sustainability

MDPI

Article

Comparative Assessment to Predict and Forecast Water-Cooled Chiller Power Consumption Using Machine Learning and Deep Learning Algorithms

Elsa Chaerun Nisa and Yean-Der Kuan *

Refrigeration, Air Conditioning and Energy Engineering Department, National Chin-Yi University of Technology, Taichung 41170, Taiwan; s4a716075@student.ncut.edu.tw

* Correspondence: ydkuan@ncut.edu.tw; Tel.: +886-4-239-24505 (ext. 8256)

Abstract: Over the last few decades, total energy consumption has increased while energy resources remain limited. Energy demand management is crucial for this reason. To solve this problem, predicting and forecasting water-cooled chiller power consumption using machine learning and deep learning are presented. The prediction models adopted are thermodynamic model and multi-layer perceptron (MLP), while the time-series forecasting models adopted are MLP, one-dimensional convolutional neural network (1D-CNN), and long short-term memory (LSTM). Each group of models is compared. The best model in each group is then selected for implementation. The data were collected every minute from an academic building at one of the universities in Taiwan. The experimental result demonstrates that the best prediction model is the MLP with 0.971 of determination (R^2), 0.743 kW of mean absolute error (MAE), and 1.157 kW of root mean square error (RMSE). The time-series forecasting model trained every day for three consecutive days using new data to forecast the next minute of power consumption. The best time-series forecasting model is LSTM with 0.994 of R^2, 0.233 kW of MAE, and 1.415 kW of RMSE. The models selected for both MLP and LSTM indicated very close predictive and forecasting values to the actual value.

Keywords: water-cooled chiller; prediction model; time-series forecasting model; machine learning algorithms; deep learning algorithms; power consumption

check for updates

Citation: Chaerun Nisa, E.; Kuan, Y.-D. Comparative Assessment to Predict and Forecast Water-Cooled Chiller Power Consumption Using Machine Learning and Deep Learning Algorithms. *Sustainability* **2021**, *13*, 744. https://doi.org/10.3390/su13020744

Received: 14 December 2020
Accepted: 8 January 2021
Published: 14 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Energy demand management is an important research area because energy resources are limited, while energy consumption is increasing due to ever-increasing industrial development, and rapid economic and population growth over the last decades [1–3]. The building sector consumes approximately 40% of total energy consumption. It has become the world's largest energy consumer [1,2]. Heating, ventilating, and air conditioning (HVAC) systems play a significant role in ensuring occupant comfort and are one of the main energy consumers in buildings [4]. Numerous researchers have performed research on how to reduce HVAC system energy consumption because more than half of the total building power is consumed by HVAC systems [5,6]. The chiller is an essential HVAC component that cools and dehumidifies the air in a wide variety of commercial, industrial, and institutional facilities. The chiller consumes 25–40% of the total amount of electricity in a building [6,7].

Predicting and forecasting power consumption are essential parts of energy management systems. Predicting power consumption is used to evaluate engine performance in advanced power control and optimization and helps building managers make enhanced energy efficiency decisions [8,9]. Forecasting power consumption is used to allocate electrical utility, safe and secure system operation, maintenance scheduling for energy savings, and also guidance for system energy optimization [1,2,10]. Prediction models can be divided into two categories: physical and data-driven. Physical power consumption modeling

needs to build complex physical models and must have detailed building information, widely using simulation tools such as EnergyPlus and eQuest. In contrast, data-driven models do not need to build complex physical models or have detailed building information. Moreover, it has high efficiency and accuracy [11]. Forecasting models can be divided into three categories: statistical, machine learning, and deep learning models. Statistical models rely heavily on historical data, which makes the model less accurate for long-term time-series forecasting if there is a lot of variability in the data. The machine learning models have received much attention because of their nonlinear feature mapping learning ability between input and output data. The deep learning models increase the number of hidden layers. This method is able to work very well in processing strong nonlinear data characteristics that have a high-level invariant structure [12–15].

The literature review revealed that many researchers predicted and forecasted energy consumption at a large scale: home [3], large office buildings type open-space [16–18], residential buildings [19–21], institutional buildings [9,22,23], provinces [24], or even an entire country [25]. Further, very few researchers focused on power consumption prediction and forecasting at the component or system level, especially in HVAC.

Sha et al. [11] predicted HVAC system energy consumption using machine-learning algorithms: support vector regression (SVR), artificial neural network (ANN), and multivariable linear regression (MLR). SVR and ANN have better performance than MLR. Pombeiro et al. [9] compared the assessment of three models: linear regression (LR), fuzzy logic, and neural networks to predict electricity consumption. Among the three models, the best models were the fuzzy systems and neural networks using occupancy related data. Wang et al. [12] proposed a novel approach based on long short-term memory (LSTM) for forecasting refrigerator periodic energy consumption. This method had better prediction performance than several traditional forecasting methods. Cam et al. [22] made an electrical demand forecast for supply fans, chillers, and cooling towers, and the total electric demand of the cooling system in a large institutional building based on the support vector machines (SVM) model. Sendra-Arranz et al. [26] proposed an LSTM model to forecast HVAC system power consumption one day ahead, situated at MagicBox, a real self-sufficient solar house with a monitoring system. Some studies used other deep learning algorithms for forecasting, but not at the component or HVAC system. MLP was used to forecast 24-h ahead short-term load. The proposed forecasting method was accurate for days with stable weather patterns with around 1–2% forecast error [27]. A one-dimensional convolutional neural network (1D-CNN) was used to forecast energy load at the individual building level. The 1D-CNN performed better than ANN and SVM [28]. MLP, 1D-CNN, and LSTM were compared to forecast the minutely residential power consumption [29]. Prediction models are also used for real-time early warning systems for sustainable and intelligent plastic film manufacturing [30]. In the last decade, the concept of Energy and Sustainability is represented by the term "Smart"; for example, some "Smart-Islands" have used renewable and sustainable sources for the energy needs of these countries [31].

The above references indicate that neural networks have better performance than other machine learning algorithms. The thermodynamic model that applies the LR algorithm has never been compared with a neural network in predicting HVAC power consumption. The MLP, 1D-CNN, and LSTM are three deep learning algorithms that have also never been compared in forecasting HVAC power consumption. The main contribution of this paper is that it compares the thermodynamic model that applies the LR algorithm with MLP as a neural network class to predict power consumption. It also compares three deep learning algorithms to forecast power consumption one-minute ahead. The models that have the minimum error in the test set were selected as the best model. The best prediction model is implemented to evaluate system performance and create maintenance schedules. Meanwhile, the best time-series forecasting model is implemented to provide advance warning for HVAC system emergency execution when the forecasted power consumption is abnormal, too high, or too low. Besides that, it has been implemented to determine which power source will be used. During the forecasted power consumption is peak load,

the power source is automatically changed from grid utility to battery bank to avoid the marginal cost of electricity. Then, during the forecasted power consumption is baseload, the power from the grid utility automatically charges the battery bank.

This paper is organized as follows. Section 2 provides an overview of the proposed approach, the basic concepts of machine learning and deep learning, and the explanation of the prediction and forecasting model. Section 3 demonstrates the experimental detail including equipment, software, hardware, and data, and demonstrates the analysis result. Section 4 presents our conclusion.

2. Methods

2.1. Overview of Proposed Approach

Figure 1 shows an overview of the proposed approach. Generally, there are four steps: data collection and preprocessing, building models, making the predictions and forecasting, and implementation, each step is described below:

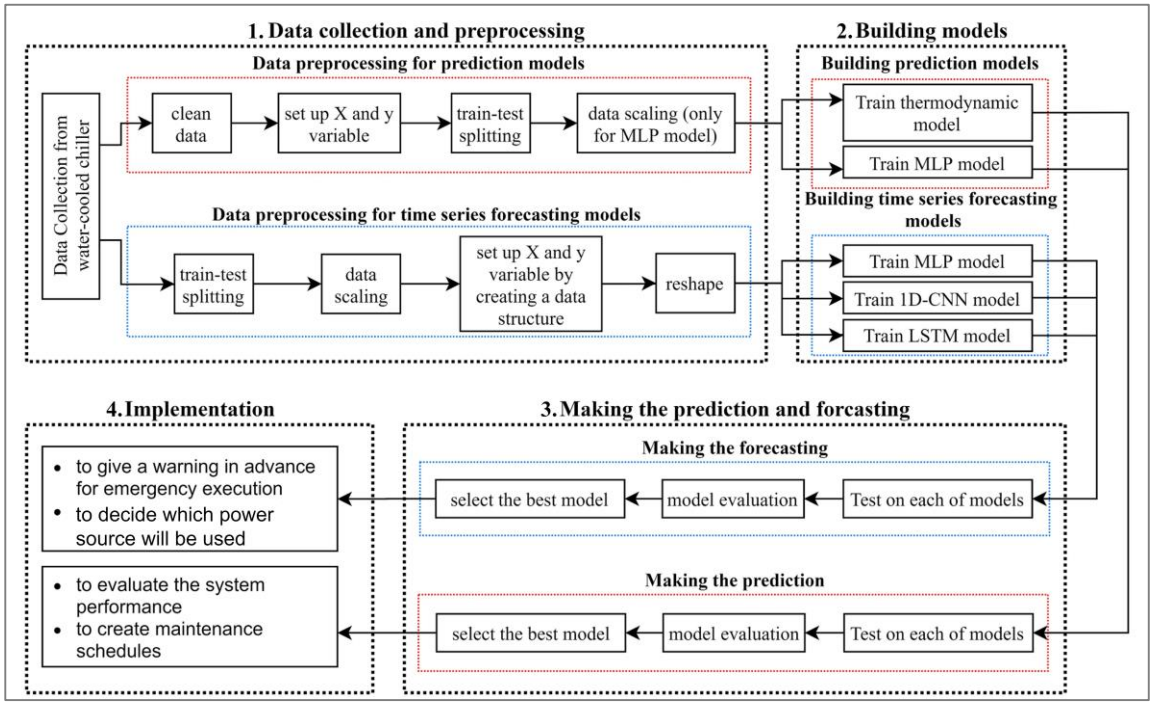


Figure 1. The overview of the proposed approach.

Step 1. Data collection and preprocessing. The data collections from the water-cooled chiller were categorized into two types, the data for prediction models and the data for time-series forecasting models. A more detailed explanation of the dataset is presented in Section 3.3. The standardization technique was used for the prediction models and the normalization technique was used for the time-series forecasting models. Equations (1) and (2) describe the standardization and normalization formulas, respectively. Where, $x_{stand}^{(i)}$, $x_{norm}^{(i)}$, $x^{(i)}$, $\overline{x}$, s, x_{min}, and x_{max} are the standardized data, normalized data, observed data, sample mean, standard deviation of the sample, the smallest value, and the largest value, respectively. The data was reshaped into three dimensions before it was trained using 1D-CNN and LSTM algorithms, while the data trained using MLP and LR algorithms were not reshaped.

$$x_{stand}^{(i)} = \frac{x^{(i)} - \overline{x}}{s} \tag{1}$$

$$x_{norm}^{(i)} = \frac{x^{(i)} - x_{min}}{x_{max} - x_{min}} \tag{2}$$

Step 2. Building models. Two types of models were built, the prediction model, and the time-series forecasting model. The prediction model refers to the algorithm output after it has been trained on a historical dataset and applied to new data. The time-series forecasting model is a sub-discipline of prediction, specifically to predict the future, based on time-series data. In this study, the prediction model was built to know the chiller power consumption in the same period as the input given. Meanwhile, the time-series forecasting model was built to forecast chiller power consumption in the future, one-minute ahead. The model parameter for each algorithm was defined before training. Hyper-parameter optimization was applied for deep learning algorithms.

Step 3. Making the prediction and forecasting. After the models had been trained, power consumption was predicted and forecasted by each model using the test set data. The prediction model is a static model, only trained once and used the test set data to make predictions. Meanwhile, the time-series forecasting model is a dynamic model. It was trained every day for three consecutive days using new data. The amount of training data was increasing day by day. Each trained model was tested using whole day data on the next day. Each model was evaluated using R^2, MAE, and RMSE. The models that had the minimum error in the test set were selected as the best model.

Step 4. Implementations. The best prediction model has been implemented to evaluate system performance and create maintenance schedules. Meanwhile, the best time-series forecasting model has been implemented to provide advance warning for emergency execution and to decide which power source will be used.

Section 2.2 gives basic machine learning and deep learning concepts. Sections 2.3 and 2.4 give detailed introductions for the prediction and time-series forecasting models respectively. Section 2.5 details the performance evaluations used in this study.

2.2. The Basic Concepts of Machine Learning and Deep Learning

Machine Learning is a subset of Artificial Intelligence (AI) that uses statistical methods and provides systems the ability to automatically learn and improve from experience. Deep learning is a particular kind of machine learning inspired by building and simulating the human brain neural network called neurons. Its concept comes from the artificial neural network [32].

LR is the simplest machine learning algorithm. It is a statistical model that predicts the relationship between two variables with the linear equation. MLP with multiple hidden layers is a kind of deep learning algorithm that can be used in prediction and time-series forecasting models [32]. 1D-CNN and LSTM are also kinds of deep learning algorithms used in the time-series forecasting model.

2.3. Power Consumption Prediction Model

2.3.1. Thermodynamic Model

Refer to ASHRAE Guideline 14-2002, a simple linear regression algorithm is applied to the thermodynamic model to predict the Coefficient of Performance (COP). Simple linear regression is a linear method used to simulate the relationship between one dependent variable (y) and one independent variable (x). The equation is defined as follows in Equation (3):

$$y = w_0 + w_1 x \tag{3}$$

where the weight w_0 is the y-axis intercept of the line and the weight w_1 is the slope of the line. The linear equation weight is learned by minimizing the sum of squares error (SSE), with the following Equation (4):

$$SSE = \sum_{i=1}^{n} \left(y^{(i)} - \hat{y}^{(i)} \right)^2 \tag{4}$$

where $y^{(i)}$ is the measured power consumption value in observation i, and $\hat{y}^{(i)}$ is the power consumption prediction using the regression model for observation i.

COP is a ratio between compressor energy usage ($P_{chiller}$) and the amount of useful cooling at the evaporator (Q_{evap}). Equation (5) is the formula to predict COP. Where, T_{cwR}, T_{chwS}, Q_{evap} are condenser water return temperature, chilled water supply temperature, and evaporator load.

$$\frac{1}{COP} = -1 + \left(\frac{T_{cwR}}{T_{chwS}}\right) + \frac{-A_0 + A_1(T_{cwR}) - A_2\left(\frac{T_{cwR}}{T_{chwS}}\right)}{Q_{evap}} \tag{5}$$

The coefficients A_0, A_1, and A_2 are found by making two linear regression models. The first linear regression model is using alpha (α) as a dependent variable and the temperature ratio (T_{cwR}/T_{chwS}, Kelvin) as an independent variable, and the second one is using beta (β) as a dependent variable and the condenser water return temperature (T_{cwR}, Kelvin) as an independent variable. The slope of the first linear regression model is the coefficient A_2. The intercept and the slope of the second linear regression model are the coefficients A_0 and A_1 respectively. The alpha (α) value is calculated using Equation (6), whereas COP is calculated using Equation (7). $P_{chiller}$ is the compressor power consumption gained from a measurement. The coefficient A_2 founded is used to find the beta (β) value calculated using Equation (8).

$$Alpha(\alpha) = \left(\frac{1}{COP} + 1 - \left(\frac{T_{cwR}}{T_{chwS}}\right)\right) Q_{evap} \tag{6}$$

$$COP = \frac{Q_{evap}}{P_{chiller}} \tag{7}$$

$$Beta(\beta) = \left(\frac{1}{COP} + 1 - \left(\frac{T_{cwR}}{T_{chwS}}\right)\right) Q_{evap} + A_2\left(\frac{T_{cwR}}{T_{chwS}}\right) \tag{8}$$

2.3.2. Multi-Layer Perceptron (MLP)

MLP is a type of ANN, which is inspired by neurons in our brain intended to imitate the way humans learn. MLP has the ability to solve nonlinear problems. Figure 2 shows the MLP architecture for the prediction model. The MLP structure used in this study consists of an input layer, two hidden layers, and an output layer. The input layer has six variables, and it is fed to the first hidden layer that has 10 neurons. The neuron output from the first hidden layer is fed as the input to the second hidden layer. The neuron output from the second hidden layer feeds into the output layer.

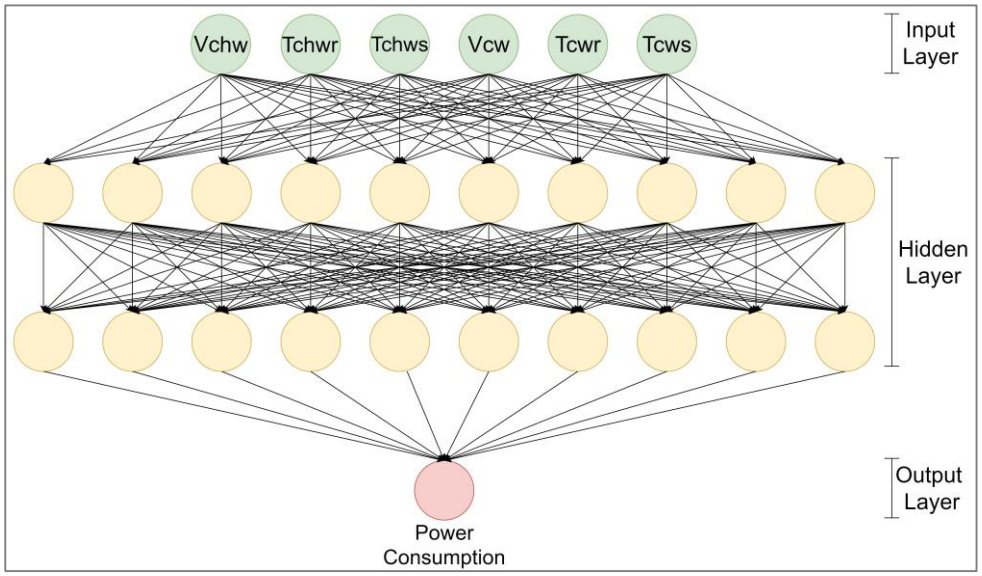


Figure 2. Multi-Layer Perceptron (MLP) architecture for the prediction model.

Equation (9) represents the neuron's output. Where x_i is the input variable, f is a nonlinear activation function, w_i and b_i are the weight and bias of the linear transformation. Rectifier linear unit (ReLU) is applied as an activation function in the hidden layers as represented in Equation (10), and the output layers do not apply the activation function.

$$y = f\left(\sum_i x_i w_i + b\right) \tag{9}$$

$$f(a) = \{0,\ if\ a < 0\ and\ a,\ if\ a \geq 0 \tag{10}$$

MLP models the relationship between inputs and outputs by learning from the recorded data. Mean squared error (*MSE*) is applied as a loss function. The formula of *MSE* is shown in Equation (11).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - P_i)^2 \tag{11}$$

2.4. Time-Series Forecasting Models

2.4.1. Multi-Layer Perceptron (MLP)

The MLP is used in time-series forecasting models as well as prediction models. To forecast the variable x at tth point in time, the preceding variable x is needed as an input. Figure 3 shows the MLP architecture for the time-series forecasting model. Four hidden layers are used in this study. An input layer represents a preceding p points in time $[X_{t-1}, X_{t-2}, \ldots, X_{t-p}]$. The input features are fed to the first hidden layer that has n neurons. The neuron output from the first hidden layer is fed as input to the next hidden layer, and so on until the last hidden layer. Finally, the last hidden layer output feeds into the output layer. The neuron output equation in the time-series forecasting model is the same as the neuron output in the prediction model as represented in Equation (9). The activation function is not applied in the hidden and output layers.

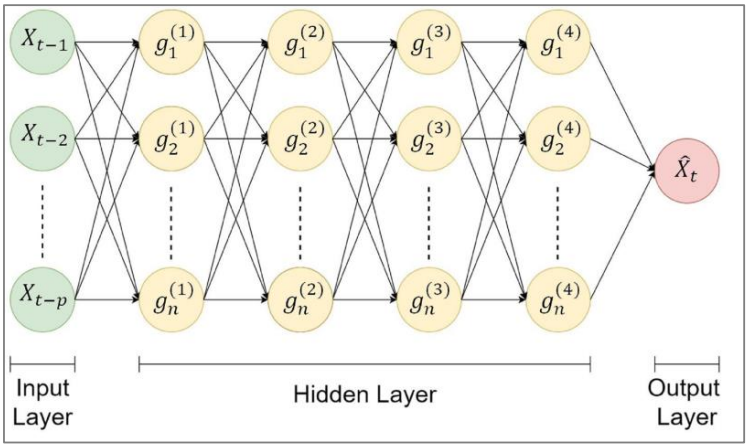


Figure 3. MLP architecture for the time-series forecasting model.

2.4.2. One-Dimensional Convolutional Neural Network (1D-CNN)

The Convolutional Neural Network (CNN) is traditionally used for images by extracting features from two-dimensional data. A similar architecture for one-dimensional time-series power consumption forecasting is used in this study. Figure 4 shows the 1D-CNN architecture for the time-series forecasting model. It consists of an input layer, convolutional layer, pooling layer, flattened layer, fully connected layer, and output layer. The input features are fed into the convolution layer. In the convolution layer, a filter is applied to an input feature to produce a feature map. The activation function is applied to the results. The output from the convolution layer feeds into the pooling layer to reduce the

size of the feature map. The pooled feature map is passed into a flattened layer to convert the data into a one-dimensional array for inputting it into the next layer. The flattened layer output is fed into the fully connected layer. In the fully connected layer, the weights are applied to process the data. The fully connected layer output is fed into the output layer. In this study, ReLU is applied in the convolution layer as an activation function. The activation function is not applied to the remaining layers.

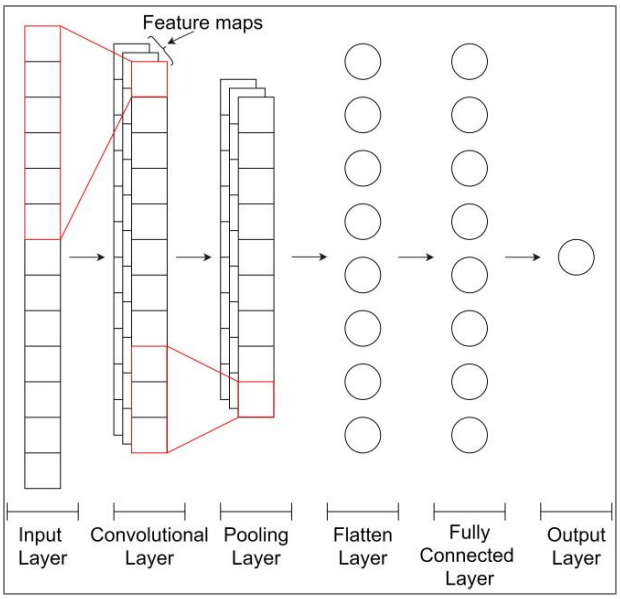


Figure 4. One-dimensional convolutional neural network (1D-CNN) architecture for the time-series forecasting model.

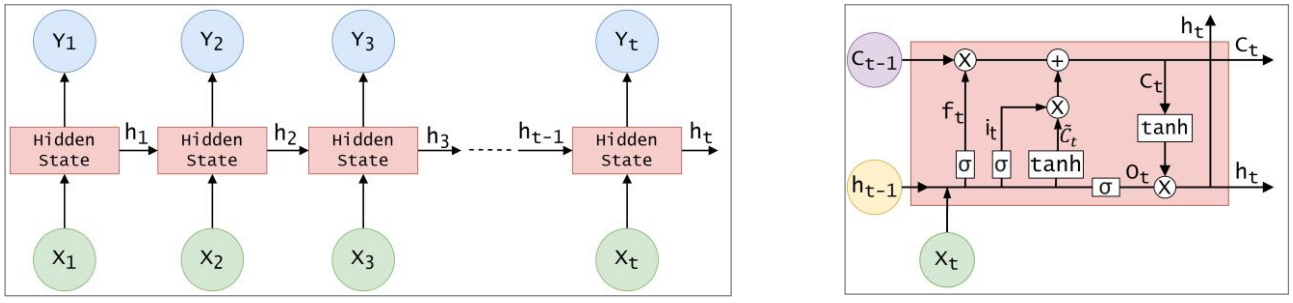
2.4.3. Long-Short Term Memory (LSTM)

The Recurrent Neural Network (RNN) is a type of neural network that uses the output of the previous steps as the current step input. All of the inputs are therefore related to each other. Figure 5a shows the RNN architecture, X_t is the input at time t that will be fed into the hidden state (h_t), and Y_t is the output. Equations (12) and (13) show the hidden state and RNN output formulas.

$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}X_t) \tag{12}$$

$$Y_t = W_{hy}h_t \tag{13}$$

LSTM is a modified version of RNN. The difference in architecture is only in the hidden state. The LSTM hidden state is described in Figure 5b. There are 4 gates: forget gate (f_t), input gate (i_t), cell gate (C_t), and output gate (o_t).



(**a**) (**b**)

Figure 5. The architectures for time-series forecasting model: (**a**) Recurrent Neural Network (RNN); (**b**) Long-Short Term Memory (LSTM).

Equations (14)–(19) represent the neuron output. Where h_t and C_t are the hidden layer vectors. x_t is an input vector. b_f, b_i, $b_{\widetilde{C}}$, b_o are the bias vectors. W_f, W_i, $W_{\widetilde{C}}$, W_o are the weights. Sigmoid (σ) and *tanh* are the activation functions. The neuron outputs in the hidden and output layers do not use an activation function.

$$f_t = \sigma(W_f \cdot [h_{t-1},\ X_t] + b_f) \tag{14}$$

$$i_t = tanh(W_i \cdot [h_{t-1},\ X_t] + b_i) \tag{15}$$

$$\widetilde{C}_t = tanh(W_{\widetilde{c}} \cdot [h_{t-1},\ X_t] + b_{\widetilde{c}}) \tag{16}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t \tag{17}$$

$$o_t = \sigma(W_o \cdot [h_{t-1},\ X_t] + b_o) \tag{18}$$

$$h_t = o_t \cdot tanh(C_t) \tag{19}$$

2.5. Performance Evaluation

The model performance is assessed using a variety of methods and metrics. The coefficient of determination (R^2), mean absolute error (MAE), and root mean square error (RMSE) are selected in this study as evaluation metrics to assess the performance of each model. These three metrics are formulated as in Equations (20)–(22):

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(Y_i - P_i)^2}{\sum_{i=1}^{N}\left(Y_i - \overline{Y}\right)^2} \tag{20}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|Y_i - P_i| \tag{21}$$

$$RMSE = \sqrt[2]{\frac{1}{N}\sum_{i=1}^{N}(Y_i - P_i)^2} \tag{22}$$

where, N, Y_i, P_i, and $\overline{Y}$, refer to the number of samples, measured value, predictive value, and average measured value, respectively. The first two evaluation metrics are scale-dependent, whereas R^2 is scale-independent, which can be used to evaluate the performance against other studies. The lower the MAE and RMSE values are, the better performance the model achieves. The R^2 value is usually between 0 and 1, but it is possible to have a negative value because the model can be arbitrarily worse.

3. Experiments and Result

3.1. Physical Equipment

The water-cooled chiller at an academic building in Taiwan was investigated in this study. A water-cooled chiller is a system that facilitates heat transfer from an internal environment to an external environment. The refrigerant condensing temperature in a water-cooled chiller is dependent on the condenser-water temperature, which is dependent on the ambient wet-bulb temperature. It is different from the refrigerant condensing temperature in an air-cooled chiller which is dependent on the ambient dry-bulb temperature. Since the wet-bulb temperature is often lower than the dry-bulb temperature, this means that the water-cooled chiller is more efficient because the compressor works and the energy consumption required is lower. Figure 6 depicts the water-cooled chiller diagram and measurement points. The black, red, and blue lines depict the refrigeration cycle, condenser water loop, and evaporator water loop, respectively. There are 7 measured data, condenser water supply temperature (T_{cws}), condenser water return temperature (T_{cwr}), condenser water velocity (V_{cw}), evaporator water supply temperature (T_{chws}), evaporator water return temperature (T_{chwr}), evaporator water velocity (V_{chw}), and power consumption ($P_{chiller}$). The power consumption data for this experiment were obtained from the compressor. The

compressor used on the chiller is a 2-stage semi-hermetic reciprocating compressor. Table 1 shows the compressor specifications.

Table 1. Compressor specifications.

Compressor Specifications	
Compressor model	S6F-30.2
Refrigerant	R22
Power Supply	230V–3–60Hz
Cooling Capacity	31.7 kW

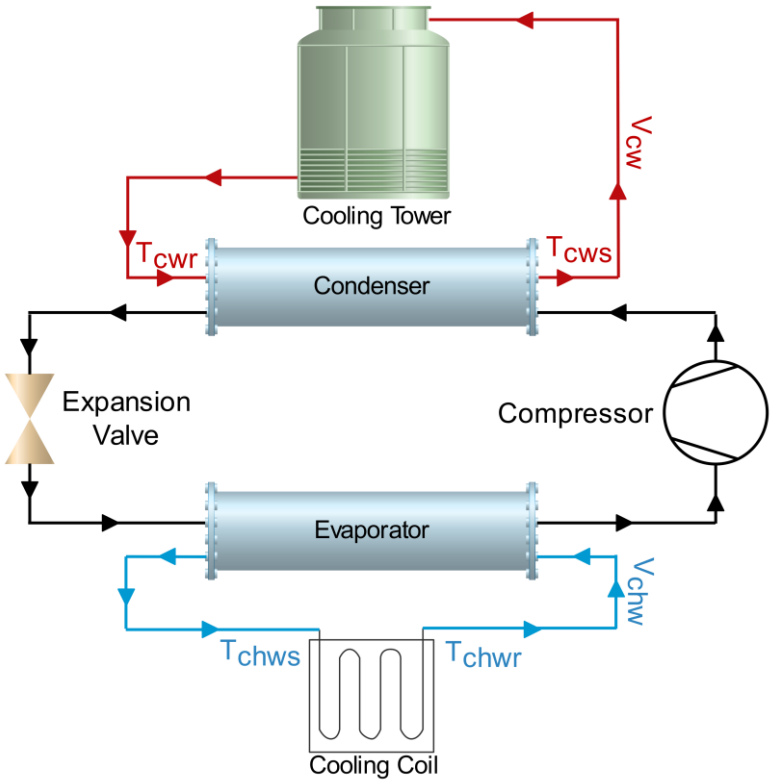


Figure 6. Water-cooled chiller diagram and the measurement points.

3.2. Software and Hardware

The software used for these experiments are Python 3.6.10, Tensorflow 2.1.0 version, Win10 Pro 64-bit Operating System, and *Keras deep learning package* [33] to implement the deep learning algorithm architectures. The hardware used are RAM 32GB, Intel(R) Core(TM) i7-6700 CPU.

3.3. Data Description

The data were collected every minute for 13 days from 20 November 2018 to 2 December 2018 from a water-cooled chiller. The chiller is running during the daytime and turns off during the nighttime. When the chiller starts-up, the power consumption is very high for several minutes before reducing to normal conditions. Two types of datasets were created, a prediction model dataset, and a time-series forecasting model dataset.

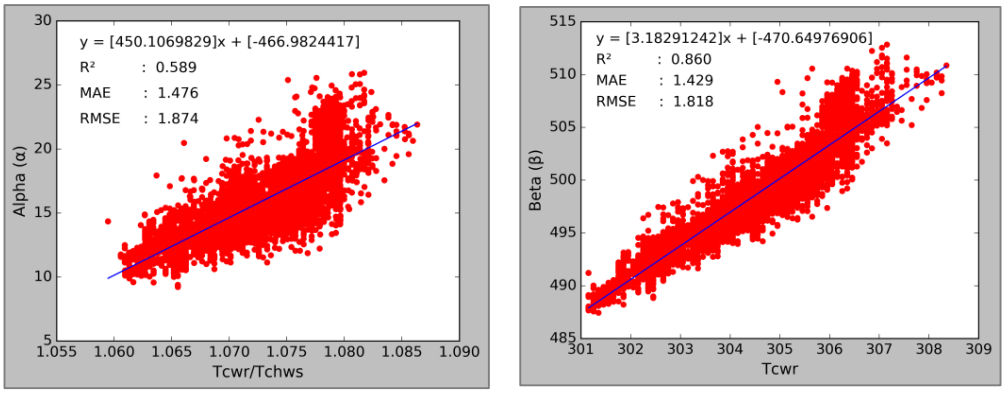
Since the prediction is carried out when the chiller is under normal conditions, only data under normal conditions is selected for the prediction model dataset, which consists of 7.238 data. Eighty percent of the data is used for training and 20% of the data is used for testing. Four measured data are used in the LR algorithm as the input, including T_{cwR}, T_{chwS}, T_{chwr}, and V_{chw}. While the MLP algorithm used two additional data, T_{cwS} and V_{cw}, so there are six inputs.

Since the forecasting is carried out one-minute ahead, one-minute observations over time are needed. The time-series forecasting models dataset is that all 13 days of data with one-minute intervals, which consists of 18.720 data. The model was trained three times with 10, 11, and 12 days of data. Each trained model was tested using whole day data on the next day. To forecast the data at time t, one day or 1440 previous data at time t is needed as input.

3.4. Power Consumption Prediction

3.4.1. Thermodynamic Model

The linear regression model is built using training set data to predict alpha and beta. Figure 7 shows the linear regression models (a) alpha prediction; (b) beta prediction. Each function of those models is obtained to predict the alpha and beta values. From those functions, the coefficient A_0 is 470.650, A_1 is 3.183, and A_2 is 450.107.



(**a**) (**b**)

Figure 7. Linear regression models (**a**) alpha prediction; (**b**) beta prediction.

When the coefficients A_0, A_1, and A_2 are found, the COP can be predicted using Equation (5) and calculate the power consumption prediction using Equation (7). Figure 8 shows the power consumption prediction using the thermodynamic model. The red one indicates the actual power consumption, while the blue one indicates the predicted power consumption using the thermodynamic model. From this figure, the thermodynamic model is inaccurate or too complex.

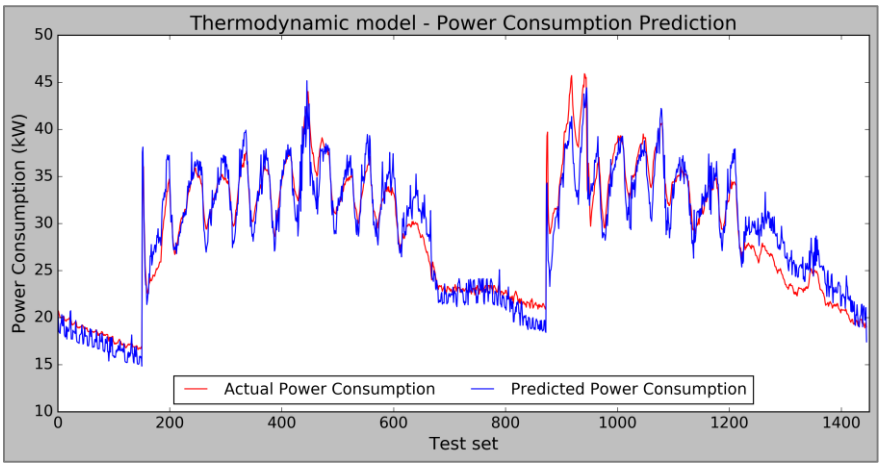


Figure 8. Power consumption prediction using the thermodynamic model.

3.4.2. Multi-layer Perceptron (MLP)

The MLP model is trained using the MSE as the loss function. The epochs are set to 300 and the selected activation function is ReLU. Hyper-parameter optimization using grid-search with five-cross-validation technique is applied to optimize the number of hidden layers, hidden neurons, and the batch size. The number of hidden layers is optimized ranging from 2 to 4 and the number of hidden neurons is optimized ranging from 5 to 25 with an increment of 5. The batch sizes used in parameter tuning are 32, 54, 64, and 128. After applied hyper-parameter optimization, the best hyper-parameters are obtained, there are two hidden layers with 10 hidden neurons, and the batch size number is 54. A dropout rate of 0.02 is applied to avoid overfitting after the first hidden layer. Figure 9 shows a comparison between the actual and predicted power consumption using the MLP model. The red one indicates the actual power consumption, while the blue one indicates the predicted power consumption using the MLP model. From that figure, the MLP model will be more successful when employed in practice.

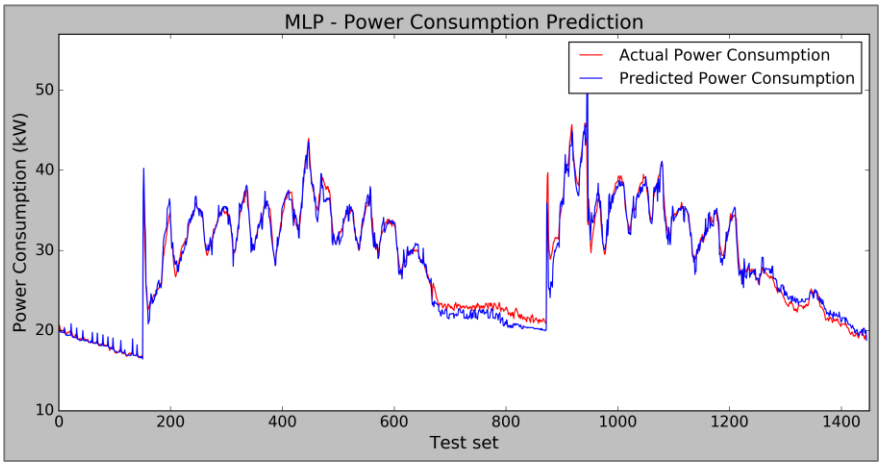


Figure 9. Power consumption prediction using the MLP model.

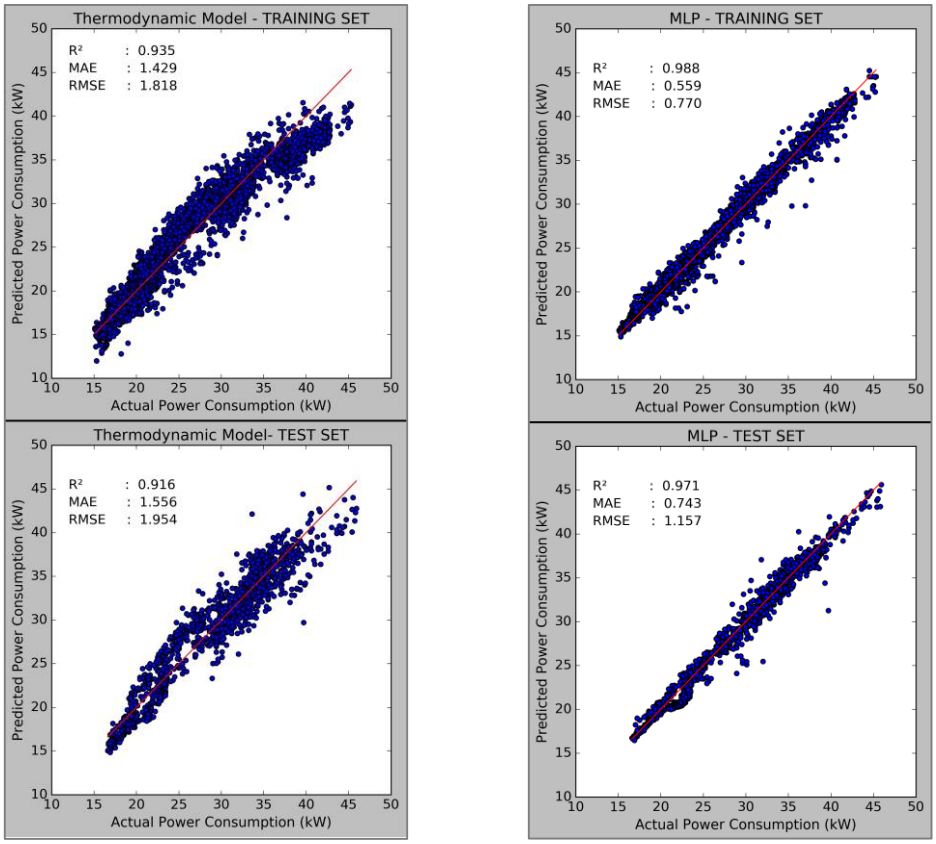
3.4.3. Power Consumption Prediction Model Performance Comparison

Figure 10 shows the scatter plots of training and testing results to compare the actual and predicted power consumption values. Figure 10 is a scatter plot between the prediction model outputs and the measured value of power consumption (a) Thermodynamic—training and test set; (b) MLP—training and test set. The X-axis is the actual power consumption obtained from the measurement. The y-axis is the predicted power consumption obtained from the model. From that scatter plot the MLP has better performance than linear regression in both training and testing.

Table 2 is the performance evaluation of the thermodynamic and MLP models. The MLP has better performance in both training and testing. Selecting the best model only considered the test set result. In the prediction models, the best performance is MLP, which has 0.971 of R^2, 0.743 kW of MAE, and 1.157 kW of RMSE.

Table 2. The thermodynamic and MLP models performance evaluation of.

Model		R^2	MAE (kW)	RMSE (kW)
Thermodynamic	Training	0.935	1.429	1.818
	Test	0.916	1.556	1.954
MLP	Training	0.988	0.559	0.770
	Test	0.971	0.743	1.157



(**a**) (**b**)

Figure 10. Scatter plots between the prediction model outputs and the measured value of power consumption. (**a**) Thermodynamic—Training and test set; (**b**) MLP—Training and test set.

3.5. Time-Series Forecasting Model

3.5.1. Three Deep Learning Algorithms

These experiments built three deep learning algorithms to forecast water-cooled chiller power consumption one-minute ahead. The model is trained three times using 10, 11, and 12 days of data. Each trained model is tested using whole day data after the training data. The input layer has 1440 neurons as a time step, and the output layer has one neuron. The models are trained using the mean absolute error as a loss function. The epochs are set to 50 and the batch size was set to 60. Since the time-series forecasting model computation time is quite long, a manual search technique is applied as the hyper-parameter optimization. The hyper-parameters are tune based on our assessment or experience. The hyper-parameters to be optimized are the number of hidden layers, hidden neurons for MLP and LSTM, and the number of feature maps for 1D-CNN. The number of hidden layers is optimized ranging from 1 to 5 with an increment of 1. The number of neurons and feature maps are optimized ranging from 10 to 50 with an increment of 10. The models are trained using the tuned hyper-parameters, evaluating the performance, and starting the process again. This loop is repeated until satisfactory performance is achieved. The best hyper-parameters are obtained after applying hyper-parameter optimization. All three algorithms are set to use 4 hidden layers. The MLP and LSTM hidden neurons are 30 and 40, respectively. The 1D-CNN feature maps are 40.

Figure 11 shows the time-series forecasting model error value using test set data on each day. Since the models are updated every day, the total data for the training set will increase and the error value decreases gradually except for the RMSE value in the MLP algorithm. The RMSE and MAE of LSTM are the lowest compared to other algorithms every day for both training and testing.

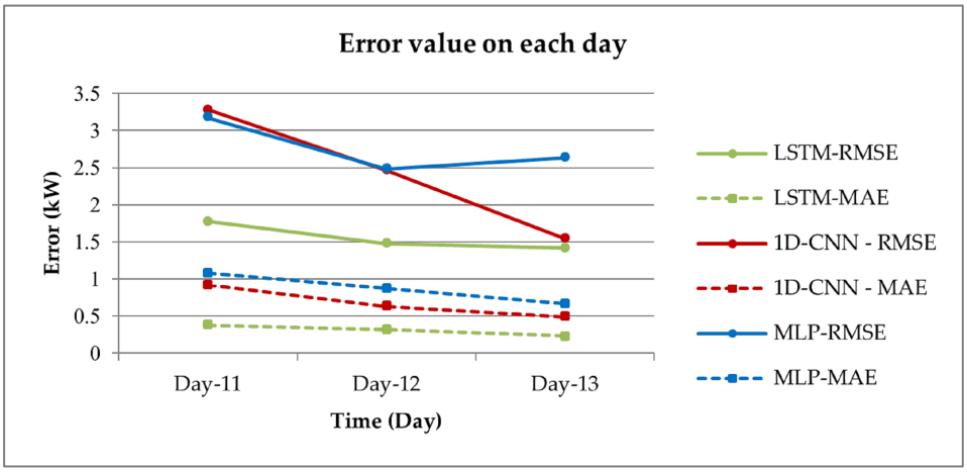


Figure 11. Error value of three deep learning algorithms on each day.

Figure 12 depicts a comparison between the actual power consumption and three deep learning models on day-13 test set data. The black line indicates the actual power consumption. The blue, red, and green lines indicate the forecasted power consumption using MLP, 1D-CNN, and LSTM, respectively.

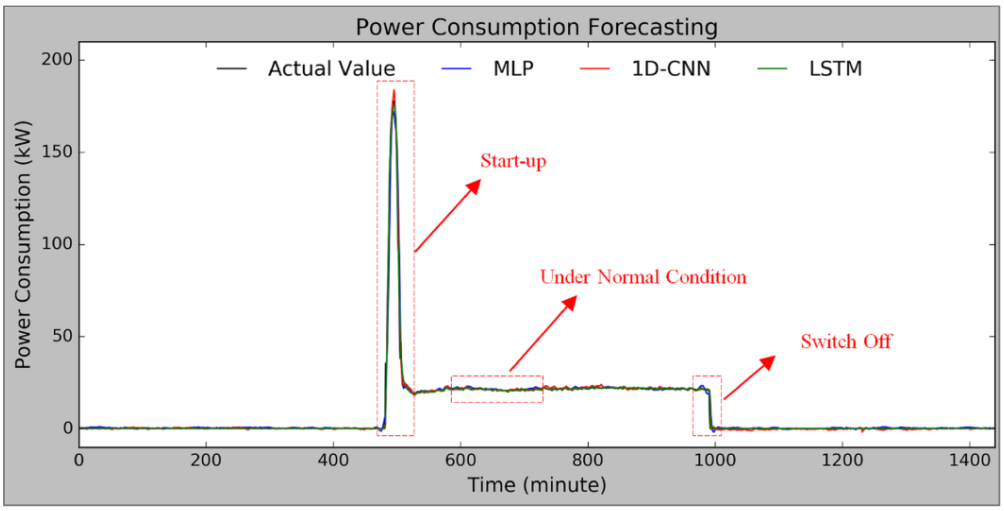


Figure 12. Comparison between the actual power consumption and three deep learning methods.

From that above figure, the results cannot be clearly seen. The clear result is when the chiller starts-up, under normal conditions, and the switch-off is plotted in the figure below. Figure 13 shows the comparison between the time-series forecasting models when the chiller is (a) started-up, (b) switched off, and (c) under normal conditions. When the chiller is started-up, all three algorithms forecast nearly the same value, with no significant differences. When the chiller is under normal condition and switched off, the LSTM looks more accurate than the 1D-CNN, and the 1D-CNN looks more accurate than the MLP.

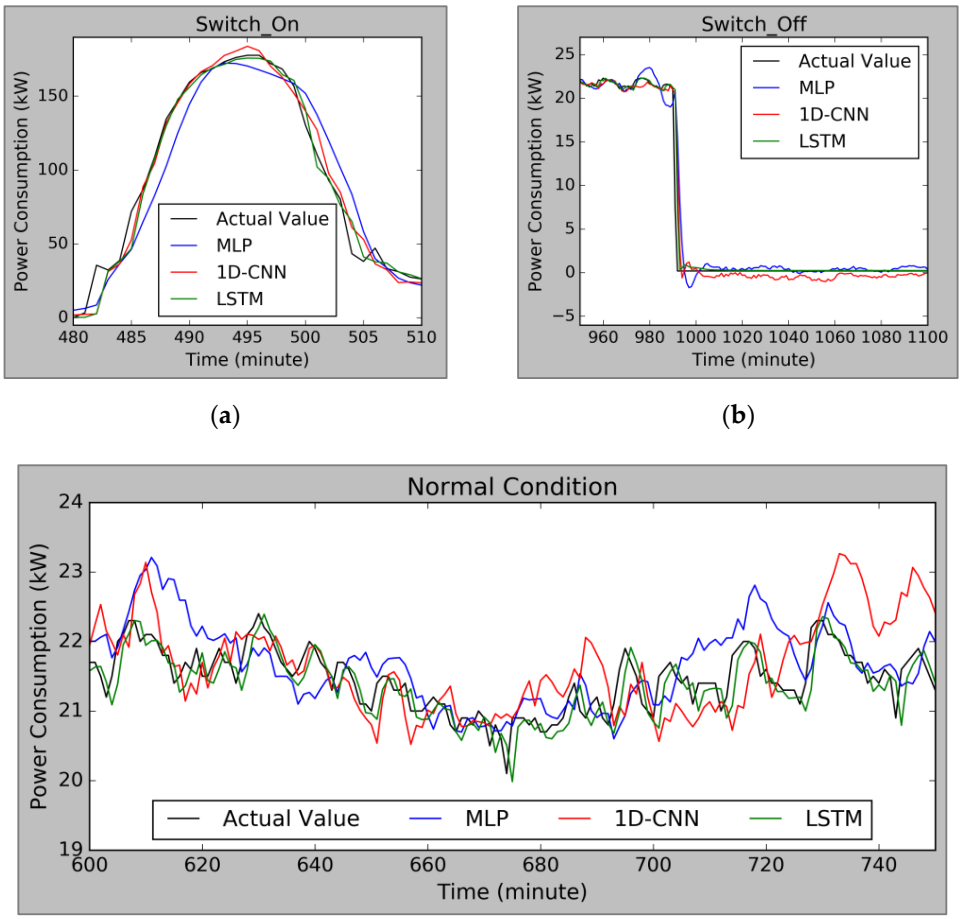


Figure 13. Comparison between time-series forecasting models when the chiller is (**a**) start-up, (**b**) switch off, and (**c**) under normal conditions.

3.5.2. Time-series Forecasting Model Performance Comparison

Figure 14 shows the scatter plots between the time-series forecasting models outputs and the measured power consumption value using 12 days training data and day-13 test set data (a) MLP—Training and test; (b) 1D-CNN—Training and test set; (c) LSTM—Training and test set. The X-axis is the actual power consumption obtained from the measurement. The y-axis is the forecasted power consumption obtained from the model. From those scatter plots, the LSTM has a better performance than 1D-CNN and MLP in both training and testing. However, the LSTM computation time is very long compared to the 1D-CNN and MLP.

Table 3 is the performance comparison between MLP, 1D-CNN, and LSTM algorithms using 12 days of training data and day-13 test set data. The LSTM has better performance in both training and testing. Selecting the best model considers only the result using the test set data. LSTM has 0.994 of R^2, 0.233 kW of MAE, and 1.415 kW of RMSE.

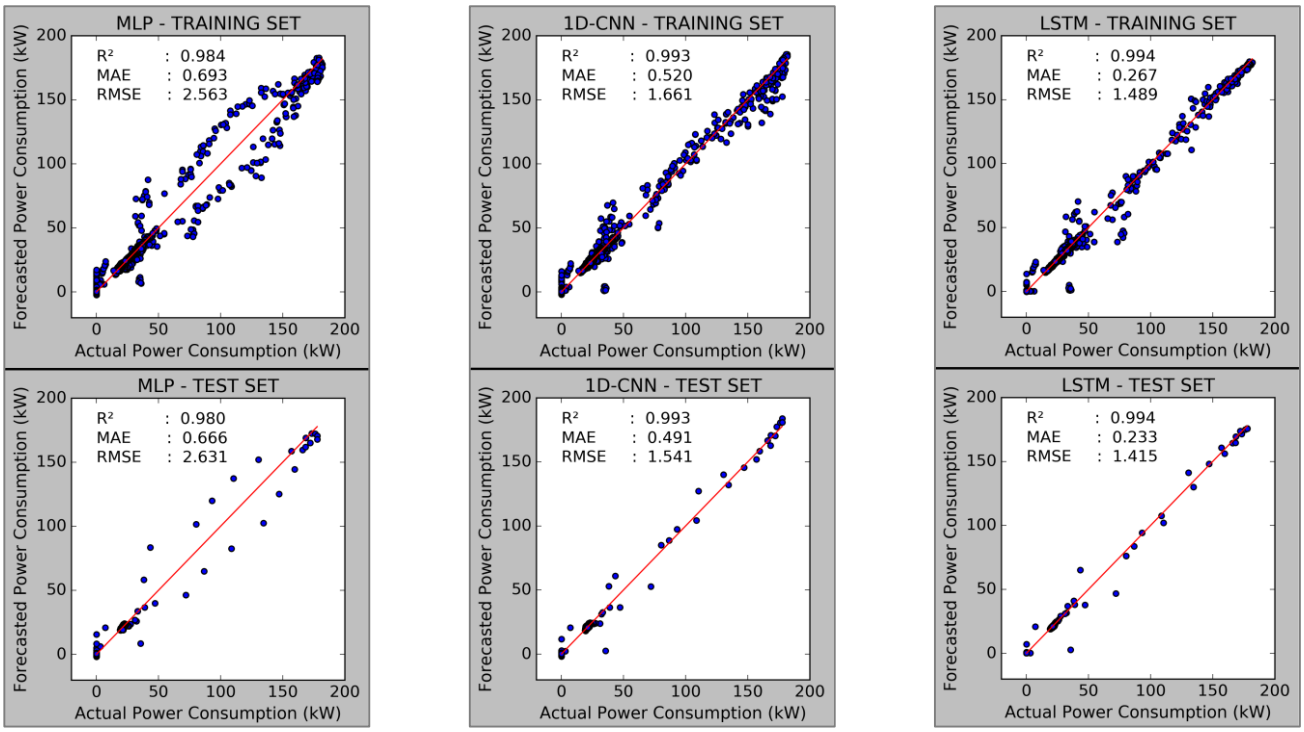


Figure 14. Scatter plots between the time-series forecasting model outputs and the measured value of power consumption using 12 days training data and day-13 test set data (**a**) MLP—Training and test set; (**b**) 1D-CNN—Training and test set; (**c**) long short-term memory (LSTM)—Training and test set.

Table 3. Performance comparison between MLP, 1D-CNN, and LSTM algorithms.

Model		R^2	MAE (kW)	RMSE (kW)
MLP	Training	0.984	0.693	2.563
	Test	0.980	0.666	2.631
1D-CNN	Training	0.993	0.520	1.661
	Test	0.993	0.491	1.541
LSTM	Training	0.994	0.267	1.489
	Test	0.994	0.233	1.415

4. Conclusions

Two types of models were built, prediction models and time-series forecasting models. The prediction model is a static model that predicts the chiller power consumption in the same period as the input data, which is trained once and uses the test set data to make predictions. While the time-series forecasting model is a dynamic model that forecasts chiller power consumption in the next minute from the input data, trains every day for three consecutive days using new data, and tested using a whole day data on the next day. The aim of this study is to compare the two prediction models: a thermodynamic model that applies the LR algorithm with the MLP algorithm; and to compare the three forecasting models: MLP, 1D-CNN, and LSTM. The data was obtained from the water-cooled chiller at the academic building at one of the universities in Taiwan. Four inputs, T_{cwR}, T_{chwS}, T_{chwr}, and V_{chw} are used to build a thermodynamic model. The input used by the thermodynamic model and two additional input data, T_{cwS} and V_{cw} are used as the inputs to build the MLP model for prediction. Three deep learning models use one feature, power consumption, and 1440-time steps as inputs. Hyper-parameter optimization, either a grid-search or manual search technique, is applied to define the best hyper-parameters before the training process. The MLP model is more successful at predicting compared to the thermodynamic model,

is selected as the best prediction model with 0.971 of R^2, 0.743 kW of MAE, and 1.157 kW of RMSE. The LSTM model has the longest computation time but the best performance compared to 1D-CNN and MLP. The error value in the LSTM model decreased every day in both the training set and the test set and is the smallest compared to the other models. The LSTM is selected as the best time-series forecasting model with 0.994 of R^2, 0.233 kW of MAE, and 1.415 kW of RMSE on the last day. Both the R^2 value from the MLP and LSTM indicate very close predictive and forecast values to the actual values. The MLP is implemented to evaluate system performance and create maintenance schedules. The LSTM is implemented to provide advance warning for emergency execution of HVAC system if the forecasted power consumption is abnormal, too high, or too low. Besides that, it is implemented to determine which power source will be used. The power source automatically changing from grid utility to the battery bank during the forecasted power consumption is the peak-load to avoid the marginal electricity cost, and automatically charged the battery bank during the forecasted power consumption is the baseload.

Author Contributions: Conceptualization, Y.-D.K.; methodology, E.C.N.; software, E.C.N.; validation, E.C.N. and Y.-D.K.; formal analysis, E.C.N. and Y.-D.K.; investigation, E.C.N. and Y.-D.K.; resources, Y.-D.K.; data curation, Y.-D.K.; writing—original draft preparation, E.C.N.; writing—review and editing, Y.-D.K.; supervision, Y.-D.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Ministry of Science and Technology of Taiwan, grant number MOST 109-2221-E-167-006.

Data Availability Statement: The data are not publicly available.

Acknowledgments: The authors thank the Ministry of Science and Technology of Taiwan.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

AI	Artificial intelligence
ANN	Artificial neural network
COP	Coefficient of performance
HVAC	Heating, ventilating and air conditioning
LR	Linear regression
LSTM	Long short-term memory
MAE	Mean absolute error
MLR	Multi-variable linear regression
MSE	Mean squared error
ReLU	Rectifier linear unit
RMSE	Root mean square error
RNN	Recurrent neural network
SSE	Sum of square error
SVM	support vector machine
SVR	Support vector regression
1D-CNN	One-dimensional convolutional neural network
α	Alpha
β	Beta
b	Bias
$b_{\widetilde{C}}$	Bias for represent candidate for cell state
b_f	Bias for forget gate
b_i	Bias for input gate neurons
b_o	Bias for output gate neurons
C_t	Cell state (memory)
C_{t-1}	Cell state (memory) from the previous block

$\widetilde{C}_t$	Represent candidate for cell state at timestamp t
f	Nonlinear activation function
f_t	Forget gate
h_t	Hidden state at time t
h_{t-1}	Output from the previous block
i_t	Input gate
N	Number of samples
o_t	Output gate
$P_{chiller}$	Compressor energy usage
P_i	Predictive value
Q_{evap}	Evaporator load
R^2	Coefficient of determination
s	Standard deviation
T_{chwR}	Evaporator water return temperature
T_{chwS}	Chilled water supply temperature
T_{cwR}	Condenser water return temperature
T_{cwS}	Condenser water supply temperature
V_{chw}	Evaporator water velocity
V_{cw}	Condenser water velocity
$W_{\widetilde{C}}$	Weight for represent candidate for cell state
W_f	Weight for forget gate neurons
W_{hh}	Weight at the recurrent neuron
W_{hy}	Weight at the output neuron
W_i	Weight for input gate neurons
W_o	Weight for output gate neurons
W_{xh}	Weight at the input neuron
w_i	Weight for neurons
w_1	Weight indicates the slope value
w_0	Weight indicates the intercept value
X_t	Input at current step
x	Independent variable
x_i	Input variable
$x^{(i)}$	Observed data
$\overline{x}$	Sample mean
x_{max}	Largest value of observed data
x_{min}	Smallest value of observed data
$x^{(i)}_{norm}$	Normalized data
$x^{(i)}_{stand}$	Standardized data
Y_i	Measured value
Y_t	Output state
$\overline{Y}$	Average measured value
y	Dependent variable or neuron output
$y^{(i)}$	Measured value in observation i
$\hat{y}^{(i)}$	Predicted value for observation i
σ	Sigmoid function

References

1. Somu, N.; Raman, M.R.G.; Ramamritham, K. A hybrid model for building energy consumption forecasting using long short term memory networks. *Appl. Energy* **2020**, *261*, 114131. [CrossRef]
2. Liu, T.; Xu, C.; Guo, Y.; Chen, H. A novel deep reinforcement learning based methodology for short-term HVAC system energy consumption prediction. *Int. J. Refrig.* **2019**, *107*, 39–51. [CrossRef]
3. Kim, T.-Y.; Cho, S.-B. Predicting residential energy consumption using CNN-LSTM neural networks. *Energy* **2019**, *182*, 72–81. [CrossRef]
4. Nasruddinn, N.; Sholahudin, S.; Satrio, P.; Mahlia, T.M.I.; Giannetti, N.; Saito, K. Optimization of HVAC system energy consumption in a building using artificial neural network and multi-objective genetic algorithm. *Sustain. Energy Technol. Assess.* **2019**, *35*, 48–57. [CrossRef]
5. Xiong, W.; Wang, J. A semi-physical static model for optimizing power consumption of HVAC systems. *Control Eng. Pract.* **2020**, *96*, 104312. [CrossRef]

6. Sala-Cardoso, E.; Delgado-Prieto, M.; Kampouropoulos, K.; Romeral, L. Predictive chiller operation: A data-driven loading and scheduling approach. *Energy Build.* **2020**, *208*, 109639. [CrossRef]
7. Yu, J.; Liu, Q.; Zhao, A.; Qian, X.; Zhang, R. Optimal chiller loading in HVAC System Using a Novel Algorithm Based on the distributed framework. *J. Build. Eng.* **2020**, *28*, 101044. [CrossRef]
8. Ahmad, M.W.; Mourshed, M.; Rezgui, Y. Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption. *Energy Build.* **2017**, *147*, 77–89. [CrossRef]
9. Pombeiro, H.; Santos, R.; Carreira, P.; Silva, C.; Sousa, J.M.C. Comparative assessment of low-complexity models to predict electricity consumption in an institutional building: Linear regression vs. fuzzy modeling vs. neural networks. *Energy Build.* **2017**, *146*, 141–151. [CrossRef]
10. García-Sanz-Calcedo, J.; Gómez-Chaparro, M. Quantitative analysis of the impact of maintenance management on the energy consumption of a hospital in Extremadura (Spain). *Sustain. Cities Soc.* **2017**, *30*, 217–222. [CrossRef]
11. Sha, H.; Xu, P.; Hu, C.; Li, Z.; Chen, Y.; Chen, Z. A simplified HVAC energy prediction method based on degree-day. *Sustain. Cities Soc.* **2019**, *51*, 101698. [CrossRef]
12. Wang, J.Q.; Du, Y.; Wang, J. LSTM based long-term energy consumption prediction with periodicity. *Energy* **2020**, *197*, 117197. [CrossRef]
13. Qing, X.; Niu, Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy* **2018**, *148*, 461–468. [CrossRef]
14. Wang, H.; Lei, Z.; Zhang, X.; Zhou, B.; Peng, J. A review of deep learning for renewable energy forecasting. *Energy Convers. Manag.* **2019**, *198*, 111799. [CrossRef]
15. Pal, A.; Prakash, P.K.S. *Practical Time Series Analysis*; Packt: Birmingham, UK, 2017; pp. 161–204.
16. Zhong, H.; Wang, J.; Jia, H.; Mu, Y.; Lv, S. Vector field-based support vector regression for building energy consumption prediction. *Appl. Energy* **2019**, *242*, 403–414. [CrossRef]
17. Cannistraro, G.; Cannistraro, M.; Restivo, R. Some Observations on The Radiative Exchanges Influence on Thermal Comfort in Rectangular Open-Space Environments. *Int. J. Heat Technol.* **2015**, *33*, 79–84. [CrossRef]
18. Cannistraro, G.; Cannistraro, M.; Restivo, R. The Local Media Radiant Temperature for the Calculation of Comfort in Areas Characterized by Radiant Surfaces. *Int. J. Heat Technol.* **2015**, *33*, 115–122. [CrossRef]
19. Wen, L.; Zhou, K.; Yang, S. Load demand forecasting of residential buildings using a deep learning model. *Electr. Power Syst. Res.* **2020**, *179*, 106073. [CrossRef]
20. Zhang, G.; Guo, J. A novel ensemble method for hourly residential electricity consumption forecasting by imaging time series. *Energy* **2020**, *203*, 117858. [CrossRef]
21. Kim, M.; Jung, S.; Kang, J. Artificial Neural Network-Based Residential Energy Consumption Prediction Models Considering Residential Building Information and User Features in South Korea. *Sustainability* **2019**, *12*, 109. [CrossRef]
22. Le Cam, M.; Zmeureanu, R.; Daoud, A. Cascade-based short-term forecasting method of the electric demand of HVAC system. *Energy* **2017**, *119*, 1098–1107. [CrossRef]
23. Kim, M.K.; Kim, Y.-S.; Srebric, J. Predictions of electricity consumption in a campus building using occupant rates and weather elements with sensitivity analysis: Artificial neural network vs. linear regression. *Sustain. Cities Soc.* **2020**, *62*, 102385. [CrossRef]
24. He, Y.; Zheng, Y.; Xu, Q. Forecasting energy consumption in Anhui province of China through two Box-Cox transformation quantile regression probability density methods. *Measurement* **2019**, *136*, 579–593. [CrossRef]
25. Hu, H.; Wang, L.; Peng, L.; Zeng, Y.-R. Effective energy consumption forecasting using enhanced bagged echo state network. *Energy* **2020**, *193*, 116778. [CrossRef]
26. Sendra-Arranz, R.; Gutiérrez, A. A long short-term memory artificial neural network to predict daily HVAC consumption in buildings. *Energy Build.* **2020**, *216*, 109952. [CrossRef]
27. Markou, M.; Kyriakides, E.; Polykarpou, M. 24-Hour Ahead Short Term Load Forecasting Using Multiple MLP. In Proceedings of the International Workshop on Deregulated Electricity Market Issues in South-Eastern Europe, Nicosia, Cyprus, 22–23 September 2008.
28. Amarasinghe, K.; Marino, D.L.; Manic, M. Deep neural networks for energy load forecasting. In Proceedings of the 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE), Edinburgh, UK, 19–21 June 2017; pp. 1483–1488.
29. Kontogiannis, D.; Bargiotas, D.; Daskalopulu, A. Minutely Active Power Forecasting Models Using Neural Networks. *Sustainability* **2020**, *12*, 3177. [CrossRef]
30. Kim, J.; Hwangbo, H. Real-Time Early Warning System for Sustainable and Intelligent Plastic Film Manufacturing. *Sustainability* **2019**, *11*, 1490. [CrossRef]
31. Cannistraro, G.; Cannistraro, M.; Trovato, G. Islands "Smart Energy" for eco-sustainable energy a case study "Favignana Island". *Int. J. Heat Technol.* **2017**, *35*, S87–S95. [CrossRef]
32. Wang, P.; Fan, E.; Wang, P. Comparative Analysis of Image Classification Algorithms Based on Traditional Machine Learning and Deep Learning. *Pattern Recognit. Lett.* **2020**, S0167865520302981. [CrossRef]
33. Keras. The Python Deep Learning API. Available online: https://keras.io/ (accessed on 17 September 2020).