

## Article

# Comparative Performance of Machine Learning Algorithms in the Prediction of Indoor Daylight Illuminances

Jack Ngarambe <sup>1</sup>, Amina Irakoze <sup>2</sup>, Geun Young Yun <sup>1</sup>  and Gon Kim <sup>1,\*</sup>

<sup>1</sup> Department of Architectural Engineering, Kyung Hee University, Yongin 17104, Korea; ngarajack@khu.ac.kr (J.N.); gyyun@khu.ac.kr (G.Y.Y.)

<sup>2</sup> Department of Architectural Engineering, University of Ulsan, Ulsan 44610, Korea; aminah079@gmail.com

\* Correspondence: gonkim@khu.ac.kr

Received: 2 May 2020; Accepted: 29 May 2020; Published: 1 June 2020



**Abstract:** The performance of machine learning (ML) algorithms depends on the nature of the problem at hand. ML-based modeling, therefore, should employ suitable algorithms where optimum results are desired. The purpose of the current study was to explore the potential applications of ML algorithms in modeling daylight in indoor spaces and ultimately identify the optimum algorithm. We thus developed and compared the performance of four common ML algorithms: generalized linear models, deep neural networks, random forest, and gradient boosting models in predicting the distribution of indoor daylight illuminances. We found that deep neural networks, which showed a determination of coefficient ( $R^2$ ) of 0.99, outperformed the other algorithms. Additionally, we explored the use of long short-term memory to forecast the distribution of daylight at a particular future time. Our results show that long short-term memory is accurate and reliable ( $R^2 = 0.92$ ). Our findings provide a basis for discussions on ML algorithms' use in modeling daylight in indoor spaces, which may ultimately result in efficient tools for estimating daylight performance in the primary stages of building design and daylight control schemes for energy efficiency.

**Keywords:** machine learning; daylighting performance; daylighting control; deep learning; decision trees; daylight forecasting; predictive modeling; time-series

## 1. Introduction

Daylighting is an essential part of modern architecture. It allows for a reduction in the use of artificial lighting and thus, indirectly, contributes to reduced anthropogenic carbon dioxide emissions [1]. In addition, daylighting has been reported to enhance productivity in workplaces [2] and to boost overall mood in indoor spaces [3]. Moreover, we now understand the potential associations between daylight and human circadian rhythms, as well as the importance of properly maintained body clocks on the general well-being of building occupants [4]. As such, it is important that built environments are designed to provide sufficient levels of daylight for purposes of energy saving, occupant well-being, and visual comfort.

Traditionally, in the design stage of a building project, there are several methods that can be used to estimate the daylighting levels that are likely to be present in a space. These methods can be broadly categorized into three groups: physical modeling, computer simulations, and mathematical formulae for analytical calculations [5].

Physical modeling involves creating a replica of the space intended to be built, usually on scales ranging from 1:8 to 1:32 [6], and using it to estimate the potential levels of daylight in the real space intended to be built. There are a variety of materials (e.g., with differing reflectance) used in developing

the physical models to ensure that real-life building scenarios are properly replicated. While this approach is effective, it is also time-consuming and can often be expensive depending on the chosen scales and the need to replicate real-life scenarios, including shading and expected outdoor obstructions, in fine detail.

The use of computer simulations to estimate daylight levels during the building design stage has been applied extensively over the last few years. There is a variety of commercially available daylight simulation software on the market today [7]. These tools may vary in complexity, modeling capability, rendering, accuracy, etc., but the fundamental concepts on which they are based are primarily the same (i.e., ray tracing, radiosity, photon mapping) [8]. While simulation tools offer flexibility and speed in the evaluation of building daylighting performance [7], they often have steep learning curves, which may be disadvantageous to building designers with busy schedules. Moreover, the accuracy of the results obtained from computer software is largely dependent on the skill set of the user and is thus prone to errors, mainly due to lack of experience by the modeler.

There are also conventional, simplified, manual calculation formulae that are still commonly used to evaluate daylighting performance in the building industry. A good example is the split flux daylight factor (DF) method, which considers the ratio of outdoor horizontal illuminance to indoor horizontal illuminance on the work plane. DF is still the basis of many building standards and daylighting design guidelines [9] (e.g., BS206-2 and BS 209-2011). However, while this method is simple and easily applied, it does not consider various important factors that affect the availability of daylight within a space (e.g., climate, shading) [10]. There are other improved daylight metrics available, such as the useful daylight illuminance (UDI) and daylight autonomy (DA) [10]. However, the feasibility of applying manual calculations for long-term daylight performance evaluations is rather minimal. In addition, such analytical methods are often unable to provide detailed assessments of the available illuminance levels (e.g., how well the illuminance is distributed within the space).

To overcome some of the challenges experienced in the evaluation of daylighting performance using the three traditional techniques discussed above, recent studies have explored the potential use of machine learning (ML) techniques to assess daylighting performance in buildings. Machine learning techniques make use of existing data to learn patterns and relationships between the response and causal variables, and upon learning such patterns, they can be used in the estimation or forecasting of the response variable at a particular time in the future or under particular conditions [11]. However, despite the potential usefulness of machine learning in daylighting studies, the application of machine learning in daylight-related studies is rather new and the performances of only a few algorithms have previously been explored [9,12–15]. In this study, we demonstrated the performance of five machine learning algorithms in the prediction and forecasting of the distribution of indoor daylight illuminance. We considered both deep-learning-based and decision tree-based algorithms and discuss the optimum algorithm. Our study contributes to the small but increasing number of research studies attempting to employ artificial intelligence (AI) concepts in daylight design evaluations. For instance, by demonstrating the potential of machine learning algorithms to accurately learn the interactions between sunlight and buildings, we provide the basis for using simple data-driven models to replace complex, time-consuming simulation tools in daylight performance analysis. Furthermore, we demonstrate the development of machine learning models that can be used in the dynamic predictive control of indoor daylight illuminance to reduce the unnecessary usage of artificial lighting in buildings and subsequently contribute to reduced building energy consumption. These models can also be employed in the automatic control of solar shading devices to enhance visual comfort in indoor spaces.

## 2. Related Works and the Considered Machine Learning Algorithms

As briefly discussed in the previous section, only a small number of studies have attempted to employ ML algorithms in the estimation of indoor daylight performance. For example, Kazanasmaz et al. [12] developed an artificial neural network (ANN) model to predict daylight illuminance levels of an office

building based on field data representing climatic and building design factors. The developed model was reported to predict indoor illuminance levels with an accuracy of 98%. Similarly, Ahmed et al. [13] compared the performance of random forest (RF) and ANN algorithms in the prediction of daylight illuminance in a classroom. They reported better predictions by RF models ( $R^2 = 0.98$ ) than ANN models ( $R^2 = 0.97$ ). Furthermore, Lorenz et al. [9] demonstrated the potential of ANN-based models to predict DA with changing window designs while considering the effects of shading from external obstructions. Their models were able to predict DA with an average error of 3 DA values between the predicted and actual DA. Further, Zhou and Liu [14] compared the performance of ANN and support vector machine (SVM) algorithms, with principal component analysis (PCA) as a feature selection tool, in the prediction of UDI. They reported better performance with ANN than SVM. A couple of previous studies have also employed machine-learning-based time series modeling to forecast indoor illuminance at a particular future time. Kurian et al. [16] demonstrated the potential usage of adaptive neural fuzzy inference systems (ANFISs) to forecast indoor daylight illuminance. Recently, Waheeb et al. [15] compared the performance of four time-series modeling techniques (seasonal naïve, seasonal ARIMA, STL, and TBATS) in the forecasting of indoor daylight illuminances provided by a light pipe daylight delivery system. They reported that seasonal and trend decomposition using loss (STL) techniques outperformed the other techniques in forecasting indoor daylight illuminances. In the following subsections, we explain briefly the mechanisms of each of the five ML algorithms explored in the current study.

### 2.1. Generalized Linear Models (GLMs)

GLMs are conventional statistical techniques that interpret the possible linear relationships between the response variable and one or a set of explanatory variables [17]. A typical GLM consists of a set of input variables ( $x_1, x_2, x_3 \dots x_n$ ) and an output variable  $Y$ . The input variables are assigned weights that represent the overall effect of each input variable on the output variable ( $\beta$ ). GLMs thus typically take the form shown in Equation (1) below:

$$Y = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \dots \beta_n x_n \quad (1)$$

where  $Y$  represents the output variable,  $x_1, x_2, x_3 \dots x_n$  represent the input variables,  $\beta_1, \beta_2, \beta_3 \dots \beta_n$  represent the regression weights, and  $\epsilon$  represents the error between the predicted and actual value.

### 2.2. Random Forest (RF)

RFs are decision tree algorithms based on ensemble methods techniques. RFs were developed to overcome the shortcomings of conventional decision trees methods (e.g., lack of robustness) [18]. They generally employ two statistical approaches: bootstrapping and bagging. Bagging is a technique used to split a given training dataset into smaller data sets, each of which can be used to learn patterns among variables in supervised learning. For example, assuming a training dataset  $X$  of size  $n$ , the bagging technique randomly generates  $m$  new smaller datasets,  $X^i$ , from the original dataset, each with size  $n_i$ . The generation of these sample sub-datasets can either be done “with replacement” or “without replacement”. Bootstrapping is a sampling technique used in the bagging process and involves sampling with replacement [19]. The general goal of the bagging technique in RF algorithms is thus to produce subsets of random samples from the original dataset. The analysis is then conducted on each subset and the outcomes are averaged to reduce errors and variance in the response variable [20]. RF can be employed in both regression and classification tasks. In our case, we used RF to predict daylight illuminances based on building design, weather, and time factors. As such, assuming a vector,  $x_1, x_2, x_3, \dots, x_n$ , that consists of all our input variables (i.e., building design, weather, time factors) and the corresponding labels (i.e., indoor daylight illuminances), RF randomly split our vector into  $m$  subsets, which are often referred to as decision trees. Since our goal was to learn relationships in our input vector that correspond to certain labels (i.e., expected output), the output was predicted for

each tree. The final output was the average value of all the outputs from all the random trees created. The general process of training a randomly generated forest is discussed by Svetnik et al. [21] and summarized as follows. First, a bootstrap sample is extracted from the training dataset. Then, a tree is developed for each bootstrap sample and the best split is selected from the subset of the bootstrap sample. The first and second steps are then repeated until a certain number of trees are grown. Finally, the average values from each tree are aggregated to obtain a final output.

### 2.3. Gradient Boosting Model (GBM)

GBMs are another ensemble method based on decision trees. GBMs create several additive models and present the optimum model that minimizes the loss function. They do this by first creating several simple models, referred to as weak learners, and then combining them in a sequential additive manner to obtain models with better performance, referred to as strong learners. The training process of GBMs is similar to that of RF models, with the primary difference being that the RF trains each decision tree independently, using random parameters, and combines the output from each independent tree, whereas GBMs train decision trees one at a time, with the new trees attempting to minimize errors made by previous trees, a technique known as boosting [22].

### 2.4. Deep Neural Networks (DNNs)

ANNs and DNNs are ML algorithms that are designed to replicate the way humans learn [23]. ANNs are capable of learning highly complex patterns between variables and have recently been applied extensively in computational modeling [24]. The primary difference between an ANN and a DNN is the number of hidden layers in the developed model. A model developed using a few hidden layers (typically less than three) is considered a shallow ANN model, whereas a model with many hidden layers is considered a deep ANN or just a DNN. A typical ANN model is made up of neurons and three layers: an input layer, a hidden layer, and an output layer. Neurons are the primary units of the model, and they are interconnected throughout the layers. When a set of features is fed into the model, the weighted combination of the input signals is aggregated and an input signal is transmitted by the interconnected neurons to the output layer. Developed models may have any number of neurons in the hidden layer, depending on the complexity of the problem being solved. However, the numbers of neurons in the input and output layers are typically equal to the number of predictor variables and response variables, respectively. Additionally, there are many ANN architectures (e.g., feed-forward, Hopfield, Elman, radial basis networks) [25], and developed models may have different transformation functions, which are functions responsible for learning non-linear patterns within the data. In the present study, we used a feed-forward network architecture trained with a back-propagation algorithm. A detailed account of how to train a neural network using a back-propagation algorithm is given by Ermis et al. [26].

### 2.5. Long Short-Term Memory (LSTM)

LSTMs are a special type of recurrent neural networks (RNNs). RNNs are neural networks with loops such that, after proper training, information learned from previous loops (past values) is used to predict desired values at a future time. For example, in a typical RNN model  $f(x_1 \dots x_t)$ ,  $x_1 \dots x_t$  indicate historical measurements at the particular time  $1 \dots t$  and the final output relates to a particular time in the future, e.g.,  $t + 1$ . A typical RNN model, therefore, is different from the conventional feed-forward network in that the loops in RNNs allow the model to operate a dynamically changing contextual window, making it possible for the model to feed the network activations from a previous time step as inputs into the network; this, in turn, allows the previously learned patterns to influence the prediction at a particular time in the future, whereas a conventional ANN operates on a fixed-size sliding window [27]. Similarly, LSTM models have been reported to outperform standard RNNs because they contain special units called memory blocks that allow LSTM models to store much more past information than standard RNNs. LSTMs also consist of multiplicative units called gates that

control the flow of information from one memory block to another and into the rest of the network. A single memory block, therefore, contains an input gate and an output gate. Additionally, LSTMs contain other special units called peephole connections that connect internal cells to the gates within the same cell. This helps the model learn the exact times of the outputs [28]. LSTMs have been successfully applied in many fields, especially in speech recognition [29]. Sak et al. [27] give a detailed discussion on typical LSTM computation.

### 3. Methodology

#### 3.1. Model Parameters

We considered three building factors, seven climatic factors, and time as input variables. For building factors, we considered the window to wall ratio (WWR), wall reflectance (WR), and distance from the window (DFW). As climatic factors, we considered global horizontal irradiance (GHI), direct normal irradiance (DNI), diffuse horizontal irradiance (DHI) global horizontal illuminance (GHIL), direct normal illuminance (DNIL), relative humidity (RH), and sky cover (SC). These same variables have been employed in a previous study that deals with ML-based predictive modeling of indoor illuminances [12]. Table 1 below shows the variables used as features to develop the GLM, RF model, GBM, and DNN model. The LSTM models were developed using only the time-dependent variables shown in Table 1, i.e., all the features except for the building design factors (WWR, WR, DFW).

**Table 1.** Input variables for model development.

Parameter	Min	Max
WWR [%]	20	60
WR [%]	20	60
DFW [m]	0	9
GHI [w/m <sup>2</sup> ]	0	992
DNI [w/m <sup>2</sup> ]	0	970
DHI [w/m <sup>2</sup> ]	0	970
GHIL [Lux]	0	121,265
DNIL [Lux]	0	97,695
RH [%]	10	98
SC [%]	0	100
Month	1	12
Day	1	365
Time [hours]	6	20
Illuminance [Lux]	0	27,580

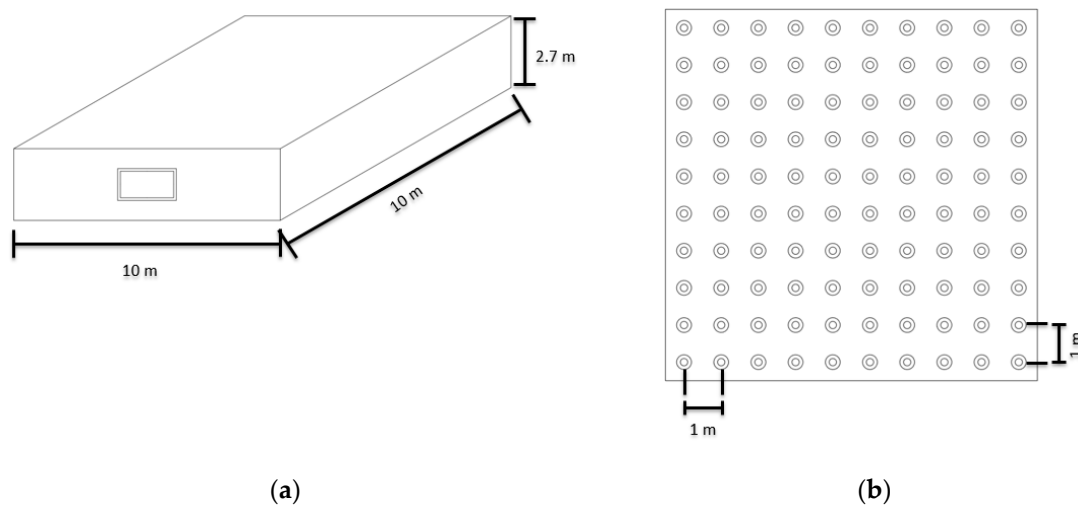
WWR—window to wall ratio, WR—wall reflectance, DFW—distance from the window, GHI—global horizontal irradiance, DNI—direct normal irradiance, DHI—diffuse horizontal irradiance, GHIL—global horizontal illuminance, DNIL—direct normal illuminance, RH—relative humidity, SC—sky cover.

#### 3.2. Data Acquisition (Simulation Design)

The data used in the training and validation of the developed models were acquired through a daylight computer simulation tool. We designed a 10 m × 10 m generic model with a height of 2.7 m (see Figure 1) based on Lorenz et al. [9]. The daylight simulations were conducted using the Radiance engine in EnergyPlus [30]. Radiance has been employed extensively in daylight simulation tools and has been validated through extensive experimental studies [31–33]. We developed several generic models to match all our intended scenarios. We thus designed nine generic models, a model for each combination of WWR and WR. For example, a model with a 20% WWR was simulated under three different values of WR (20%, 40%, and 60%), and this was repeated for the remaining two WWR categories (40% and 60%). The simulations were conducted for indoor illuminance at hourly intervals at a work plane height of 0.8 m. In addition, there were 10 sensors spaced in 1 m intervals from the



window to the rear of the room and from the west-facing wall to the east-facing wall; this resulted in a total of 100 sensor points.



**Figure 1.** Generic diagram of the designed space: (a) dimensions of the space and (b) top view of the daylight sensors.

### 3.3. Model Development and Optimization Techniques

Most supervised machine learning algorithms require a minimum of two sets of data. One set of data is used to train the model (i.e., learn the patterns between the explanatory and response variables) and the second set is used to validate the developed model (i.e., assess how the model would perform on a set of new data). In our case, for all four models (i.e., GLM, DNN, RF, and GBM), 80% of the data were used to train the model and 20% were used in model validation.

Certain other model parameters are most likely to influence how the developed model performs. For example, algorithms belonging to the decision tree family are largely influenced by the number of trees within the model; this is typically referred to as “tree depth.” The number of trees required for a good model performance is largely dependent on the complexity of the available data. In the present study, to determine the optimum model performance, we assessed the relationship between the performance of the developed RF model and GBM and the number of trees. We thus developed initial models with 2 trees for the RF model and 10 trees for the GBM and gradually increased the number of trees while observing the changes in the performance of the model. The optimum model was then developed with the number of trees that minimized the error between the predicted and simulated values.

As with the number of trees in decision tree-based algorithms, the number of hidden layers in a DNN model relies on the complexity of the data. As such, to determine the number of hidden layers that would result in the best model performance, the models were initially developed with two layers and we then increased the number of layers, one by one, until we observed no further reductions in the produced errors. For the number of neurons per layer, there is no agreed-upon methodology for determining the suitable number of neurons per hidden layer. However, some researchers have suggested simple arithmetic formulas to determine the optimum number of neurons in the hidden layers. For example, Heaton [34] suggested that the optimum number of neurons can be obtained by  $\frac{\text{No. of inputs} + \text{No. of outputs}}{2}$ . Similarly, Hecht-Nielsen [35] stated that the optimum number of neurons in hidden layers can be determined by  $2n + 1$ , where  $n$  is the total number of input variables. In our case, we followed the latter case to determine the initial number of neurons in the hidden layers and gradually increased them until there were no further reductions in the produced errors. Following the above criteria, therefore, our DNN model was made of five hidden layers, each with 29 neurons, as a higher number of hidden layers of neurons did not result in any improvements in the performance of

the model. For the activation function, we employed rectified linear units (ReLU). A typical ReLU function has the format  $f(x) = \max(0, x)$ , where  $x$  represents the input into the neuron. The ReLU function has been reported to enable better performance of DNN models than the Tanh or Maxout functions [36].

For the developed LSTM models, factors related to building design (WWR, WR, DFW) were not considered as input variables, as they are not time-dependent and hence using the term forecasting model rather than the predictive models discussed above. The models were thus trained and validated using weather data, time, and indoor illuminance as input variables; ReLU as an activation function; and Adam as an optimizer. Additionally, the number of hidden layers was gradually increased from one layer until there were no further significant changes in the root mean square error (RMSE) and the coefficient of determination ( $R^2$ ). Furthermore, for the forecasting of indoor illuminances, we used historical data recorded by the models in the last 1 h (i.e.,  $t-1$ ) to forecast the indoor illuminance at the present time (e.g.,  $t$ ). Table 2 shows the optimization parameters used in developing the models and Table 3 shows the descriptive statistics of the training and validation datasets.

**Table 2.** Optimization parameters for the developed models.

	RF	GBM	DNN	LSTM
Number of hidden layers	n/a	n/a	5	3
Number of neurons in the hidden layer	n/a	n/a	29	25
Number of trees	10	12	n/a	n/a
Epochs	n/a	n/a	1000	1000
Learning rate	n/a	0.1	0.01	0.01
Loss function	MSE	LS	RMSE	RMSE
Optimizer	n/a	n/a	Adam	Adam
Maximum features	Auto	Auto	n/a	n/a
Activation function	n/a	n/a	ReLU	ReLU
Sample rate	0.8	0.8	n/a	n/a

RF—random forest, GBM—gradient boosting model, DNN—deep neural network, LSTM—long short-term memory, MSE—mean square error, LS—least squares regression, RMSE—root mean square error, ReLU—rectified linear units.

**Table 3.** Descriptive statistics of the datasets.

Dataset	N	Range	Minimum	Maximum	Mean	Std. Deviation
Training	337,256	27,492.60	0.00	27,492.60	794.88	2233.90
Validation	83,989	27,580.40	0.00	27,580.40	784.84	2214.13

### 3.4. Evaluation of Model Performance

The developed models were evaluated using the root mean square error (RMSE) and the mean absolute error (MAE). RMSE and MAE are scale-dependent metrics used to calculate the overall difference between the measured value and the value predicted by a given developed model [37,38]. We also used the coefficient of determination ( $R^2$ ) to quantify the total variance of indoor illuminance explained by the developed models. Equations (2)–(4) show mathematical illustrations of the MAE, RMSE, and  $R^2$ , respectively.

$$MAE = \frac{1}{n} \sum_{i=1}^n |x_i - \hat{x}_i| \quad (2)$$

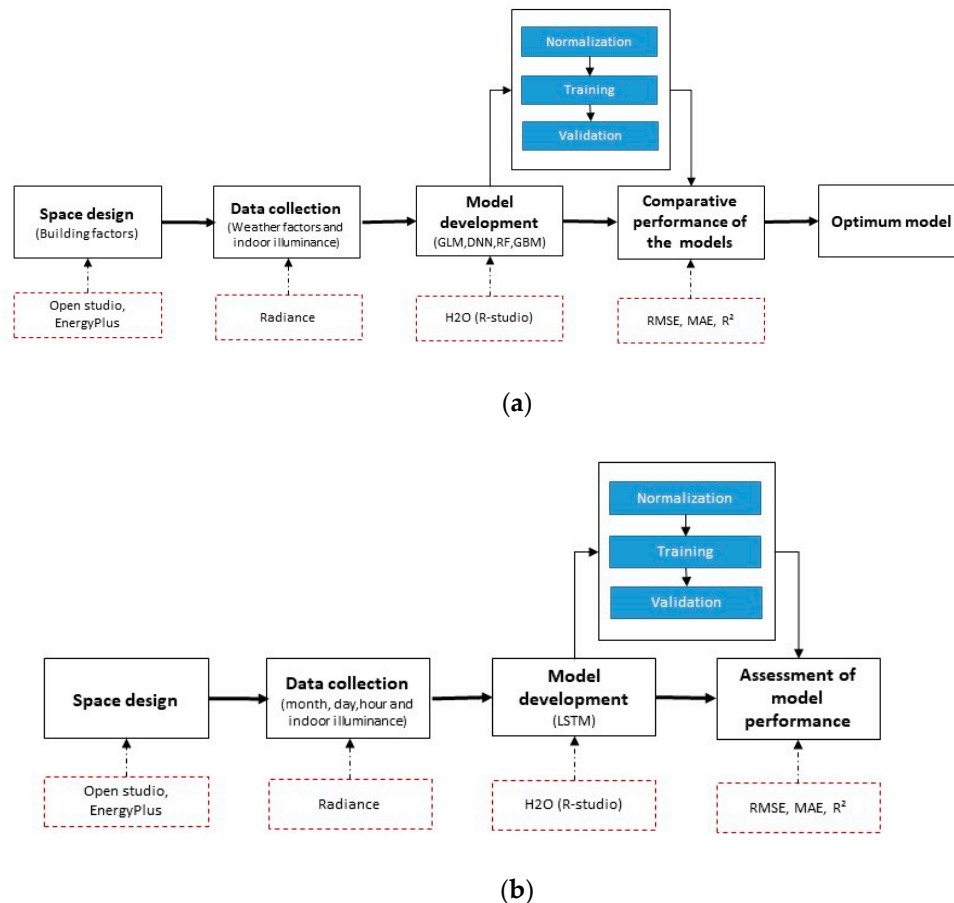
$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2} \quad (3)$$

$$R^2 = \left( \frac{\sum_{i=1}^n \hat{x}_i x_i - \sum_{i=1}^n \hat{x}_i \sum_{i=1}^n x_i / n}{(\sum_{i=1}^n \hat{x}_i^2 - (\sum_{i=1}^n \hat{x}_i)^2 / n)(\sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2 / n)} \right)^2 \quad (4)$$

where  $x_i$  is the simulated data at the time  $i$ , and  $\hat{x}_i$  is the corresponding predicted value at ( $i = 1, 2, \dots, n$ ).

### 3.5. Study Process

The general process of the study involved first designing our generic model space in OpenStudio and using the Radiance engine in EnergyPlus to analyze daylight behavior (i.e., indoor daylight illuminance) and subsequently gather weather data, building design data, and time factors. The collected data were used to develop the machine learning models presented in the current study. Figure 2 shows the study design adopted in the current manuscript.



**Figure 2.** Study process for (a) other machine learning (ML) models and (b) LSTM model. GLM—generalized linear model, MAE—mean absolute error, R<sup>2</sup>—coefficient of determination.

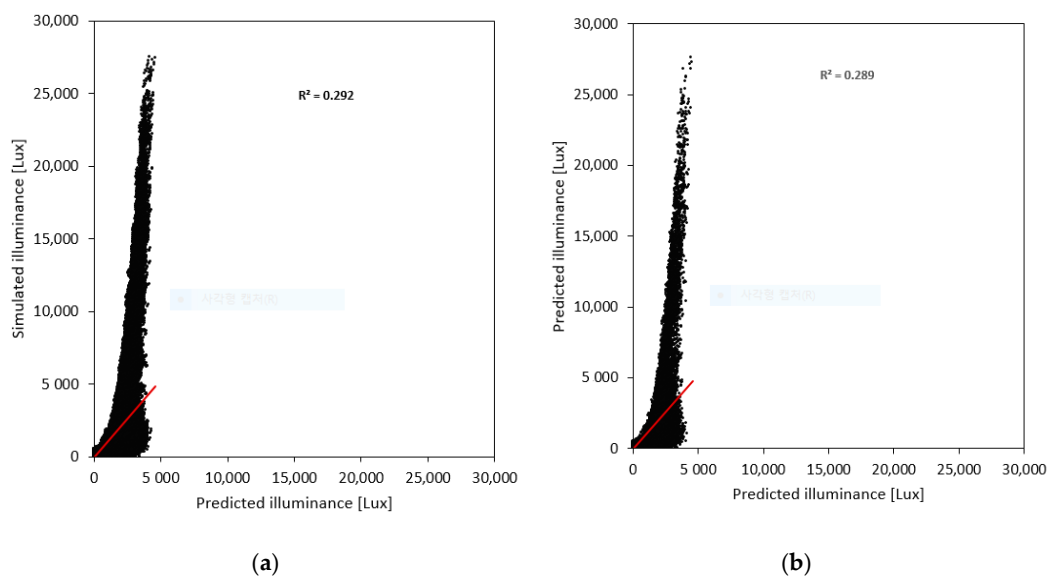
## 4. Results

### 4.1. Performance of the Developed Models

#### 4.1.1. GLM

GLMs were able to explain 29% of the variations in indoor illuminance levels for the training dataset. A similar performance was obtained for the validation dataset ( $R^2 = 28.9\%$ ), indicating no signs of overfitting or underfitting. Figure 3 shows the performance of the developed GLM.

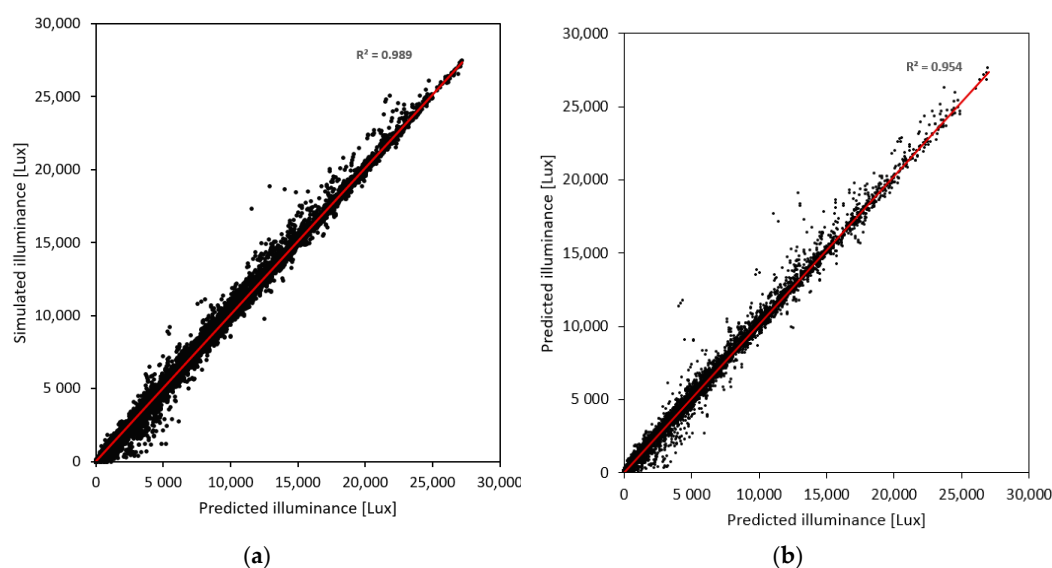




**Figure 3.** Simulated illuminances vs. GLM-predicted illuminances for (a) training dataset and (b) validation dataset.

#### 4.1.2. RF

The RF models indicated good model performance on both the training dataset ( $R^2 = 0.998$ ) and the validation dataset ( $R^2 = 0.995$ ). The small difference between the results of the training and validation datasets indicates no signs of overfitting or underfitting of the model. In addition, our results showed that the number of trees has a significant impact on the performance of the RF models. For example, a model with two trees indicated an  $R^2$  of 0.978, RMSE value of 255.5, and MAE value of 56.714. However, doubling the number of trees to four trees increased the  $R^2$  value to 0.984, and decreased the RMSE value to 184.811 and the MAE value to 47.802. The increase in the number of trees increased the model performance and ultimately reduced the error in the model. The optimum performance of the model (i.e., maximum  $R^2$  and minimum RMSE and MAE) was achieved when the number of trees was 10. Figure 4 shows the performance of the developed RF models and Table 4 shows the changes in RF model performance as a result of increasing the number of trees.



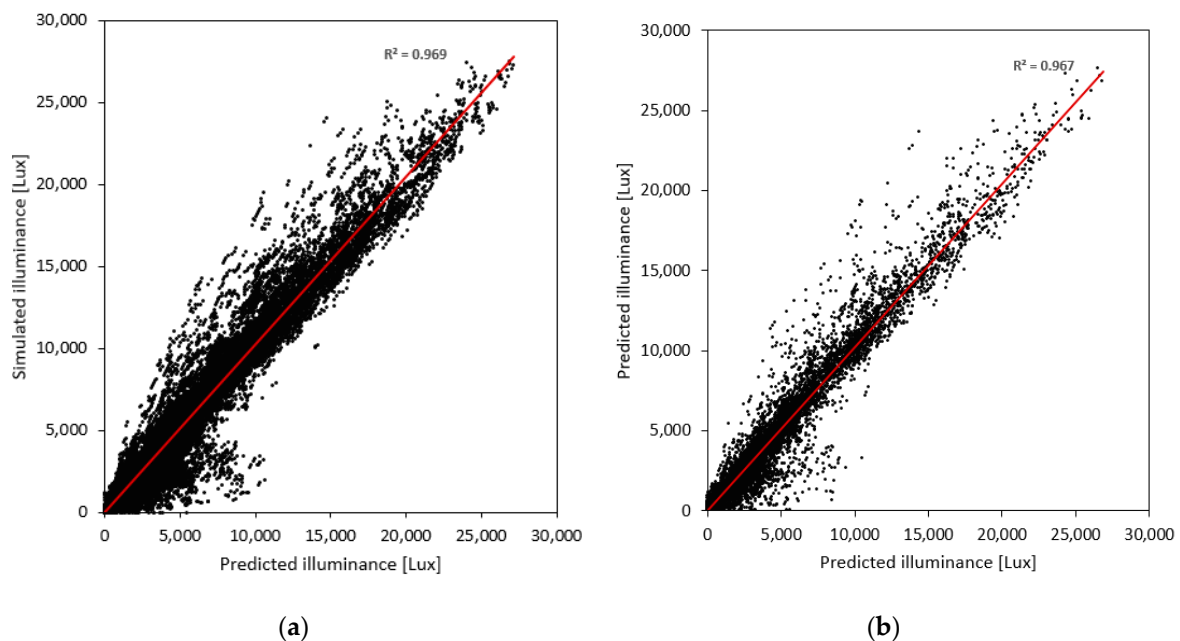
**Figure 4.** Simulated illuminance vs. RF-predicted illuminance for (a) training dataset and (b) validation dataset.

**Table 4.** Performance of RF models based on the number of trees.

Number of Trees	Training Dataset			Validation Dataset		
	RMSE	MAE	R <sup>2</sup>	RMSE	MAE	R <sup>2</sup>
2	328.328	64.333	0.978	255.553	56.714	0.986
4	277.693	59.137	0.984	184.811	47.802	0.983
6	262.218	56.401	0.986	171.599	44.934	0.983
8	262.218	53.383	0.988	161.833	42.268	0.984
10	228.256	52.387	0.989	155.564	41.713	0.954

#### 4.1.3. GBM

The developed GBMs also showed good predictive abilities. They indicated a high  $R^2$  value of 0.967 on both the training and validation datasets. However, compared to RF models, GBMs tended to achieve acceptable model performance at a relatively higher number of trees. For example, when 10 trees were used, the developed GBM had a low  $R^2$  of 0.139, a relatively high RMSE of 2072.213, and an MAE of 934.812. However, increasing the number of trees to 50 increased the  $R^2$  to 0.503 and decreased both the RMSE and MAE to 1573.895 and 682.162 respectively. Similarly, doubling the number of trees from 50 to 100 tended to increase the  $R^2$  value to 0.726 and lower the RMSE and MAE values to 1168.233 and 487.698, respectively. The optimum performance of the models was achieved with 1000 trees, which had an  $R^2$  value of 0.967, an RMSE value of 393.956, and an MAE value of 120.943. Figure 5 shows the performance of the developed GBMs, and Table 5 shows the changes in the performance of GBMs with changes in the number of trees.



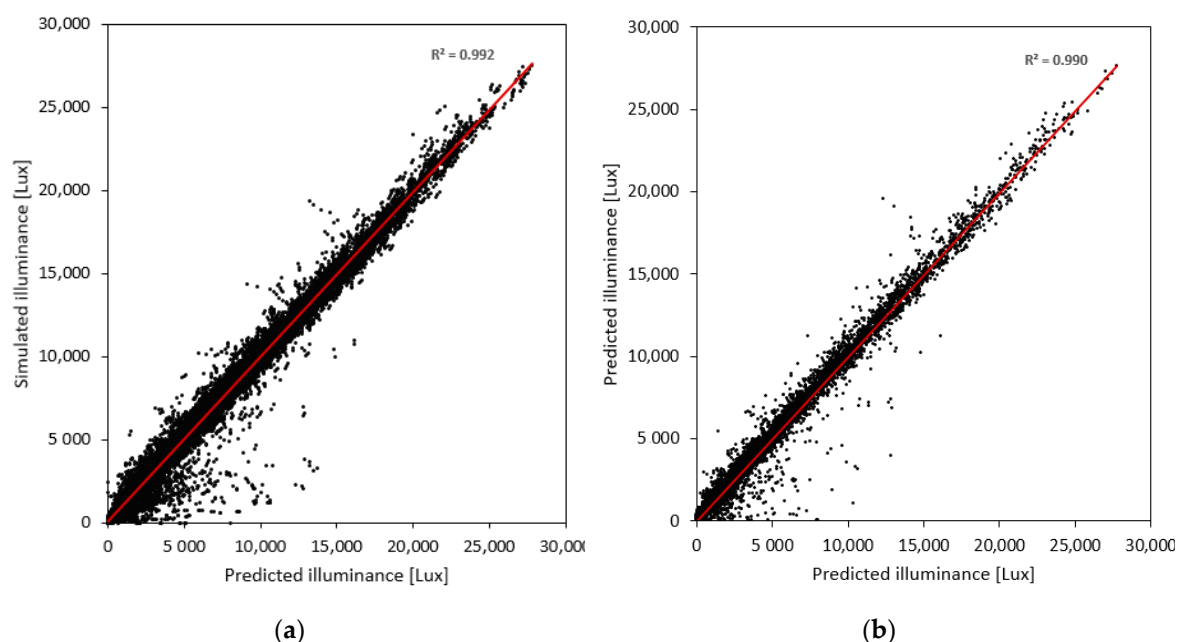
**Figure 5.** Simulated illuminance vs. GBM-predicted illuminance for (a) training dataset, and (b) validation dataset.

**Table 5.** Performance of GBM models based on the number of trees.

Number of Trees	Training Dataset			Validation Dataset		
	RMSE	MAE	R <sup>2</sup>	RMSE	MAE	R <sup>2</sup>
10	2072.213	934.812	0.139	2054.344	929.155	0.139
50	1573.895	682.162	0.503	1562.909	679.473	0.501
100	1168.233	487.698	0.503	1562.909	486.725	0.724
500	461.895	144.498	0.957	470.002	144.967	0.954
1000	393.956	120.943	0.969	403.989	122.190	0.967

#### 4.1.4. Deep Neural Network (DNN)

DNN models also indicated a good ability to predict indoor daylight illuminances. They demonstrated an R<sup>2</sup> value of 0.991 on the training dataset and 0.990 on the validation dataset. Similar to the GBM and RF model discussed above, the small difference between the training and validation results indicates that the model was properly fitted (i.e., no overfitting or underfitting). Additionally, we found that the number of hidden layers had a significant impact on the performance of the model. For example, with a single hidden layer, the model showed an R<sup>2</sup> value of 0.955, an RMSE value of 471.389, and an MAE value of 222.267. However, with two hidden layers, the RMSE value decreased by almost half to 241.282, the MAE value decreased to 86.860, and the R<sup>2</sup> increased to 0.988. The optimum model performance was achieved with five hidden layers, which showed an R<sup>2</sup> value of 0.992, and RMSE and MAE values of 199.274 and 69.136, respectively. Figure 6 shows the overall performance of the DNN model and Table 6 shows the changes in model performance with an increasing number of layers.



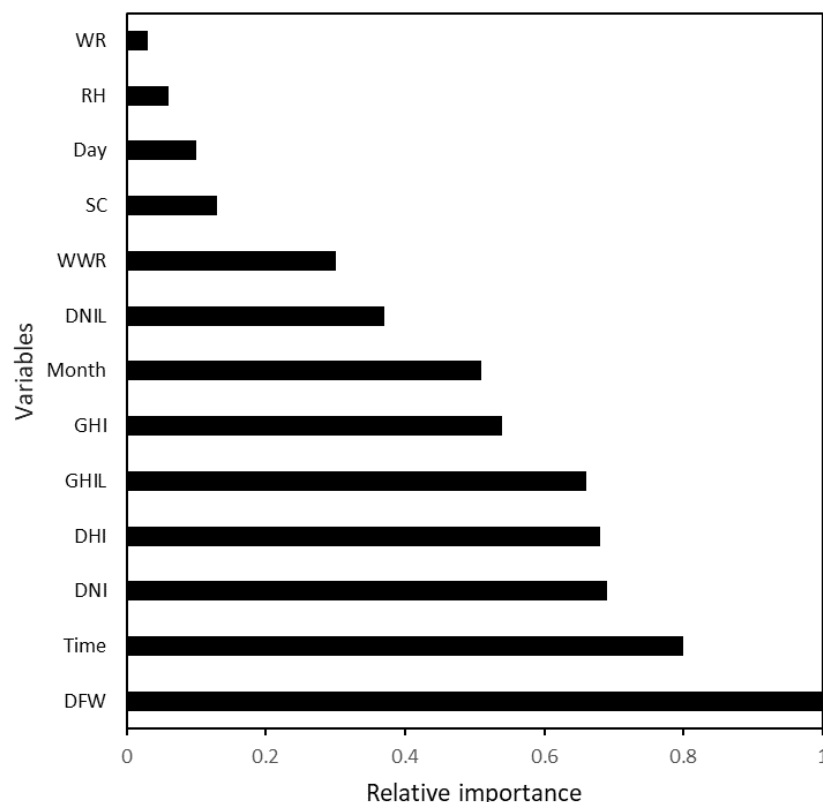
**Figure 6.** Simulated illuminances vs. DNN-predicted illuminances for the (a) training dataset and (b) validation dataset.

**Table 6.** Performance of DNN models based on the number of hidden layers.

Number of Trees	Training Dataset			Validation Dataset		
	RMSE	MAE	R <sup>2</sup>	RMSE	MAE	R <sup>2</sup>
1	471.389	222.267	0.955	477.517	224.106	0.953
2	241.282	86.860	0.988	252.232	88.079	0.987
3	207.614	79.638	0.991	228.320	81.580	0.989
4	195.301	67.558	0.992	211.443	69.166	0.990
5	199.274	69.136	0.991	222.791	71.893	0.990

#### 4.2. Relative Importance of the Factors Affecting Daylight Indoor Illuminance

Using deep learning methods described by Garson [39], we identified which input variables were most important in explaining the distribution of indoor daylight illuminance. This method determines the strength of the association between each input variable and the output variable by identifying all the weighted connections between the nodes that connect the input and output variables. This process was repeated for all explanatory variables and the results were scaled to obtain a single value for each variable that indicated the strength associated with each explanatory variable relative to the other explanatory variables. Our results showed that DFW, time of day (TIME) (time of the day), and DNR were the three most important factors explaining the distribution of indoor daylight illuminance. Our results also showed that, in general, indoor daylight illuminance was affected more by weather factors than by building design factors. Figure 7 shows the relative importance of the explanatory variables for indoor daylight illuminance.

**Figure 7.** The relative importance of model features. TIME—time of the day.

#### 4.3. Comparative Performance of the Developed Models

Using the deep learning-based relative importance of variables, discussed above, we divided our explanatory variables into two categories: important variables and all variables. Important variables

are the five variables that ranked highest on the relative importance scale (DFW, Time, DNI, DHI, GHIL). The “all variables” category includes all the explanatory variables used in the study. We then compared the performance of the four developed models, first using all variables and then using only the important variables. Using only a few important variables in predictive analytics, especially those dealing with ML, rather than several variables has the potential to reduce model dimensionality and thus reducing overfitting chances. It also serves to reduce computational complexity and thus improving model efficiency.

Our results showed that the models developed using all variables outperformed those developed using only the five most important variables for all algorithms. For example, while the  $R^2$  value was 0.992 for the DNN model developed using all variables, the  $R^2$  value for the DNN model developed using only the important variables was just 0.782. The same trend was observed across the other three algorithms (i.e., GLM, RF, and GBM). However, the  $R^2$  values for all four models developed using only the five most important variables were still quite high (above 0.65) for both the training and validation datasets.

Furthermore, our results indicated that DNN was the best-performing model among the four machine learning algorithms considered in this study. For example, while the  $R^2$  value of the DNN model trained on all variables was 0.992, the corresponding  $R^2$  values for the GLM, RF, and GBM were 0.290, 0.989, and 0.968 respectively. For the models trained with only the five most important variables, DNN still outperformed the rest of the models. For example, while the  $R^2$  value for the DNN model was 0.782, the  $R^2$  values for the GLM, RF, and GBM were 0.264, 0.695, and 0.768, respectively. Table 7 shows the comparative performance of the models developed using all variables and Table 8 shows the comparative performance of the models developed using only the five most important variables. Table 9 shows the time taken to train each model. However, it should be noted that the time taken for model training is primarily dependent on the capacity of the computers used and may thus vary from computer to computer. In our case, we used workstations with NVIDIA GeForce GTX650 graphics processing unit (GPU) and Intel® core™i5-7500 central processing unit (CPU) with a processor of 3.4 GHz and a memory of 8 GB.

**Table 7.** Comparative performance of the models developed using all variables.

	Training Dataset			Validation Dataset		
	RMSE	MAE	$R^2$	RMSE	MAE	$R^2$
GLM	1881.14	995.811	0.290	1868.031	992.051	0.288
RF	228.256	52.387	0.989	470.002	144.967	0.954
GBM	393.956	120.943	0.969	403.989	122.190	0.967
DNN	199.274	69.136	0.992	222.791	71.893	0.990

**Table 8.** Comparative performance of the models developed using only the five most important variables.

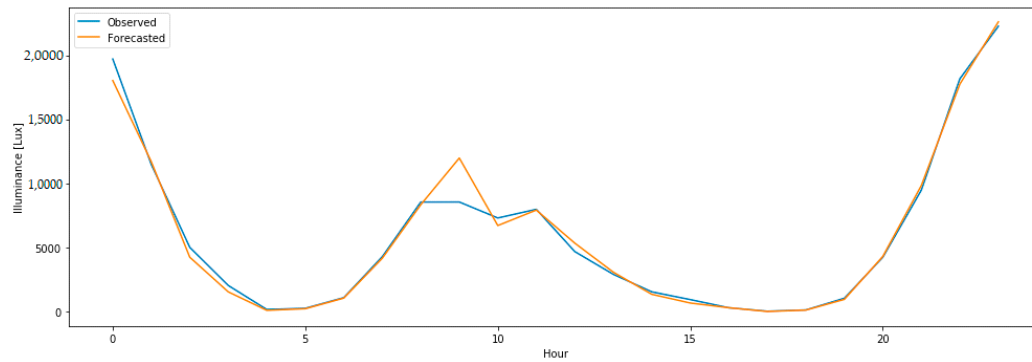
	Training Dataset			Validation Dataset		
	RMSE	MAE	$R^2$	RMSE	MAE	$R^2$
GLM	1915.851	987.344	0.264	1900.645	982.359	0.263
RF	1237.218	390.242	0.695	1061.408	340.497	0.774
GBM	1074.865	347.405	0.768	1081.591	346.671	0.761
DNN	1042.133	346.390	0.782	1043.933	344.732	0.777

**Table 9.** Time taken for model training (seconds).

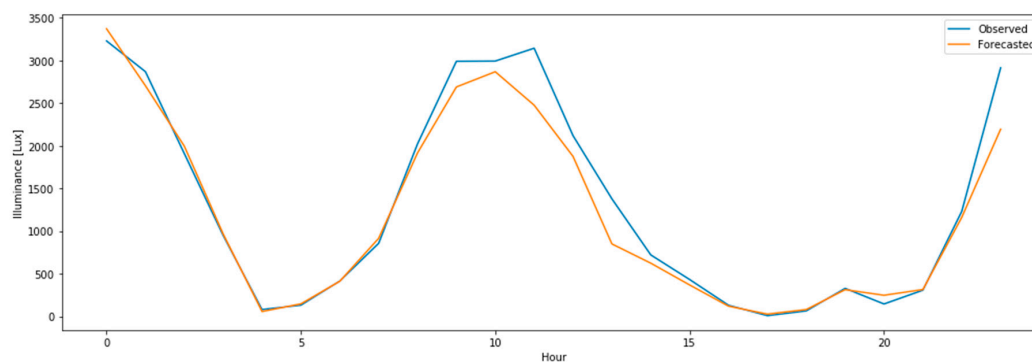
Model	All Variables	Important Variables
GLM	0.64	0.15
RF	150	113
GBM	129	107
DNN	204	143

#### 4.4. Forecasting of Daylight Indoor Illuminance Using LSTM

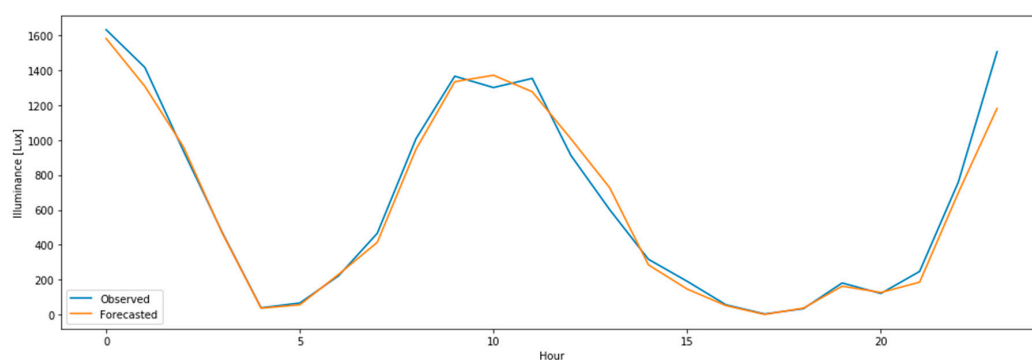
The LSTM models were developed for each sensor line (i.e., at 1 m intervals from the window); since our model space measured 10 m by 10 m (see Figure 1), we ended up with 10 LSTM models. The features used to develop the LSTM models are only time-dependent factors and are discussed in Section 3.1. Our results (See Figure 8) showed that LSTM algorithms could accurately mimic distributions in indoor daylight illuminance.



(a)



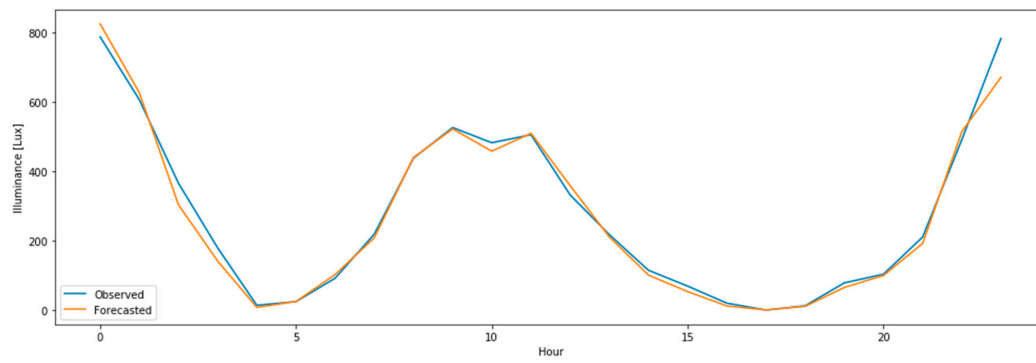
(b)



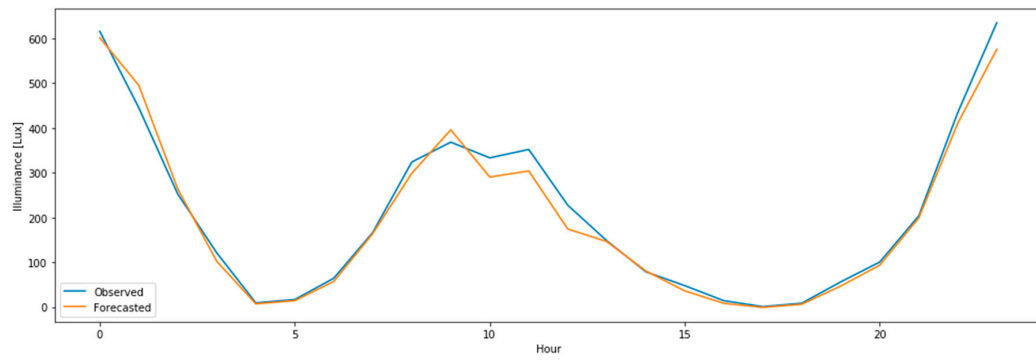
(c)

Figure 8. Cont.

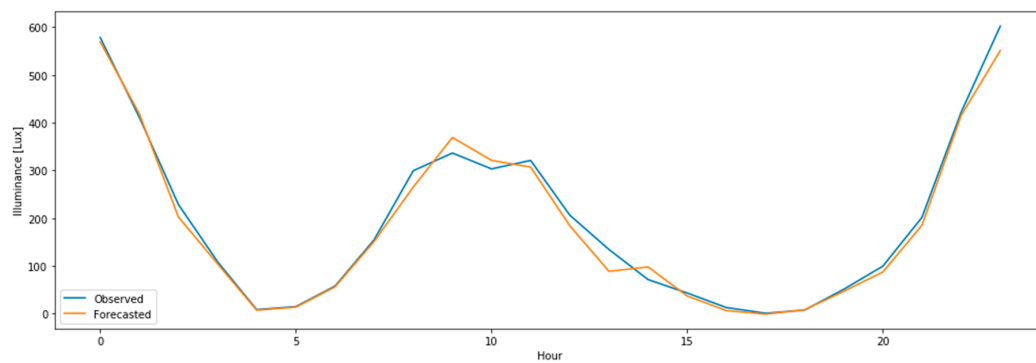




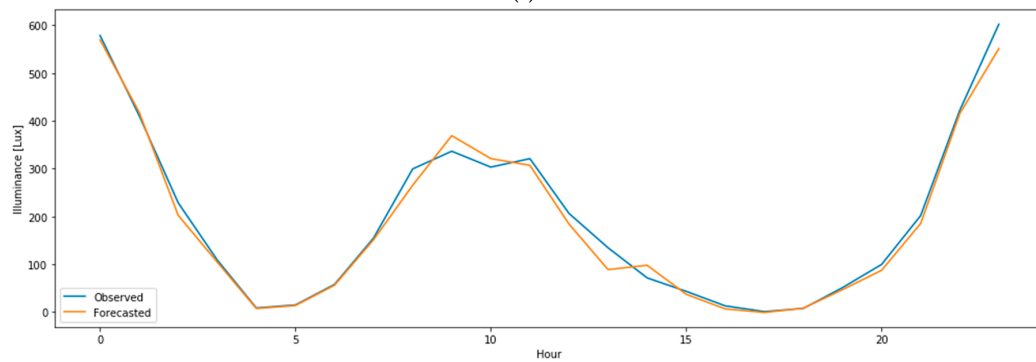
(d)



(e)

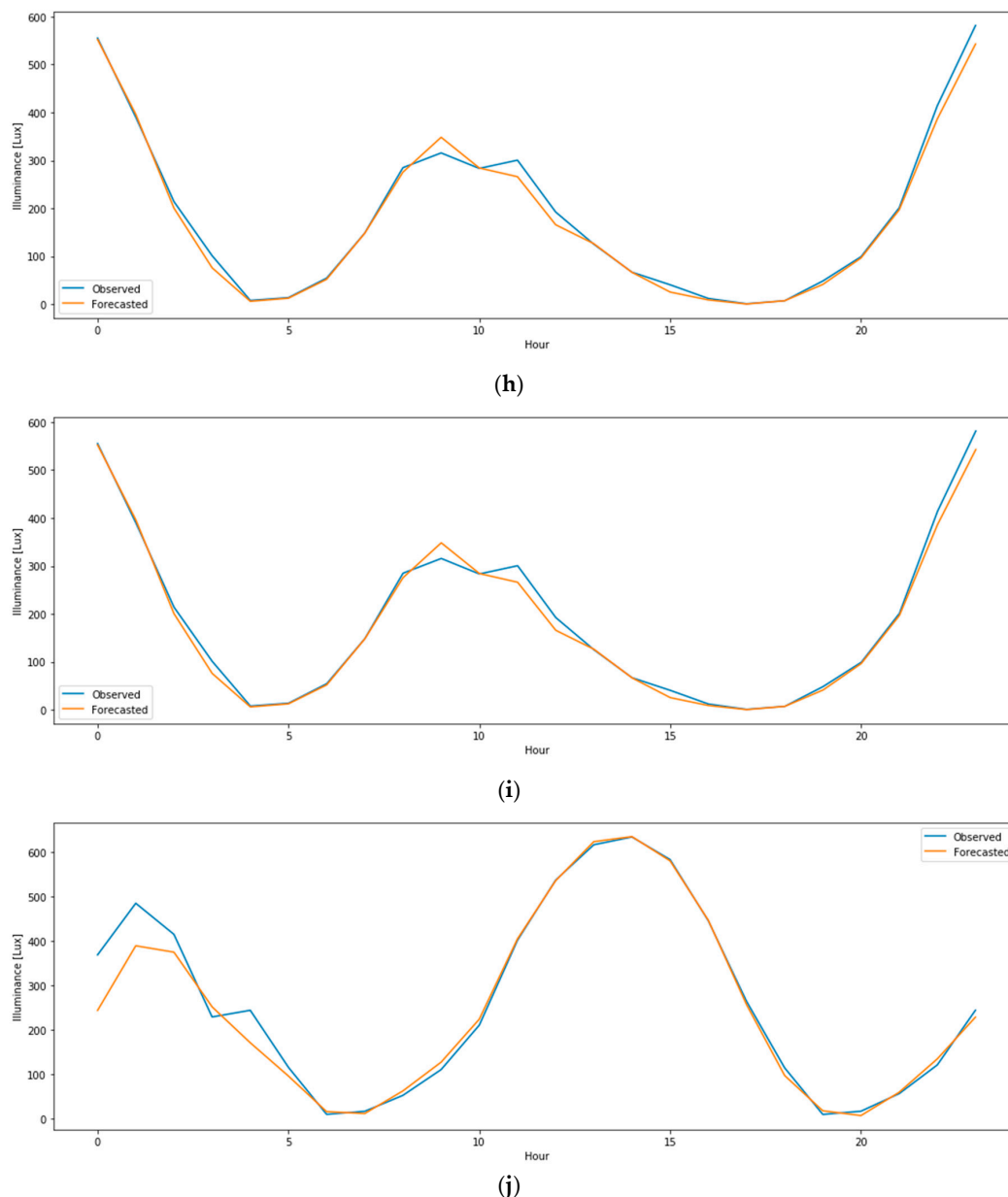


(f)



(g)

Figure 8. Cont.



**Figure 8.** LSTM model performance at (a) 0 m, (b) 1 m, (c) 2 m, (d) 3 m, (e) 4 m, (f) 5 m, (g) 6 m, (h) 7 m, (i) 8 m, and (j) 9 m.

## 5. Discussion

### 5.1. Potential Applications of Data-Driven Models in the Daylight Design of Buildings

In the current study, we developed four machine learning algorithms that predicted the distribution of indoor illuminance. While the use of data-driven methods in the estimation of daylighting performance is a recent field of research, it is one that could have large implications on how daylighting is estimated in the early stages of building design. This is because data-driven predictive models, once trained, require less time and expertise to provide daylight estimations compared to simulation programs, while at the same time providing accurate estimations. Similarly, using data-driven methods to predict daylighting behavior in the early stage of building design is likely to become a preferred alternative to scale modeling methodologies, which are often associated with high costs and difficulties in replicating real-life daylighting scenarios. Additionally, predictive models are better alternatives to the commonly used split flux daylight factor method, which has been reported to be too simple

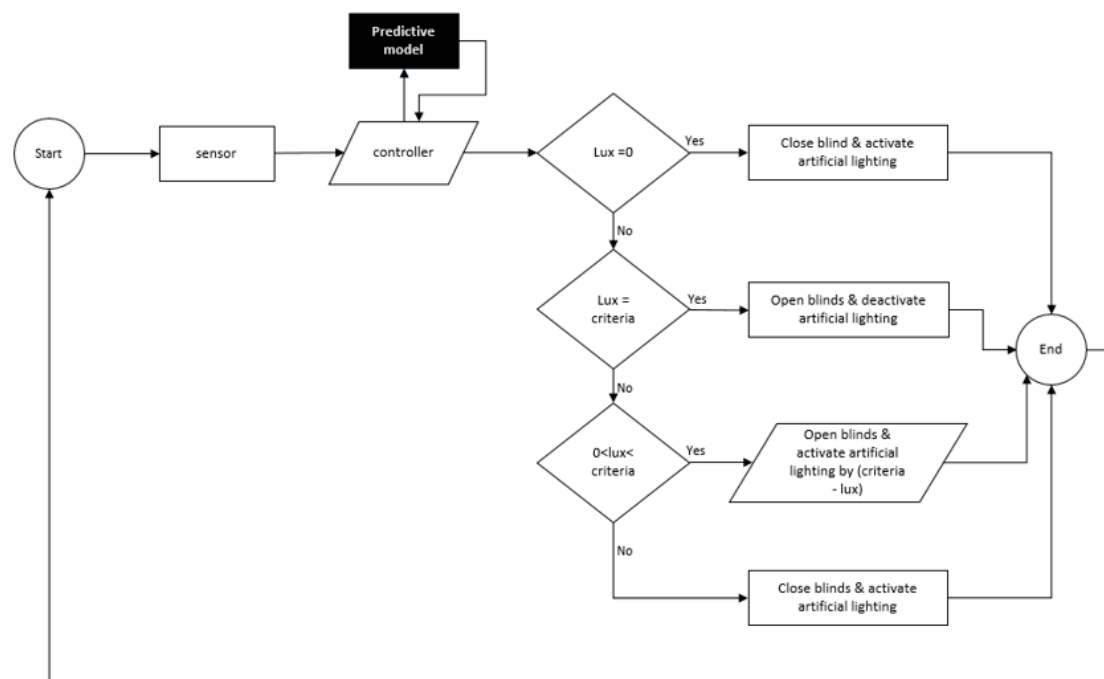
to capture the complex nature of daylighting behavior, and thus results in inaccurate estimations of daylight levels [40]. Figure 9 shows a schematic diagram illustrating how an indoor illuminance predictive model, such as the ones developed here, could be used to predict illuminance distribution in the early stages of building design.



**Figure 9.** The potential use of illuminance-predictive models in the early stages of building design.

### 5.2. Potential Applications of Data-Driven Models in Automated Daylight Control Systems

Properly trained indoor illuminance predictive models can be employed in daylighting control systems. Current methods used in, for example, the control of motorized blinds and artificial lighting, utilize open-loop methods and closed-loop methods [41]. An open-loop control system is designed with a single sensor located on the outside or window area of an indoor space. The sensor sends signals to the controller, prompting the controller to automatically adjust blinds or artificial lighting depending on the levels of illumination available outside. This kind of control system, however, fails to account for the levels of lighting already available in indoor spaces (i.e., from electric lighting) and is thus likely to allow an influx of light, causing visual glare. To address the shortcomings of open-loop systems, closed-loop systems are designed with two sensors: an indoor and an outdoor sensor. The outdoor sensor sends signals to the controller regarding the amount of illumination available outside. At the same time, the controller receives signals from the indoor sensor regarding the desired amount of illumination indoors (e.g., based on the type of visual task at hand or the amount of lighting available from artificial light sources). Upon receiving the signals from the outdoor and indoor sensors, the controller compares the two signals and adjusts the blinds or artificial lighting to provide the desired amount of illuminance levels indoors. While these two methods possess individual advantages, as discussed by Mukherjee [41], the integration of well-trained predictive models in daylighting control systems offers an alternative way to control daylighting systems using few sensors. Figure 10 is a schematic diagram illustrating how a daylighting predictive model could be integrated into a control system. In the theoretical system below, a single outdoor sensor sends signals to the controller regarding outdoor conditions. The controller feeds the received information into a pre-trained indoor illuminance predictive model, which subsequently determines or predicts the amount of indoor illuminance likely to be available depending on the status of outdoor conditions (i.e., inputs from the controller) at a given moment. The predicted indoor illuminance value is then passed back to the controller, which goes through a series of decision loops and adjusts the blinds system accordingly.



**Figure 10.** The potential integration of predictive models in blinds control systems.

### 5.3. Potential Applications in Data-Driven Models in the Preemptive Control of Daylighting

We also developed LSTM models that forecast indoor luminance levels at a particular future time,  $t$ , using patterns learned from past data values (e.g., the previous 1 h) of indoor illuminance. Currently, two studies have attempted to forecast indoor illuminance from a time-series modeling approach. Kurian et al. [16], using simulated data, demonstrated how neural fuzzy inference systems (NFISs) algorithms can be used to forecast indoor illuminance at a particular future time, allowing easy integration of daylighting schemes into buildings. Similarly, Waballah et al. [15] compared four time-series algorithms in the prediction of indoor illuminance levels at a particular future time using field measurements from a light pipe system. However, ANFIS models are based on “IF-THEN” rules usually dictated by experts based on a specific set of data, and are thus unable to learn continuously [40]. In addition, in conventional time-series models such as SARIMA and ARIMA, the gradient tends to dominate future weight adaptation or get lost over time, which can affect the accuracy of the forecasted values [42]. To address the shortcoming of traditional time-series modeling, LSTM algorithms were developed. LSTM models have an extended memory and can remember past information for longer periods than traditional time-series models [43]. As such, LSTM models are better suited in attempts to forecast future indoor illuminance distribution levels.

### 5.4. Optimum Machine Learning Algorithms for the Prediction of Indoor Illuminance Levels

Another contribution of our study deals with optimum algorithms for predicting indoor illuminances. There are many algorithms that can be used to predict daylighting behavior, and they differ in their predictive capabilities. Decisions regarding which algorithms to use must consider several factors such as ease of training the models, prediction time, and computational load. Previous studies have mainly relied on artificial neural network (ANN) algorithms to predict indoor daylighting illuminances [9,12,13]. Decision trees [13] and support vector machines [14] have also been used to study daylight behavior. We compared four machine learning algorithms that are commonly used in other areas of building design and identified the optimum algorithm for daylighting predictions. Our results showed that the DNN model outperformed the other three models (see Section 4.3). Our results differ from the results of Ahmed et al. [13], who reported better prediction performance of indoor daylight illuminances by RF algorithms than ANN algorithms.

### 5.5. Limitations and Future Research

The current study faced certain limitations. The main limitation was related to the nature of the data used to train and validate the developed models. We used data simulated using a Radiance-based daylight simulation engine. This is because designing for daylighting performance is a complicated task. It requires a thorough consideration of multiple factors, including building factors such as WWR and wall reflectance, weather factors such as solar irradiation, and time-related factors such as time of the day and season. It is unlikely to obtain such diverse data using real data from existing buildings. Using simulations, therefore, allows for the manipulation of different elements (e.g., WWR) and enables the developed models to be trained on a wide range of influential factors. However, despite the difficulties associated with obtaining daylighting data from real buildings, data from real buildings (e.g., buildings with different configurations) are likely to provide more accurate models than simulated data. Therefore, future studies should develop indoor illuminance-predictive models trained using real field data from multiple buildings with different design configurations. Furthermore, the current study considered daylight behavior in a square building of 10 m × 10 m × 2.7 m, which is not necessarily a general representative of many built spaces despite being utilized in previous daylight studies [9]. Future studies must attempt to establish a prototype model that can be used for daylight studies.

Secondly, due to limited computational capacity, we used manual splitting techniques to obtain two datasets: training and validation datasets. For similar reasons (i.e., limited computational capacities), we also used manual tuning to identify the best parameter combinations that would result in optimum models. Future studies, especially those attempting to develop virtual sensors for actual deployment in daylighting control schemes, should employ advanced optimization techniques (e.g., cross-validation method and Bayesian processes) to improve the accuracy and general performance of the ML-based daylight models.

The third limitation deals with how most machine learning models work in general. For example, unlike decision tree algorithms, which are transparent, deep learning models are opaque (i.e., black-box models), and it is therefore nearly impossible to determine how they make certain decisions. Consequently, it is difficult to determine cause–effect relationships among variables from such models. However, current efforts in machine learning research have made significant strides in developing methods for the extraction of causal relationships from machine learning models. For example, Athey and Imbens [43] propose diverse tuning techniques that can be used in machine learning algorithms to solve causal relationship tasks.

## 6. Conclusions

In this study, we developed five machine learning algorithms that predict the distribution of indoor daylight illuminances. All developed models, except for the GLM, were capable of predicting indoor illuminance levels with high accuracy ( $R^2$  value above 0.9). As such, we demonstrated that machine learning algorithms could be used to accurately predict the distribution of indoor illuminance. These kinds of models offer considerable benefits in terms of cost, time, and ease of application to architects in the process of estimating daylight performance in the early stages of building design. In addition, machine learning-based predictive models can be used in building control systems for automated blinds and artificial lighting to improve energy efficiency and visual comfort in indoor environments. However, the current study faced certain limitations particularly in terms of the daylight data used in training and validating the developed models (obtained from simulated experiments). Future studies should endeavor to assess the application of ML algorithms to model indoor daylight illuminances using datasets from actual buildings rather than those obtained from simulation tools.

**Author Contributions:** Conceptualization, J.N. and G.K.; methodology, J.N.; software, J.N. and A.I.; validation, J.N., G.Y.Y., and G.K.; formal analysis, J.N.; investigation, J.N.; resources, G.Y.Y. and G.K.; data curation, J.N. and A.I.; writing—original draft preparation, J.N.; writing—review and editing, J.N., G.Y.Y., and G.K.; visualization, J.N.; supervision, G.K.; funding acquisition, G.Y.Y. and G.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The results presented in the current article appear as a part of a Ph.D. dissertation submitted to Kyung Hee University entitled, “Machine Learning in Buildings: Predicting daylighting and thermal performance in indoor environments”.

## References

1. Yu, X.; Su, Y. Daylight availability assessment and its potential energy saving estimation –A literature review. *Renew. Sustain. Energy Rev.* **2015**, *52*, 494–503. [\[CrossRef\]](#)
2. Shishegar, N.; Boubekri, M. Natural Light and Productivity: Analyzing the Impacts of Daylighting on Students’ and Workers’ Health and Alertness. *IJACEBS* **2016**, *3*. [\[CrossRef\]](#)
3. Chen, X.; Zhang, X.; Du, J. Exploring the effects of daylight and glazing types on self-reported satisfactions and performances: A pilot investigation in an office. *Archit. Sci. Rev.* **2019**, *62*, 338–353. [\[CrossRef\]](#)
4. Bellia, L.; Bisegna, F.; Spada, G. Lighting in indoor environments: Visual and non-visual effects of light sources with different spectral power distributions. *Build. Environ.* **2011**, *46*, 1984–1992. [\[CrossRef\]](#)
5. Ayoub, M. 100 Years of daylighting: A chronological review of daylight prediction and calculation methods. *Sol. Energy* **2019**, *194*, 360–390. [\[CrossRef\]](#)
6. Boccia, O.; Zazzini, P. Daylight in buildings equipped with traditional or innovative sources: A critical analysis on the use of the scale model approach. *Energy Build.* **2015**, *86*, 376–393. [\[CrossRef\]](#)
7. Jakica, N. State-of-the-art review of solar design tools and methods for assessing daylighting and solar potential for building-integrated photovoltaics. *Renew. Sustain. Energy Rev.* **2018**, *81*, 1296–1328. [\[CrossRef\]](#)
8. Ochoa, C.E.; Aries, M.B.C.; Hensen, J.L.M. State of the art in lighting simulation for building science: A literature review. *J. Build. Perform. Simul.* **2012**, *5*, 209–233. [\[CrossRef\]](#)
9. Lorenz, C.L.; Packianather, M.; Spaeth, A.B.; De Souza, C.B. Artificial neural network-based modelling for daylight evaluations. In Proceedings of the 2018 Symposium on Simulation for Architecture and Urban Design (SimAUD 2018), San Diego, CA, USA, June 2018; Society for Modeling and Simulation International (SCS): Delft, The Netherlands, 2018.
10. Zomorodian, Z.S.; Tahsildoost, M. Assessing the effectiveness of dynamic metrics in predicting daylight availability and visual comfort in classrooms. *Renew. Energy* **2019**, *134*, 669–680. [\[CrossRef\]](#)
11. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260. [\[CrossRef\]](#)
12. Kazanasmaz, T.; Günaydin, M.; Binol, S. Artificial neural networks to predict daylight illuminance in office buildings. *Build. Environ.* **2009**, *44*, 1751–1757. [\[CrossRef\]](#)
13. Ahmad, M.W.; Hippolyte, J.-L.; Mourshed, M.; Rezgui, Y. Random forests and artificial neural network for predicting daylight Illuminance and energy consumption. In Proceedings of the 15th Conference of International Building Performance Simulation Association, San Francisco, CA, USA, March 2018.
14. Zhou, S.; Liu, D. Prediction of daylighting and energy performance using artificial neural network and support vector machine. *Am. J. Civ. Eng. Archit.* **2015**, *3*, 1–8.
15. Waheeb, W.; Ghazali, R.; Ismail, L.H.; Kadir, A.A. Modelling and forecasting indoor illumination time series data from light pipe system. In *International Conference of Reliable Information and Communication Technology*; Springer: Cham, Switzerland, 2018; pp. 57–64.
16. Kurian, C.P.; George, V.I.; Bhat, J.; Aithal, R.S. ANFIS model for the time series prediction of interior daylight illuminance. *Int. J. Artif. Intell. Mach. Learn.* **2006**, *6*, 35–40.
17. Mathew Biju, S. Analyzing the predictive capacity of various machine learning algorithms. *IJET* **2018**, *7*, 266. [\[CrossRef\]](#)
18. Ibrahim, I.A.; Khatib, T. A novel hybrid model for hourly global solar radiation prediction using random forests technique and firefly algorithm. *Energy Convers. Manag.* **2017**, *138*, 413–425. [\[CrossRef\]](#)
19. Biau, G.; Scornet, E. A random forest guided tour. *TEST* **2016**, *25*, 197–227. [\[CrossRef\]](#)



20. Hayes, M.M.; Miller, S.N.; Murphy, M.A. High-resolution landcover classification using Random Forest. *Remote Sens. Lett.* **2014**, *5*, 112–121. [\[CrossRef\]](#)
21. Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J.C.; Sheridan, R.P.; Feuston, B.P. Random forest: A classification and regression tool for compound classification and QSAR modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1947–1958. [\[CrossRef\]](#)
22. Qi, C.; Chen, Q.; Fourie, A.; Zhang, Q. An intelligent modelling framework for mechanical properties of cemented paste backfill. *Miner. Eng.* **2018**, *123*, 16–27. [\[CrossRef\]](#)
23. Aljarah, I.; Faris, H.; Mirjalili, S. Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Comput* **2018**, *22*, 1–15. [\[CrossRef\]](#)
24. Shanmuganathan, S. Artificial Neural Network Modelling: An Introduction. In *Artificial Neural Network Modelling; Studies in Computational Intelligence*; Shanmuganathan, S., Samarasinghe, S., Eds.; Springer International Publishing: Cham, Switzerland, 2016; Volume 628, pp. 1–14, ISBN 978-3-319-28493-4.
25. Suzuki, K. (Ed.) *Artificial Neural Networks: Methodological Advances and Biomedical Applications*; BoD—Books on Demand: Norderstedt, Germany, 2011.
26. Ermis, K.; Ere, A.; Dincer, I. Heat transfer analysis of phase change process in a finned-tube thermal energy storage system using artificial neural network. *Int. J. Heat Mass Tran.* **2007**, *50*, 3163–3175. [\[CrossRef\]](#)
27. Sak, H.; Senior, A.W.; Beaufays, F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *arXiv* **2014**, arXiv:1402.1128.
28. Kong, W.; Dong, Z.Y.; Jia, Y.; Hill, D.J.; Xu, Y.; Zhang, Y. Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network. *IEEE Trans. Smart Grid* **2019**, *10*, 841–851. [\[CrossRef\]](#)
29. Weninger, F.; Erdogan, H.; Watanabe, S.; Vincent, E.; Le Roux, J.; Hershey, J.R.; Schuller, B. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR. In *Latent Variable Analysis and Signal Separation; Lecture Notes in Computer Science*; Vincent, E., Yeredor, A., Koldovský, Z., Tichavský, P., Eds.; Springer International Publishing: Cham, Switzerland, 2015; Volume 9237, pp. 91–99, ISBN 978-3-319-22481-7.
30. Crawley, D.B.; Lawrie, L.K.; Winkelmann, F.C.; Buhl, W.F.; Huang, Y.J.; Pedersen, C.O.; Strand, R.K.; Liesen, R.J.; Fisher, D.E.; Witte, M.J.; et al. EnergyPlus: Creating a new-generation building energy simulation program. *Energy Build.* **2001**, *33*, 319–331. [\[CrossRef\]](#)
31. McNeil, A.; Lee, E.S. A validation of the Radiance three-phase simulation method for modelling annual daylight performance of optically complex fenestration systems. *J. Build. Perform. Simul.* **2013**, *6*, 24–37. [\[CrossRef\]](#)
32. Yoon, Y.; Moon, J.W.; Kim, S. Development of annual daylight simulation algorithms for prediction of indoor daylight illuminance. *Energy Build.* **2016**, *118*, 1–17. [\[CrossRef\]](#)
33. Reinhart, C.F.; Andersen, M. Development and validation of a Radiance model for a translucent panel. *Energy Build.* **2006**, *38*, 890–904. [\[CrossRef\]](#)
34. Heaton, J. *Introduction to Neural Networks with Java*; Heaton Research, Inc.: St. Louis, MO, USA, 2008.
35. Hecht-Nielsen, R. Theory of the Backpropagation Neural Network\*\*Based on “nonindent” by Robert Hecht-Nielsen, which appeared in Proceedings of the International Joint Conference on Neural Networks 1, 593–611, June 1989. © 1989 IEEE. In *Neural Networks for Perception*; Elsevier: Amsterdam, The Netherlands, 1992; pp. 65–93, ISBN 978-0-12-741252-8.
36. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [\[CrossRef\]](#)
37. Willmott, C.; Matsuura, K. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Clim. Res.* **2005**, *30*, 79–82. [\[CrossRef\]](#)
38. Chai, T.; Draxler, R.R. Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature. *Geosci. Model Dev.* **2014**, *7*, 1247–1250. [\[CrossRef\]](#)
39. Garson, G.D. *Interpreting Neural-Network Connection Weights*; AI Expert, Miller Freeman, Inc: San Francisco, CA, USA, 1991; Volume 6, pp. 46–51.
40. Jang, J.-S.R. ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Trans. Syst. Man Cybern.* **1993**, *23*, 665–685. [\[CrossRef\]](#)
41. Mukherjee, S.; Birru, D.; Cavalcanti, D.; Shen, E.; Patel, M.; Wen, Y.J.; Das, S. Closed loop integrated lighting and daylighting control for low energy buildings. In Proceedings of the 2010 ACEEE 2010, Summer Study on Energy Efficiency in Buildings, California, CA, USA, 15–20 August 2010; pp. 252–269.

42. Gamboa, J.C.B. Deep Learning for Time-Series Analysis. *arXiv* **2017**, arXiv:1701.01887.
43. Athey, S.; Imbens, G.W. Machine Learning Methods for Estimating Heterogeneous Causal Effects. *Stat* **2015**, *1050*, 1–26.



© 2020 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 (<http://creativecommons.org/licenses/by/4.0/>).