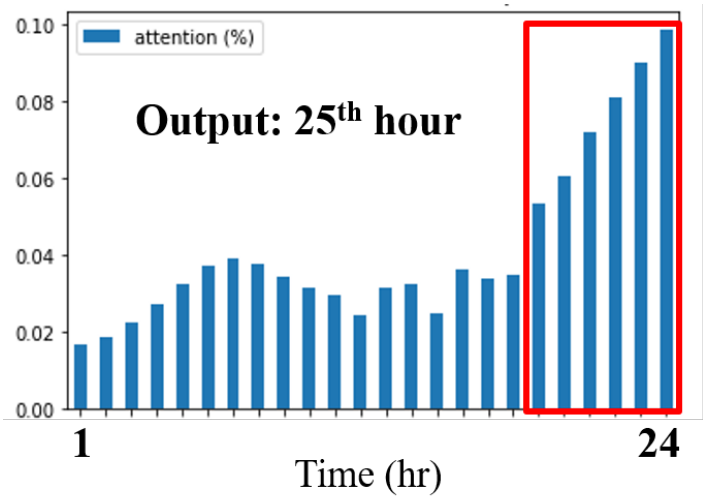


Fig. 12. Attention vector (output: cooling load at 25th hour)

To better present the intrinsic characteristics in the attention-based RNN model, the color-map of the attention matrix (the concatenation of all context vectors) is shown in Fig. 11(a), where the

color represents the value of weights (the darker the lower). The attention weight of a co-ordinate (X, Y) indicates the influence of X on Y. For example, the value of the top right corner (24, 25) displays the influence of cooling load at the 24^{th} hour on that at the 25^{th} hour during the model's prediction-making process. And Fig. 12 shows the attention vector corresponding to energy predictions for the 25^{th} hour (i.e., the first element of the output sequence). For the upper right corner area '1', the attention weights are high, showing that the energy consumption in the past several hours has the most significant influence on the energy prediction of the following several hours. This phenomenon can be explained by the building thermal resistance. For the middle area '2' and lower right area '3', the attention weights show that when the model predicts energy consumption at the 32-48th hours, the energy consumption 24 hours in the past makes the greatest contribution. This is because building energy consumption is highly influenced by occupancy behavior and outdoor climate, which shows clear periodicity. The autocorrelation analysis on the time series of the cooling load was adopted to verify the attention matrix obtained. The result of autocorrelation function of cooling load time series, as shown in Fig 11 (b), indicates similar characteristics: **Area A** in Fig. 11(b) corresponds to **Area 1** in Fig. 11(a), and **Area B** corresponds to **Area 2**. This can serve as a proof that the attention-based model indeed learned some useful and reasonable information which can be explained by the domain knowledge [19].

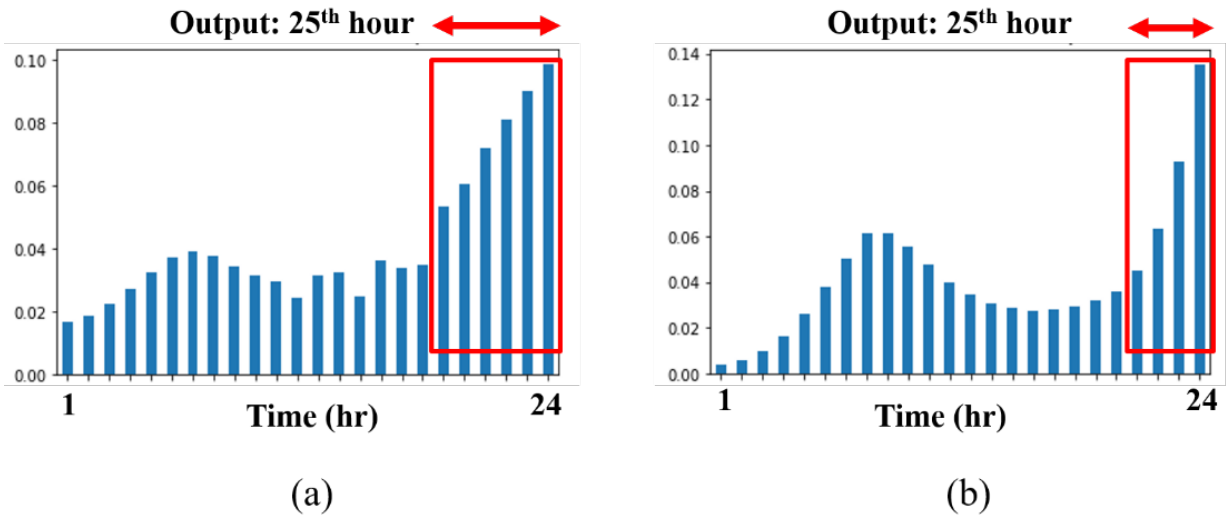


Fig. 13. Attention vector (a) ICC data; (b) another smaller residential building B

For further analysis, operational data from another residential building B is used to train the same attention-based RNN model. The attention vector for the energy prediction at the 25th hour is presented in Fig. 13 for comparison. As can be seen from the figure, the region with high attention weights is selected. For the ICC, a high-rise commercial building, the energy consumptions of the previous 6 hours have a great influence on 1-hour ahead energy prediction. For Building B, only the previous three hours' energy consumption shows a significant impact. This result accords with the different thermal inertia and capacity of these two buildings. Such visualization of the attention matrices confirms that the attention-based RNN model has actually thoroughly learnt internal building operation patterns, and is making reasonable predictions which can be explained with domain knowledge.

In Case II, instead of using only previous energy consumption data as input, several more relevant variables are added to the input as shown in Fig. 8, including (1) time variables which describe the date and daily time; (2) outdoor variables which describe the outdoor environment (e.g., outdoor dry-bulb temperature, relative humidity and solar radiation). The dimension of input is increased

from 1 to 30. And the sequence length of input and output is still set as 24. The time interval is one hour. Table 2 shows the prediction accuracies using different prediction approaches. It can be seen that the prediction accuracies are all acceptable under the four test conditions. It can be surmised from the comparison of Table 1 and Table 2 that, as expected, using more relevant input variables achieves higher accuracy.

Table 2. The prediction accuracies using different prediction approaches (Case II)

Prediction strategy	RNN units	RMSE	MAE	CV-RMSE	Training time per epoch
Attention mechanism	LSTM	524	377	0.162	8.8s
	GRU	565	412	0.177	7.6s
No attention	LSTM	576	419	0.178	6.6s
	GRU	641	459	0.198	5.3s

However, the attention mechanism can still bring improvement to the model accuracy, but to a smaller extent than in Case I. After checking and visualizing the attention vectors and attention matrix, the phenomena and patterns in Case 1 do not show again in Case II. The attention matrix appears unordered, which means that no clear knowledge or explanation can be obtained. Possible reasons include but are not limited to: (1) the training dataset is not enough for the model to learn the intrinsic relationships between different parts of input sequences and output sequences; (2) the relationship between different parts of input sequences and output sequences vary, and treating them together using one recurrent layer will not lead to reasonable results.

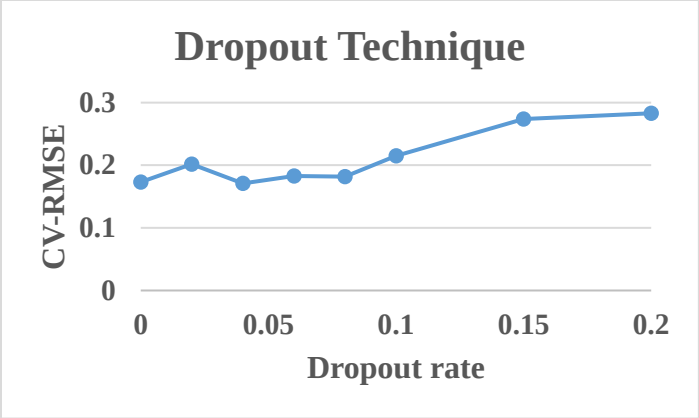


Fig. 14. Model performance with different dropout rate

This study also investigates the influence of different dropout rates on model performance. Fig. 1 presents the results with an attention-based RNN model using the LSTM architecture. It can be seen from the figure that when the dropout rate increases, the model accuracy decreases slightly, and then increases to the maximum (dropout rate = 0.04). But, as the dropout rate continues to increase, the model performance falls significantly. Although this dropout technique, which is supposed to prevent overfitting, behaves well in some reported studies of neural networks. In this study, the dropout technique is not recommended. Its stochastic characteristics may lead to clashes with the attention mechanism, which needs the model to focus on different parts of input sequences.

4.3. Discussions

The research results show that attention mechanisms can enhance recurrent neural networks for multi-step building energy prediction with both LSTM or GRU units. When the energy consumption of the previous 24 hours is inputted, the enhancement brought by the attention mechanism to the models is significant. The results show that LSTM achieves a slightly better result, but with a higher computation load, while GRU architecture achieves similar accuracy with less computation time.

Digging into the model, the visualization of attention vectors and matrices show intrinsic periodicities in building energy data: (1) when predicting building energy consumption of several hours ahead, the energy consumption of the previous several hours has the greatest influence, due to building thermal inertia; (2) When predicting energy consumption 32-48 hours into the future, the influence of energy consumption 24 hours ago is most significant. The results accord with the daily periodicities of building operation. The visualization of attention vectors also shows the crucial thermal characteristics of the building, which are hard to obtain using traditional methods. Trained by operational data from two buildings differing in scale and type, the attention vectors for the first output time step (cv_1) show similar trend (i.e. the energy consumption of the most recent several hours impact energy consumption predictions more), the window size labeled in Fig. 13 varies: the energy consumption patterns of the larger building shows longer continuity. The attention vectors obtained can be utilized to explain and debug the model, and even guide input selection for building energy modeling and prediction.

In Case II, the input variables include not only the energy consumption of previous time steps, but also outdoor environment data and time-related indicators which are transformed by one-hot encoding. The increase of input variables improves model accuracy. However, the attention-based RNN cannot well learn the relationships between inputs and outputs. Some more complex attention-based Seq2Seq architecture, for example, treating the different parts of input sequences separately, have been tested. However, the results are not encouraging. The model design may have been too ideal, and the amount and quality of training data are insufficient to support the convergence of such a deep model.

In terms of the calculation region of the context vector, the attention mechanism can be classified into three categories: soft attention, hard attention and local attention. This research adopts the soft

attention for its parameterization and easier implementation. The implementation of hard attention requires a rather precise positioning of the calculation region. However, hard attention calculates the context vector based on the whole input sequence which may lead to high computation load and convergence difficulty. For future research, it would be valuable to investigate the implementation of hard attention and local attention.

5. Conclusions

Machine learning techniques have gained increasing popularity in building energy management due to their flexibility in model development and the rich data available in modern buildings. Machine decision-making is increasingly vital in smart building energy management. Although machine learning has become more and more powerful, the complexity of models developed in recent years, especially artificial neural networks, has increased dramatically, which results in lower model interpretability.

The study investigates the impact of attention mechanisms when developing RNN models with different architectures for multi-step building energy prediction. The model developed based on this architecture is assessed using real building operational data, and shows improved accuracy and interpretability compared with recurrent neural networks without attention mechanisms and other baseline models using a recursive approach. The visualization of obtained attention vectors shows why predictions are made by the model, as well as the proportional influence of inputs on the output. Further analysis of attention-based recurrent models trained with operational data from different buildings, provides crucial temporal information for understanding the building dynamics, like the thermal response of the building. This research also investigates the implementation of different recurrent units (LSTM and GRU), an early-stopping training scheme, and the dropout technique.

The research results help to bridge the gap between building professionals and advanced machine learning techniques. The insights obtained can be used as guidance and reference for the development, fine-tuning, explanation and debugging of data-driven building energy prediction models.

Acknowledgement

The authors gratefully acknowledge the support of this research by the Research Grant Council of the Hong Kong SAR (152133/19E).

Appendix

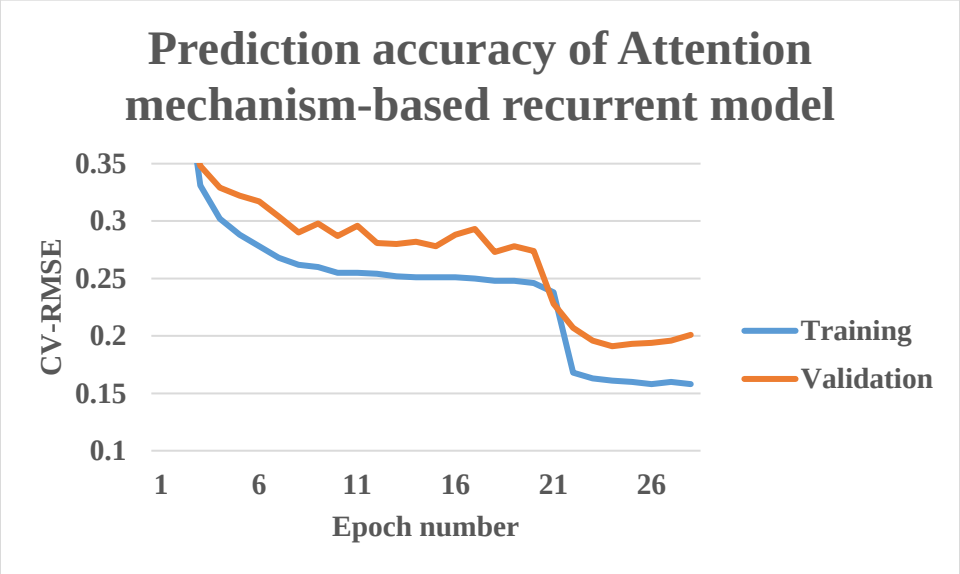


Fig. 15. The prediction accuracy of attention mechanism-based recurrent model under training dataset and validation dataset

As shown in Fig. 15, the prediction accuracy of the attention mechanism-based recurrent model shows similar trends under the training and validation datasets as the epoch number increases. For the validation dataset, the accuracy fluctuates slightly as the epoch number increases from 7

to 19. If the patience is set to less than 3, the early-stopping scheme will stop the training process in this period, and the accuracy of the obtained model will be much lower than optimal. As the epoch number increases from 22 to 27, the model accuracy under the validation dataset declines slightly, which is a signal of overfitting. Based on these preliminary results, $patience$ in this research is set to 3.

References

1. IEA (2018a), World Energy Statistics and Balances 2018, OECD/IEA, Paris.
2. EMSD, Energy end-use data, 2019.
3. Li X, Wen J. Review of building energy modeling for control and operation[J]. Renewable and Sustainable Energy Reviews, 2014, 37: 517-537.
4. Asadi S, Amiri S S, Mottahedi M. On the development of multi-linear regression analysis to assess energy consumption in the early stages of building design[J]. Energy and Buildings, 2014, 85: 246-255.
5. Xue X, Wang S, Sun Y, et al. An interactive building power demand management strategy for facilitating smart grid optimization[J]. Applied Energy, 2014, 116: 297-310.
6. Wang Z, Hong T, Li H, et al. Predicting City-Scale Daily Electricity Consumption Using Data-Driven Models[J]. Advances in Applied Energy, 2021: 100025.
7. Fan C, Xiao F, Zhao Y. A short-term building cooling load prediction method using deep learning algorithms[J]. Applied energy, 2017, 195: 222-233.
8. Zhang F, Deb C, Lee S E, et al. Time series forecasting for building energy consumption using weighted Support Vector Regression with differential evolution optimization technique[J]. Energy and Buildings, 2016, 126: 94-103.
9. Wang Z, Wang Y, Zeng R, et al. Random Forest based hourly building energy prediction[J]. Energy and Buildings, 2018, 171: 11-25.
10. Rahman A, Srikumar V, Smith A D. Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks[J]. Applied energy, 2018, 212: 372-385.

11. Kim T Y, Cho S B. Predicting residential energy consumption using CNN-LSTM neural networks[J]. Energy, 2019, 182: 72-81.
12. Molnar C. Interpretable machine learning[J]. Lulu. com, 2019.
13. Arjunan P, Poolla K, Miller C. EnergyStar++: Towards more accurate and explanatory building energy benchmarking[J]. Applied Energy, 2020, 276: 115413.
14. Bastani O, Kim C, Bastani H. Interpreting blackbox models via model extraction[J]. arXiv preprint arXiv:1705.08504, 2017.
15. Altmann A, Toloşi L, Sander O, et al. Permutation importance: a corrected feature importance measure[J]. Bioinformatics, 2010, 26(10): 1340-1347.
16. Wachter S, Mittelstadt B, Russell C. Counterfactual explanations without opening the black box: Automated decisions and the GDPR[J]. Harv. JL & Tech., 2017, 31: 841.
17. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[J]. arXiv preprint arXiv:1409.0473, 2014.
18. Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421. Association for Computational Linguistics, 2015.
19. Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In Francis Bach and David Blei (eds.), Proceedings of the 32nd International Conference on Machine Learning, volume 37 of Proceedings of Machine Learning Research, pp. 2048–2057, Lille, France, 07–09 Jul 2015. PMLR.
20. Yunchen Pu, Martin Renqiang Min, Zhe Gan, and Lawrence Carin. Adaptive feature abstraction for translating video to text. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pp. 7284–7291, 2018.
21. Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In NIPS, pp. 289–297, 2016.
22. Han Zhang, Ian J. Goodfellow, Dimitris N. Metaxas, and Augustus Odena. Self-attention generative adversarial networks. CoRR, abs/1805.08318, 2018.

23. Goodfellow I, Bengio Y, Courville A. Deep learning. 1st ed. Cambridge, London, England: MIT Press; 2016.
24. Chollet F, Allaire JJ. Deep learning with R. 1st ed. Shelter Island, New York, USA: Manning Publications; 2018.
25. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9:1735–80.
26. Chung J, Gulcehre C, Cho K H, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. arXiv preprint arXiv:1412.3555, 2014.
27. Fan C, Wang J, Gang W, et al. Assessment of deep recurrent neural network-based strategies for short-term building energy predictions[J]. Applied energy, 2019, 236: 700-710.
28. Bontempi G. Long term time series prediction with multi-input multi-output local learning[J]. Proc. 2nd ESTSP, 2008: 145-154.
29. Taieb S B, Bontempi G, Atiya A F, et al. A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition[J]. Expert systems with applications, 2012, 39(8): 7067-7083.
30. Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.
31. Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]//Advances in neural information processing systems. 2014: 3104-3112.
32. J. Weston, S. Chopra, and A. Bordes. Memory networks. arXiv:1410.3916, 2014.
33. Galassi A, Lippi M, Torroni P. Attention, please! a critical review of neural attention models in natural language processing[J]. arXiv preprint arXiv:1902.02181, 2019.
34. K. Cho, B. Van Merri□enboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259, 2014.
35. Xu K, Ba J, Kiros R, et al. Show, attend and tell: Neural image caption generation with visual attention[C]//International conference on machine learning. 2015: 2048-2057.
36. Allaire JJ, Chollet F and etc. Keras. Version 2.1.6. The Comprehensive R Archive Network; 2018.<https://keras.rstudio.com>.
37. Schuster M, Paliwal K K. Bidirectional recurrent neural networks[J]. IEEE transactions on Signal Processing, 1997, 45(11): 2673-2681.

38. Chen Y, Shi Y, Zhang B. Modeling and optimization of complex building energy systems with deep neural networks[C]//2017 51st Asilomar Conference on Signals, Systems, and Computers. IEEE, 2017: 1368-1373.
39. Reddy T A, Maor I, Panjapornpon C. Calibrating detailed building energy simulation programs with measured data—Part I: General methodology (RP-1051)[J]. Hvac&R Research, 2007, 13(2): 221-241.
40. Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors[J]. arXiv preprint arXiv:1207.0580, 2012.