

ANN-based thermal load prediction approach for advanced controls in building energy systems

Byeongmo Seo¹, Yeo Beom Yoon¹, Suwon Song², Soolyeon Cho^{1,*}

¹North Carolina State University, Raleigh, NC

²Korea Institute of Civil Engineering and Building Technology, Ilsan, South Korea

ABSTRACT: The Artificial Neural Network (ANN) technology has been used in various areas. In the building industry, however, ANN is relatively less utilized due to its complexity and uncertain benefits of its application along with the costs associated with its development. This paper introduces ANN regarding its applicability and potential benefits in building operations, especially for energy savings. Thermal loads calculations are most widely used for the operation of building energy systems. An ANN model was developed to predict a large office building's cooling loads. The EnergyPlus simulation program was used to generate thermal loads data and the Python program to develop an ANN model. The initial ANN model predicted a case study building's cooling loads within the CVRMSE value of 7.3% initially, and later 6.8% after optimization, which is within the tolerance range of 30% recommended by the ASHRAE Guideline 14. This study showed the potential benefit of energy savings that can be achieved by utilizing the ANN model for accurately predicting the cooling loads.

KEYWORDS: ANN, Optimal Control, Python, Load Prediction, Learning Method

INTRODUCTION

Since the Industrial Revolution, energy usage has increased as global urbanization progresses rapidly. Researches are continuously carried out in various fields to improve energy efficiency. As such, researches on the energy conservation in both new and existing buildings have been actively carried out. In the past, high-efficiency Heating, Ventilation, and Air-Conditioning (HVAC) systems, high-performance windows, and high-insulation walls have been the focus of research (Seo, 2017, Seo and Lee, 2016). However, most of these were mostly focused on new building design and remodeling cases. In the case of remodeling, the implementations are usually high. It is required first to maximize the efficiency of the existing systems by developing optimal control strategies. Various studies on the optimal control methods of HVAC systems were conducted utilizing advanced technologies such as Artificial Intelligence (AI) controls. Artificial Neural Network (ANN) is one of the advanced AI technologies that can learn and predict future behavior of buildings based on historical data with appropriate variables. ANN-based control methods are increasingly utilized these days to solve complex problems in various fields due to its ability to learn and analyze mapping relationships including non-linear phenomena (Abiodun et al., 2018).

Recent research trends show that the ANN controls are applied to the energy systems operations in buildings. Kang et al. Implemented an ANN-based prediction model to find the optimal supply air setpoint temperature of the Variable Refrigerant Flow (VRF) system. The input variables of the ANN were indoor, outdoor temperature and humidity, cooling loads, condenser water flow rate, condenser water temperature, supply air temperature, and cooling energy consumption. Comparing the predicted value of ANN with the actual value, the Coefficient of Variance of Root Mean Squared Error (CVRMSE) was 10.3%, and the ANN-based control method saved about 28% of cooling energy (Kang et al., 2018). Reynolds et al. constructed a model combining ANN and Genetic Algorithms (GAs) to reduce building energy consumption. In order to predict the energy consumption and indoor air temperature, the training was performed by specifying the weather data, the number of occupants, and the

indoor temperature as input variables. The energy savings of 25% was observed utilizing the ANN model over the fixed-temperature control method (Reynolds et al., 2018). Afram et al. conducted an energy performance evaluation of ANN-based model predictive control against the fixed-temperature control method of residential buildings. Their results showed that the ANN-based predictive control model consumed up to 10% more cooling energy than the fixed-temperature control method in July. This is because the weather is very extreme in July and the residential building cannot hold enough cooling to shift the load to off-peak hours. However, it confirmed that the heating energy was used 70% less than the fixed-temperature control method in October (Afram et al., 2017). Deb et al. conducted a study to predict the cooling loads of public institution buildings using ANN to achieve energy savings. Two-year energy consumption data of three buildings were analyzed, and the ANN model was constructed based on outdoor temperature, humidity, solar radiation, and energy consumption variables (Deb et al., 2016). Mba et al. conducted a study to predict the room temperature and humidity in the past using ANN to reduce cooling energy in residential buildings. Two years of indoor and outdoor temperature and humidity data were collected. The ANN model predicted the indoor air temperature and relative humidity with the accuracy of about 98% (Mba et al., 2016). Many other ANN-based optimal control studies have been reported. However, most studies had focused on identifying the results of applying ANN. These studies, however, do not clearly articulate on how the AI and ANN-based control models can be developed, applied, and utilized in detail. This paper focuses on explaining the general procedure of developing the ANN-based load predicting models for the future use of optimal control of building energy systems. A case study is used to show an example of the ANN model application and its accuracy.

2.0 ARTIFICIAL NEURAL NETWORK (ANN)

2.1. Machine learning (ML) methods

There are three types of basic ML methods: 1) supervised learning, 2) semi-supervised learning, and 3) unsupervised learning. Supervised learning requires an input data set and the same number of correct answer data sets. Semi-supervised learning requires a smaller amount of correct data sets than the input data sets. Unsupervised learning uses only the input data sets. The learning method is selected according to the purpose of ML. Supervised learning is mainly used for classification, prediction and regression analyses. Semi-supervised learning is used for clustering and classification. Unsupervised learning is used for clustering, visualization, and feature extraction (Muller and Guido, 2017). Supervised learning, for example, can mainly be used for lighting on/off control, load prediction, and load calculation per unit area. Semi-supervised learning can be used to identify the amount of clothing through image analysis. Unsupervised learning can be used to classify patterns of building users. Among these, the supervised learning method is used for ANN.

ANN was proposed by Warren McCulloch and Walter Pitts, which is based on human neurotransmission structures and learning processes. When ANN is applied to the study of building thermal environment, it performs Predictive controls and Adaptive controls. Predictive control predicts the future thermal environmental conditions and system parameter values to find a way of enhancing both the thermal comfort and building energy performances. Adaptive control adapts itself by continuous self-learning, which has an advantage of outputting an accurate and stable control value by itself (Geron, 2017).

Supervised learning requires both 'input' and 'correct answer' data sets. The 'correct answer' datasets here mean the results that the ANN model generates for a given input data sets. For example, the case in a model for predicting the cooling loads, the input data sets are the outdoor dry bulb temperature, outdoor relative humidity, and solar radiation; the ANN model outputs are the predicted cooling loads. The learning process in the supervised learning is a process of updating the weight factors so that the error is reduced by comparing the output (or predicted) values of the ANN model with the 'correct answer' values. Therefore, when the specific data sets are input, the ANN model, which has been well learned by the supervised

learning, processes the patterns of the input values and generates the calculation (or predicted) results, which is determined to be closest to 'correct answer' data.

2.2. Structure of artificial neural network (ANN)

Figure 1-(a) shows a representative multi-layer neural network model of ANN. An ANN model consists of an input layer, one or more hidden layers, and an output layer. Figures 1-(b) and (c) show the step function and the sigmoid function, which are kind of Activate functions. There are a variety of Activate functions, but in this paper, we will only describe two of those commonly used. The Activate function is used to simulate the behavior of biological neurons. The difference between the step function and the sigmoid function is that the result is different. The step function calculates 0 or 1 as the result value, and the sigmoid function calculates the value between 0.00 and 1.00 as the result value. When designing the ANN model, it is essential to select the Activate function depending on the application. For example, the step function is used for a linearly separable problem such as an on/off control, and the sigmoid function is used for a non-linearly separable problem such as load predictions and energy consumption predictions (Kyurkchiev and Markov, 2015).

Each Neuron is connected by weight factors as shown in Figure 1-(a). Each neuron adds the product of the input values and the weight factors that are connected and pass this value to the input value of the Activate function. Also, if the input values passed to the previously calculated Activate function are not large enough to exceed a certain threshold, there will be no outputs. Conversely, if the input value given to the active function exceeds the threshold, the neuron is activated to transmit the data to the next step. In summary, we can say that ANN computes the results by analyzing the interaction between data, weight factors, and the Activate function (Moon, 2015).

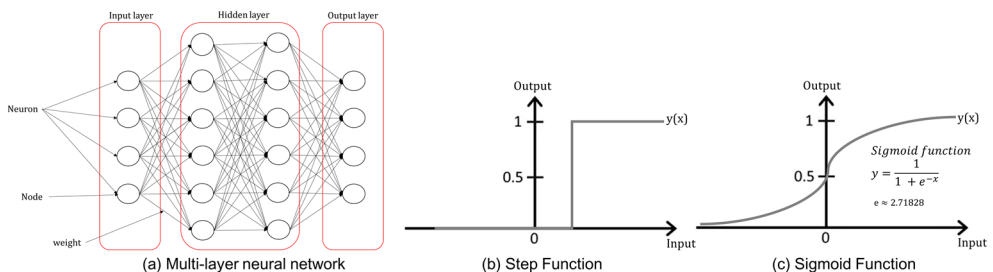


Figure 1: ANN structure and Activate function

2.3. Example of ANN training method

Figures 2 and 3 are simplified illustrations of the ANN learning process shown in Figure 1-(a), which are divided into four stages to facilitate understanding. As discussed, supervised learning refers to a learning method that pairs 'input' and correct 'answer data,' where the 'correct answer' data is the value that the ANN model should generate as outputs for a given 'input' data sets. In Figure 2-(a), "Input_#" means the input value, $w_{a,b}$ the weight factors between the nodes a and b, and node (neuron) a point at which data is gathered. Answer_# means 'correct answer' output to input data. Output_# is the calculated (or predicted) value from the ANN model. As discussed, the learning process is a process of updating the weight factors of the model, so that the error is reduced by comparing the output value of the ANN model for the 'Input' with the 'correct answer' of the held 'Input' data. Therefore, the learning is divided into feed-forward propagation to calculate the output of the ANN model and back-propagation to calculate the error and update the weight factors by reflecting on the learning process (Rashid, 2016).

Figure 2 shows the feed-forward propagation that sequentially transmits the data. Figure 2-(a) shows the initialization of the weight factors. Figure 2-(b) illustrates the process of calculating the results by multiplying the sum of the values input to each node by the Activate function.

The reason for choosing the weight factors randomly at the first stage of the ANN is that the optimal weight factors combination is different depending on the input data. In addition, since the neural network finds the optimal weight factors through the repeated learning process, it is common to randomly select all the weight factors except for setting all the weight factors to zero or setting them all to the constant of the same values.

In Figure 2-(a), Input_1 is 0.8, Input_2 is 0.6, 'Answer' and weight factors were randomly assigned. The Activate function was applied to the sigmoid function, and the bias and learning rate were not considered because they did not affect the learning process. Figure 2-(b) is a diagram of feed-forward propagation for calculating the output of the ANN model. As shown in Figure 2-(b), the input values to Output node 1 are Input 1 $\times w_{1,1}$ and Input 2 $\times w_{2,1}$, and the sigmoid function outputs 0.74 by using 1.06 that is the sum of input values. Also, Output node 2 outputs 0.61 through the same procedure.

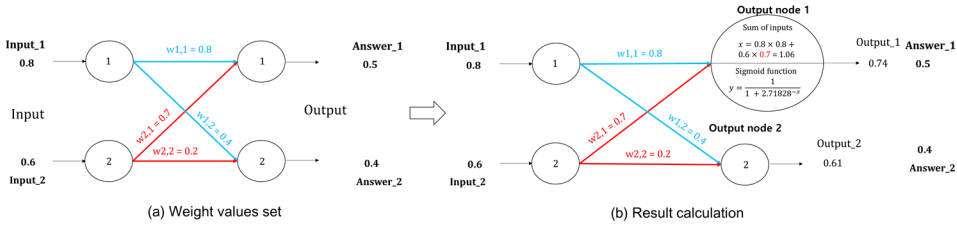


Figure 2:

Figure 3 shows the back-propagation process for updating the weight factors. Figure 3-(a) shows the back-propagation process of calculating the 'error' which is the difference between the outputs of the ANN and 'correct answer,' and updating the weight factors to reduce the error. Figure 3-(b) shows that the weight factors are finally updated. We need to adjust the Activate function or weight factors to reduce the error. However, since adjusting the Activate function requires a relatively complicated task, most studies adjust the weight factors. In Figure 3-(a), the ANN outputs 0.74 and 0.61. They are compared with the answers 0.50 and 0.40, resulting in an error of -0.24 and -0.21. These errors are called e_x , which are used for back-propagation. For example, we input -0.24 to the e_1 in the formula $\frac{w_{1,1}}{w_{1,1} + w_{2,1}} \times e_1$ and $\frac{w_{2,1}}{w_{1,1} + w_{2,1}} \times e_1$ which update the weight factors $w_{1,1}$, $w_{2,1}$ which are connected to the output node1. As discussed, the ultimate goal of the ANN is to converge the error closer to zero. Thus we adjust the weight factors to reduce it in the learning process. Output_1 is affected by $w_{1,1}$ and $w_{2,1}$. Also, e_1 , the error value for Output_1, affects only $w_{1,1}$ and $w_{2,1}$ in the back-propagation process. In this case, the denominator in the above equation is the sum of the weight factors, and the two equations are the same, but the numerator is different by $w_{1,1}$ and $w_{2,1}$. The sum of $\frac{w_{1,1}}{w_{1,1} + w_{2,1}}$ and $\frac{w_{2,1}}{w_{1,1} + w_{2,1}}$ is always 1, and the larger the weight factors is more affected by e_1 . Thus, when back-propagation proceeds, the more error rate is reflected in the

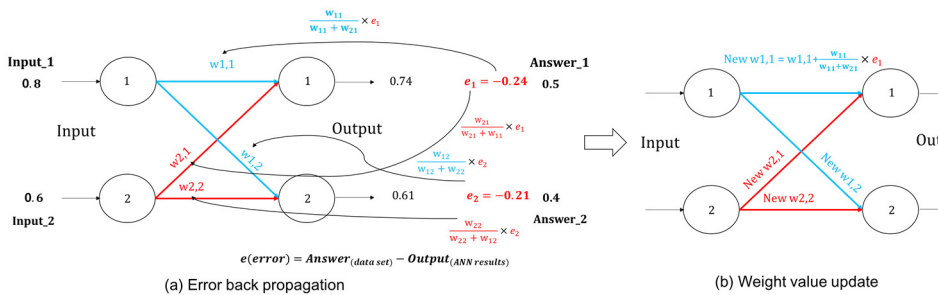


Figure 3: Back-propagation process

weight factors having a relatively large value. As a result, the value reflecting the error rate is added to the existing weight factors as shown in Figure 3-(b), and the learning ends. After this process, we have completed the first learning. In the second learning, we do not initialize the weight factors, rather we repeat the process of Figure 2-(b) and Figure 3-(a) up to the number of learning iterations.

2.4. Fundamental functions of ANN

In order to implement ANN models, there are at least three functions required (Rashid, 2016).

- Initialization: variables, number of hidden, output nodes, number of hidden layers, l earning rate, bias value setting
- Training: learning through training data and updating weight factors accordingly
- Test: outputs (or predicted values)

Besides, one additional task is required to apply the ANN to the building control field (Kang et al., 2018). First, the initialization part for selecting variables, the training part for learning based on the selected variables, and the Test & Optimization part for verifying and optimizing the ANN-based prediction models that have completed the learning. The last one is the control logic part. Through this process, ANN predicts some values, which is applied to building control logic for finding optimal control values. The control value is input to building control logic such as HVAC to apply ANN to building control. Figure 4 shows the development process of the ANN-based prediction model using Python (Muller and Guido, 2017).

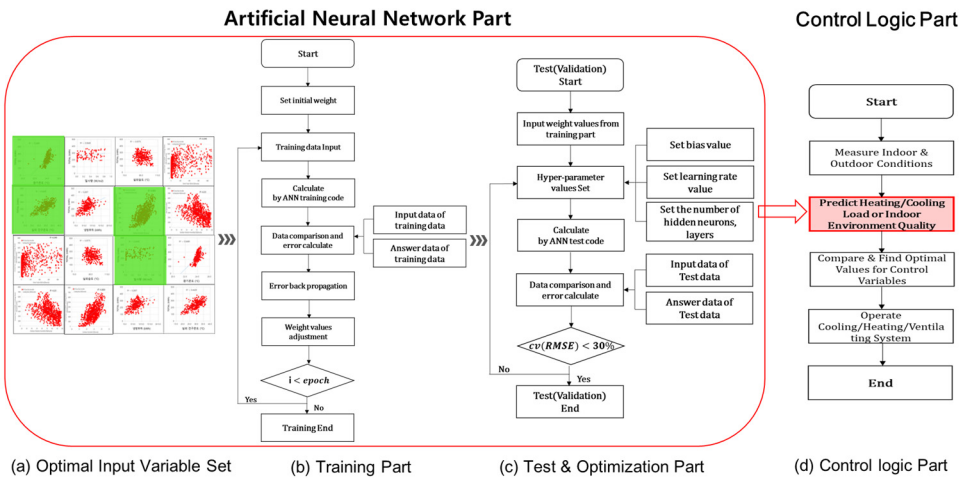


Figure 4: ANN-based load prediction logic

The first step to construct an ANN-based prediction model is to select variables with a high correlation with the output to be predicted. The reason for analyzing the correlation of variables is to select variables through objective indicators to improve the efficiency of training. Figure 4-(a) shows the input variable selection process. For example, when constructing a model for predicting cooling loads of a building, it is important to select the input variables with a relatively high correlation by analyzing r^2 , which represents the correlation between all the input variables that can be calculated and the cooling loads. If a variable with low correlation between input and output variables is selected, it is difficult to expect a proper prediction value even if repeated training is performed. The next step is data collection. This is because the ANN model is less adaptive when training with too few data. The sigmoid function used in this study requires a data range from 0.00 to 1.00. If you enter a larger input value, the ANN becomes saturating, which does not work properly. Therefore, through the normalization that divides the data range of the cooling load to be predicted through the selected input data and

the ANN by the correlation analysis, it should be adjusted from 0.00 to 0.99 or -1 to +1 (Rashid, 2016).

Figure 4-(b) is a flow chart showing the training part, the second part of the ANN-based prediction model. When we start training, we initialize the weight factors which are the heart of the learning. Next, add the sum of the values obtained by multiplying the input data and weight factors by each node, and bias. Input this value into the Activate function to output values (Rashid, 2016). The error rate between the result of the ANN and the 'correct answer' is reflected in the next training, and the error rate reduction training is repeated by the value of the epoch. Epoch means the number of training iterations. Again, the goal of learning is to obtain the weight factors at which the error rate between the output of the ANN and the 'correct answer' becomes the lowest.

Figure 4-(c) is a flow chart showing the test part that the third part of the ANN-based prediction model. In order to confirm the adaptability of the ANN model that has been trained through the training part, the task of confirming the error rate between the ANN output and the 'correct answer' of the verification data through the verification data, which is new data that was not experienced during the learning process, is called a test data or verification data. Unlike the previous learning part, the test process does not adjust the weight factors according to the error rate between the ANN results and the 'correct answer' and confirms the prediction accuracy rate of the ANN model that completed training process. Also, the optimization of ANN means to adjust the values of bias, learning rate, the number of hidden neurons, and the number of hidden layers, which are Hyper-parameters of the ANN model. This process is repeated until getting an acceptable error rate.

The test and optimization processes are; first, after the training part, the values of optimized weight factors and Hyper-parameters are specified in the same way as the training part. The test data is input to output the result of ANN. The next step is to confirm the predictive performance and reliability of the ANN model. To confirm these, a statistical term or Coefficient of Variance of the Root Mean Square Error (CVRMSE) is used with the acceptable tolerance ranges suggested in the American Society of Heating, Refrigerating, and Air-Conditioning Engineers (ASHRAE) Guideline 14. The tolerance ranges (or error rates) mean the prediction accuracy of the ANN model compared to the 'correct answer' data. The CVRMSE value closer to zero % means that the better prediction performance. ASHRAE recommends the CVRMSE value of less than 30% as the error rate tolerance of time data. It includes the recommended r^2 value, which indicates correlation, to be 0.80 or higher (ASHRAE, 2002). If the CVRMSE value is more than 30%, the user adjusts the Hyper-parameter values and performs the optimization process by repeating a series of operations until the tolerance value becomes less than 30%, or a specific value (or goal) is reached.

There are many ways to utilize ANN technology in buildings. Typically, it can be used for optimal control to reduce energy consumption. The ANN-based optimal control uses the ANN results to the control logic part to calculate the optimal control value that uses the least amount of energy and control the building through it without compromising human comfort. This process is shown in Figure 4-(d). Also, through the ANN, optimal control can be achieved by controlling the AHU discharge temperatures, fan speeds, and chilled water temperatures by effectively predicting cooling, heating, and ventilation loads. An ANN-based optimal AHU discharge temperature control, for example, can predict the cooling loads through ANN and applies it to the control logic to calculate the optimal discharge temperatures that use the least energy under the predicted cooling loads. In addition, it can be used in areas where control is available, such as blind controls and lighting controls

3.0 EXAMPLE OF ANN FOR LOADS PREDICTION

3.1. Tools (EnergyPlus and Python)

There are two software programs used in this study; EnergyPlus ver. 8.9 for data (or thermal loads) generation and Python ver. 3.6 for developing ANN. EnergyPlus is a simulation program

developed by The US Department of Energy, which combines the merits of BLAST in the loads analysis part and the advantages of DOE-2 in the system analysis part. In addition, EnergyPlus uses the heat balance calculations recommended by ASHRAE (USDOE, 2018). Python is an interpreted language developed by Amsterdam's Guido Van Rossum in 1990. Python is used in practice as well as for educational purposes. Typical examples are Google's software programs and Dropbox. Python has been used across the entire spectrum of social computing, including web programming, numerical computation, data analysis, object-oriented programming, graphical user interface (GUI) programming, system utility building, software development (Lutz and Ascher, 2013). NumPy and SciPy are important libraries to construct ANN models in Python. NumPy is a core Python package for performing scientific calculations, especially for N-dimensional matrix calculations. With NumPy, there is a big advantage that ANN's complex learning process can be quickly solved with simple coding. SciPy is used to deal with more complex problems such as integration, finding eigenvectors of sparse matrices, and checking the consistency of distributions using NumPy arrays. Those are mainly used for processing tasks such as optimization and data fitting (Bressert, 2013).

3.2. Simulation Modeling

A large number of datasets are required to construct an ANN-based model for loads predictions. The building thermal loads data were generated using the "RefBldgLargeOfficeNew2004" model available in the EnergyPlus program as an example. The Chicago weather condition was used. The analysis period was a cooling season (June 1 to August 31). Figure 5-(a) shows the building model and building schedule for a large office building. The model was a basement and three-story office building with an area of 46,320 square meters and Window to Wall ratio of 38%. The ASHRAE Standard 90.1-2004 was used for the selection of building materials, constructions, and window properties. The U-value of windows is 3.236 W/m²-K with the Solar Heat Gain Coefficient (SHGC) of 0.39. The U-value of the outer wall is 0.698 W/m²-K, roof U-value of 0.358 W/m²-K, and the floor U-value of 2.193 W/m²-K. To simplify the test, only the middle layer (2nd floor) data was used. Also, weekday data is only used. The room thermostat is set to 24 °C from 7 AM to 10 PM and 26.7 °C in the other hours. The data only from 07:00 to 22:00 was used with the office temperature settings of 24 °C. Figure 5-(b) shows the lighting, equipment, and occupancy schedules. All the schedules reach 90% or more during the office hours, and it can be seen that the occupancy and the equipment schedules are low only at lunchtime. In night time, the device and lighting are operated at a minimum level.

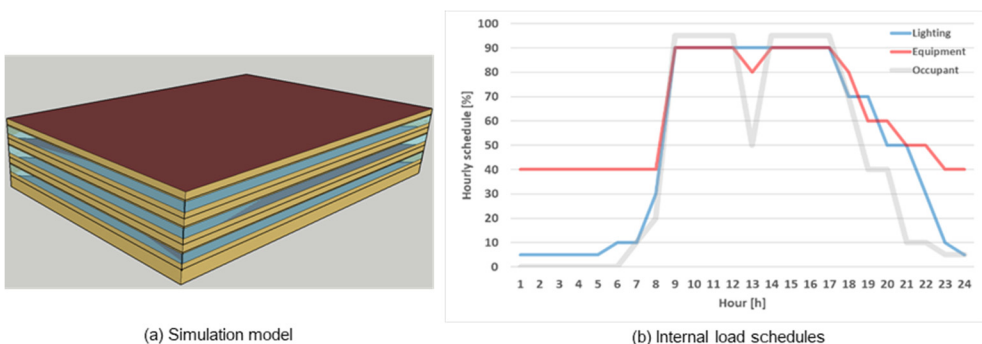


Figure 5: Simulation modeling & internal load

3.3. Selection of input variables for ANN

The datasets are divided into 1) learning and 2) verification data. The training data was only used in the training part and test data used in the test part. It is because if the same data set is used in both the training and test parts, the ANN can predict well in specific conditions, but it may not predict well when a new pattern data is used. Therefore, we implemented the Training and Test datasets by randomly dividing the EnergyPlus simulation results at 9:1 ratio to confirm the ANN adaptability to the new patterns that were not experienced in the Training

part. In addition, input variables were selected by four variables with relatively high R^2 values through correlation analysis with the cooling loads as follows: Site Outdoor Air Drybulb Temperature [C] (Hourly), Site Diffuse Solar Radiation Rate per Area [W/m²] (Hourly), Site Direct Solar Radiation Rate per Area [W/m²] (Hourly).

3.4. ANN model for loads prediction

Figure 6-(a) shows the structure of the cooling load prediction model constructed for the example case. The structure of the initial model had four input nodes, two hidden layers, and 12 hidden nodes. It has one output node, and the sigmoid function was used as Activate function. The learning rate was set at 10%, which means that only 10% of the error is used to update the weight factors when the error of 1 occurs. The reason for not fully reflecting 100% of the error was that the optimal weight factors were set by using the gradient descent method. This method has advantages when it has to cope with a function with multiple parameters. This method uses a step-by-step approach to find optimal answers, so choosing a learning rate that is too high or too low will not lead to reaching the value that minimizes the error. Bias is a variable that controls how easily a neuron is activated. Epoch refers to the number of repetitions of learning. For example, when learning is performed with Epoch 100 with 1000 training data, 1000 x 100 times weight factors update is completed. Datasets to be used for ANN must match the range from 0.00 to 1.00 as required by the sigmoid function. Therefore, the range of the simulation result data from 0.00 to 1.00 was adjusted by dividing the simulation result data by the maximum value of each variable through normalization. Initial weight factors were arbitrarily specified. Figure 6-(b) is a graph comparing the 'correct answer' with the predicted value of the initial ANN model. The results showed that the CVRMSE of the cooling loads predicted by the ANN model was 7.34% compared to the "correct answer (measured data; however, simulated data in this example case)," which is within the acceptable tolerance range of 30% recommended in the ASHRAE Guideline 14.

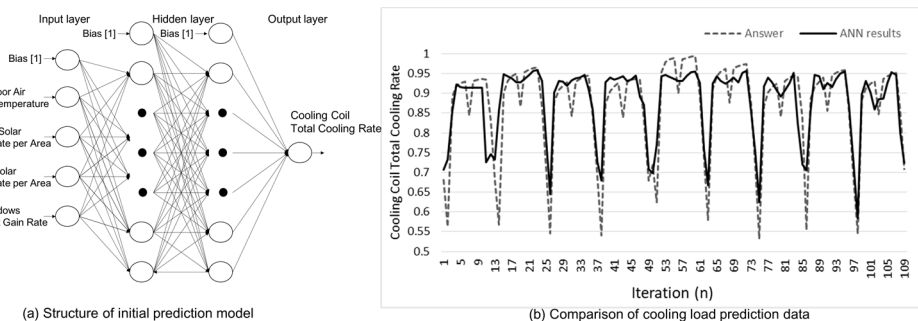


Figure 6: Structure of Initial ANN model and result comparison

3.5. ANN model optimization

ANN model optimization refers to the process of finding an optimal learning rate, hidden nodes, hidden layers, and epochs to optimize prediction performance. There are no known steps to determine the values of a Hyper-parameter for any problem. Currently, the best approach is to check the results and adjust the learning rate, hidden nodes, hidden layers, and epoch values by the trial and error processes. It is repeated until the error rate comes into the acceptable uncertainty ranges proposed in the ASHRAE Guideline 14. Since the optimal parameter values vary depending on the type of data, optimization tasks can take a long time depending on the experience and expertise of the user. In this study, about ten times of optimization trials were conducted to achieve the goal of model optimization. Table 1 includes the Hyper-parameter values with the lowest CVRMSE and the possible input ranges. After the optimization process, the CVRMSE value decreased to about 6.8%. It can be confirmed that the error rate is lower than that of the earlier model although the simplified optimization is performed. The results indicate that the ANN model developed for loads prediction in the example case can be used for the optimal control of providing only necessary cooling energy to the spaces without compromising human comfort.

Table 1: Optimal ANN structure and parameter values

Division	Range	Optimize values
Number of Hidden Layers	1~n	2
Number of Hidden Neurons	1~n	10
Learning Rate	0.01~1.00	0.05
Epochs	1~n	1,000

CONCLUSION

Many studies on AI-based building control methods have been conducted. However, most studies focused on identifying the results of applying ANN, not a detailed explanation of how they used and built the AI-based building control or optimization models.

This research focused on explaining the general procedure of developing the ANN-based load predicting models for the future use of optimal control of building energy systems to activate AI-based research. As part of AI technology, the ANN structure and learning methods were discussed along with the way on how to apply the ANN-based prediction model to building control field, including on how the ANN-based control models can be developed, applied and utilized in detail. To be specific, an ANN-based load prediction model was developed to predict the cooling loads in an example case of a large office building in Chicago. The results of the example study showed that the ANN technology has a large potential for energy savings as ANN models can be developed for the optimal control of building energy systems that require thermal loads to be able to provide the necessary energy to spaces.

Our following research will include the optimal control of building using the ANN-based load prediction model developed through this study. Detailed application methods and the possibility of ANN-based optimal control will be further discussed. Advanced studies will be conducted to compare and analyze the general control methods and ANN-based optimal control methods.

ACKNOWLEDGEMENT

This study was supported by the R&D Project (No. 20180576-001) of the Korea Institute of Civil Engineering and Building Technology.

REFERENCES

Abiodun, OI., Jantan, A., Omolara, AE., Dada, KV., Mohamed, NA., Arshad, H. 2018. *State-of-the-art in artificial neural network applications: A survey*. Heliyon.

Afram, A., Sharifi, FJ., Fung, AS., Raahemifar, K. 2017. *Artificial neural network based model predictive control and optimization of HVAC systems: A state of the art review and case study of a residential HVAC system*. Energy and Buildings.

Bressert, E. 2013. *Scipy and Numpy: An Overview for Developers*. O'Reilly

Deb, C., Eang, LS., Yang, J., Santamouris, M. 2016. *Forecasting diurnal cooling energy load for institutional buildings using Artificial Neural Networks*. Energy and Buildings.

Geron, A. 2017. *Hands-On Machine Learning with Scikit-Learn & TensorFlow: concepts, tools, and techniques to build intelligent systems*. O'Reilly

Hargan, MR. 2002. *ASHRAE guideline 14-2002 measurement of energy and demand savings*, American Society of Heating Refrigerating and Air-Conditioning Engineers Inc.

Kang, IS., Lee, KH., Lee, JH., Moon, JW. 2018. *Artificial neural network-based control of a variable refrigerant flow system in the cooling season*. Energies.

- Kyurkchiev, N., Markov, S. 2015. *Sigmoid functions: some approximation, and modelling aspects*. Some Moduli in Programming Environment Mathematica.
- Lutz, M., Ascher, D. 2013. *Learning Python, 5th Edition*. O'Reilly
The official website of Python. *Applications for Python*.
<https://www.python.org/about/apps/> (accessed January 22, 2019).
- Mba, L., Meukam, P., Kemajou, A., 2016. *Application of artificial neural network for predicting hourly indoor air temperature and relative humidity in modern building in humid region*. Energy and Buildings.
- Moon, JW. 2015. *Introduction to Artificial Intelligence Theory*. Korea Institute of Architectural Sustainable Environment and Building Systems.
- Muller, AC., Guido, S. 2017. *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly
- Rashid, T. 2016. *Make your own neural network*. Create Space Independent Publishing Platform.
- Reynolds, J., Rezgui, Y., Kwan, A., Piriou, S. 2018. *A zone-level, building energy optimization combining an artificial neural network, a genetic algorithm, and model predictive control*. Energy.
- Seo, BM., Lee, KH. 2016. *Detailed analysis on part load ratio characteristics and cooling energy saving of chiller staging in an office building*. Energy and Buildings.
- Seo, BM. 2017. *Comparison of cooling energy performance between conventional AHU system and DX AHU-water source VRF heat pump system in an office building*. Hanbat National University. South Korea. Master's thesis
- The U.S. Department of Energy. 2018. *EnergyPlus version 8.9 documentation 2018 input output reference*. The U.S. Department of Energy.