Article

Application of Multilayer Extreme Learning Machine for Efficient Building Energy Prediction

Muideen Adegoke [1], **Alaka Hafiz [1],***, **Saheed Ajayi [2]** and **Razak Olu-Ajayi [1]**

1   Big Data Technologies and Innovation Laboratory, University of Hertfordshire, Hatfield AL10 9AB, UK
2   School of Built Environment, Engineering and Computing, Leeds Beckett University, Leeds LS2 8AG, UK
*   Correspondence: h.alaka@herts.ac.uk

Abstract: Building energy efficiency is vital, due to the substantial amount of energy consumed in buildings and the associated adverse effects. A high-accuracy energy prediction model is considered as one of the most effective ways to understand building energy efficiency. In several studies, various machine learning models have been proposed for the prediction of building energy efficiency. However, the existing models are based on classical machine learning approaches and small datasets. Using a small dataset and inefficient models may lead to poor generalization. In addition, it is not common to see studies examining the suitability of machine learning methods for forecasting the energy consumption of buildings during the early design phase so that more energy-efficient buildings can be constructed. Hence, for these purposes, we propose a multilayer extreme learning machine (MLELM) for the prediction of annual building energy consumption. Our MLELM fuses stacks of autoencoders (AEs) with an extreme learning machine (ELM). We designed the autoencoder based on the ELM concept, and it is used for feature extraction. Moreover, the autoencoders were trained in a layer-wise manner, employed to extract efficient features from the input data, and the extreme learning machine model was trained using the least squares technique for a fast learning speed. In addition, the ELM was used for decision making. In this research, we used a large dataset of residential buildings to capture various building sizes. We compared the proposed MLELM with other machine learning models commonly used for predicting building energy consumption. From the results, we validated that the proposed MLELM outperformed other comparison methods commonly used in building energy consumption prediction. From several experiments in this study, the proposed MLELM was identified as the most efficient predictive model for energy use before construction, which can be used to make informed decisions about, manage, and optimize building design before construction.

Keywords: energy prediction; building energy consumption; machine learning; energy efficiency

1. Introduction

In the building sector, it is clear that energy demand is rising greatly, due to the increased population, along with other factors such as social demand and rapid urbanization [1]. Building construction must be energy efficient and sustainable because it contributes a large percentage to the world's energy consumption and greenhouse gas emissions [2]. Hence, it is vital to understand the energy consumption pattern in the building construction phase. This has attracted the interest of many researchers and practitioners, who have contributed to concepts of building energy efficiency and savings [3–6].

Without doubt, due to modern technology in the building sector, such as the development of industry 4.0, the insights behind the data can be explored and investigated using artificial intelligence (AI) techniques, including machine learning (ML).

One way to analyze building energy consumption patterns is based on physical simulation techniques. In this case, tools such as EnergyPlus and DOE-2 are often used. One of the drawbacks with this method is that to use these tools, the users must provide a

large amount of detailed information, namely details about the materials, roofs, windows, HVAC (heating, ventilation, and air conditioning) systems, a building's geometry, thermal settings, occupancy loads, lighting, equipment, and weather conditions. Due to the long processing times and the lack of information, these tools are insufficient for detailed building energy consumption investigations [6].

In order to tackle the aforementioned issue, an alternative approach often explored has been the data-driven approach [1]. In this method, building energy performance is assessed based on historical data. Hence, with the availability of open data and the emergence of the Internet of Things (IoT) and AI, building energy consumption performance can be studied and understood well. With these opportunities, various machine learning methods have been proposed to realize building energy consumption patterns. Some of the machine learning methods and models found in the literature that have been explored for the study of building energy consumption patterns include support vector machine (SVM) [7–9], tree-based methods, such as decison tree [1,10,11] and random forest [12–14], artificial neural networks (ANNs) [15–18], adaptive neuro-fuzzy inferring system (ANFIS) [19–21], and others.

There are several benefits associated with efficient energy consumption. When energy is used efficiently, the cost of energy is reduced. In addition, it causes greenhouse gas reduction and improved energy security. Thus, it increases monetary savings on energy utilities and other factors [22,23].

It is important to mention that in the literature, most of the models for building energy prediction have been developed around a few data samples or a single dataset. Thus, this leads to poor generalization of the models for energy use prediction. To our knowledge, only a few studies have evaluated the predictive performance of models on building energy consumption using multiple datasets. In addition, only a small number of the existing methods have used more than 1000 buildings in a dataset to train the model for good generalization performances. Obviously, the accuracy prediction of machine learning models is dependent on the training algorithm and the quality and quantity of the dataset used to train a model [24]. In choosing the best algorithm among the comparison agorithms for building prediction, it is equally important to evaluate the algorithms on the same dataset, otherwise, it cannot be concluded that one algorithm is better than the other [25]. Hence, in this study, we utilized 5000 data samples, a large dataset. In addition, to improve the quality of the dataset used in this study, we explored a feature extraction technique. In production application, an important point to consider is the execution time of the machine learning algorithm to prevent dragging and improve decision making. Considering the aforementioned points and gaps, we propose a multilayer extreme learning machine (ML-ELM) for annual building energy consumption prediction. Our proposed method fused autoencoders (AEs) and an extreme learning machine (ELM) into a single model. The AEs were utilized to extract features from the data in order to improve the quality of the data employed in our machine learning algorithm. In addition, we utilized ELM for decision making. The ELM is suitable for decision making due to its algorithm's universal approximation capability [26]. Hence, in this paper, the fused model is called the multilayer extreme learning machine (MLELM). This study is unique because it investigated how effective the combination of stacks of autoencoders (AEs) and the extreme learning model was at predicting energy consumption across a range of buildings. In addition, the proposed model was compared with several machine learning models, namely support vector machine (SVM), random forest (RF), decision tree (DT), multilayer perceptron (MLP), linear regression (LR), stacking, deep neural network (DNN), and the original extreme learning machine (ELM).

In this investigation, we evaluated the proposed model and other comparison models using the standard machine learning evaluation metrics, namely the mean square error (MSE), the mean absolute error (MAE), the root mean square error (RMSE), and the co-efficient of determination (R^2-score). In addition, to evaluate how effective the proposed

MLELM was in predicting the annual building energy consumption, we investigated the percentage improvement ratio (PIR) of the MLELM's performance.

In essence, this study aims to develop a model that is faster to train and also provides accurate information about building energy consumption. This model enables the building designer to input the vital parameters of a building's design. In response, it produces/forecasts the annual average building energy consumption based on a large dataset of multiple buildings.

In summary, the key contributions of this research are as follows:

- We built a fast learner machine learning model that provides accurate annual average building energy consumption using a large dataset with multiple buildings.
- We built autoencoders (AEs) to extract features from the input data and an extreme learning machine to make decisions on energy consumption predictions.
- We proposed a multilayer extreme learning machine (MLELM), which fused autoencoders (AEs) and an extreme learning machine (ELM).
- We compared the performance of the proposed models (MLELM) and other machine learning methods commonly used for building energy consumption prediction.
- We investigated the percentage improvement ratio (PIR) and the computation time of the compared models.

The remainder of the paper proceeds as follows. Section 2 provides a review of the relevant literature. Section 3 provides the theory and the proposed models. Section 4 discusses the research methodology, where the datasets and evaluation metrics are presented. Furthermore, Section 5 presents the results and discussion. In Section 6, we conclude and offer some remarks.

2. Related Works

In much of the related literature, the existing methods have focused on predicting or forecasting building energy consumption. These methods have been based on traditional machine learning algorithms. In addition, it is evident that there is no general rule or agreement that a particular machine learning algorithm is the most suitable for energy prediction [23,27]. Generally, each machine learning algorithm/model has a unique strength that makes it performs well on a particular problem. Similarly, a machine learning model that performs well on one problem may perform poorly on other tasks. In addition, some machine learning models are easy to implement, while others require longer training time. Hence, it is important to explore different machine models for a prediction problem. In the literature, several machine learning models/algorithms, such as support vector machine (SVM), artificial neural network (ANN), decision tree (DT), and many others, have been utilized for energy use predictions [15,28]. However, fewer studies have explored flat-structure neural networks, such as extreme learning machines and broad learning systems, for energy prediction. Table 1 provides the details of some of the machine learning algorithms that have been applied for the energy consumption prediction task. In addition, the references in Table 1 and the references therein detail other machine learning algorithms that have been applied for predicting building energy consumption. The algorithms can be grouped into neural network-based algorithms, such as ANN, tree-based methods, such as decision tree (DT) and random forest (RF), statistically based algorithms, such as Bayes (GBN), and others, such as K-nearest neighbors (KNN), support vector machine (SVM), linear regression (LR), etc.

In addition to the data-driven machine learning methods presented in Table 1, many other methods have been proposed for predicting building energy consumption. This issue has gained researchers' attention in recent years. Hence, many review studies [1] have focused on building energy consumption prediction. For instance, according to Zhao and Magoulès [29], building energy consumption prediction methods can be classified as elaborate engineering methods, simplified engineering methods, statistical methods, artificial neural networks, SVM-based methods, and gray models. In addition, they analyzed the model complexity, ease of use, speed of runtime, inputs needed, and accuracy of the

models. Further, in [15], ANN-based, SVM-based, and hybrid methods were reviewed, along with their advantages and disadvantages.

Table 1. Machine learning methods for energy prediction in the literature.

Building Type	Type of Energy Consumption Considered	Size of Sample Used	Type of Learning Algorithm Used	Performances in Term of RMSE	References
Non-Residential	Hourly load	507	Artificial neural network (ANN)	5.71	[30]
Non-Residential	Hourly load	1	Support vector machine (SVM) Radial basis function NN (RBFNN) General regression neural network (GRNN) Artificial neural network (ANN)	1.17 1.43 1.19 2.22	[31,32]
Non-Residential	Hourly load	2	Stacking Random forest (RF) Decision tree (DT) Support vector machine Extreme gradient boosting K-nearest neighbor (KNN)	13.81 26.34 19.20 16.12 15.37 17.81	[33]
Non-Residential	Hourly load	5	Random tree (RT) Random forest (RF) M5 model trees	7.99 6.09 5.53	[6]
Residential	Hourly load	N/S	Decision tree (DT) Support vector machine (SVM) Artificial neural network (ANN) General linear regression (GLR)	1.84 1.65 1.68 1.74	[34]

Similarly, as part of comprehensive reviews on energy consumption, in [35], the state-of-the-art studies on building energy modeling and prediction, as well as modeling of the critical components of buildings were examined (e.g., photovoltaic power generation), building energy modeling for demand response (such as weather forecasting), agent-based building energy modeling, and system identification for building energy modeling.

Many of these studies reviewed energy consumption prediction research efforts with a focus on machine learning methods/algorithms used in the research. Many machine learning methods have universal approximation capabilities, making them suitable for many applications, such as energy prediction, parameter selection for dynamic models, etc., [36]. Specifically, data-driven-based machine learning techniques are common methods for building energy consumption prediction in the literature. Unlike Naive methods such as the time of the week (TOTW) algorithm, machine learning techniques can generalize well and take into account many factors that can affect building energy consumption for better decision making.

From the literature survey, for instance, in Table 1 and the references therein, we notice that the existing works used less than a thousand data samples for building energy prediction. Generally, using small data samples for building machine learning models will not generalize well on unseen data, and it implies poor model performances would be obtained. Moreover, in the literature, fewer studies have focused on designing and investigating the suitability of machine learning techniques for predicting building energy consumption at the early design stage of the building. However, studying the energy consumption at the early stage of building construction can prevent or minimize the number of inefficient buildings being constructed. In addition, the existing works on energy prediction have focused on non-residential buildings and hourly load. However, in this current study, we targeted residential buildings. In other words, this research focused on predicting and forecasting the annual building energy consumption of residential buildings. Unlike other studies, we used a large dataset of 5000 data points, which covered a large distribution of residential buildings. Hence, to predict building energy use, we proposed a multilayer extreme learning machine (MLELM) to predict annual building energy consumption at the early stage of construction using a large dataset. Our proposed

MLELM was a flat-structure neural network. It had a fast learning speed. In addition, it had universal approximation capability. In the following sections, we present the theoretical details of the proposed multilayer extreme learning machine and the description of the data employed in this research.

3. Theory

This section presents the proposed multilayer extreme learning machine and its building blocks. The building blocks are autoencoders (AE) and an extreme learning machine (ELM).

3.1. Extreme Learning Machine (ELM)

An ELM is a single-layer feed-forward neural network. Structurally, it has three layers: an input layer, a hidden layer, and an output layer. Figure 1 shows the structure of a single-layer feed-forward (SLFNN) ELM network. It is widely used in many applications [37–39]. It has a fast learning speed with good generalization performance [40]. It is regarded as a universal approximation machine learning model. With sufficient hidden nodes, an ELM network can approximate any function [41]. In the ELM concept, the weights (**W**) and bias (**b**) between the input layer and the hidden layer are randomly generated and fixed throughout the training time of the network. In the hidden layer, the product of the input data and weight plus the bias are computed. In addition, a nonlinear function commonly called the activation function, e.g., a sigmoid function, is then applied to the result. The output of the hidden nodes is passed to the output layer where the output weights β are computed. In the original ELM concept, the output weight β is computed using the least squares approach.
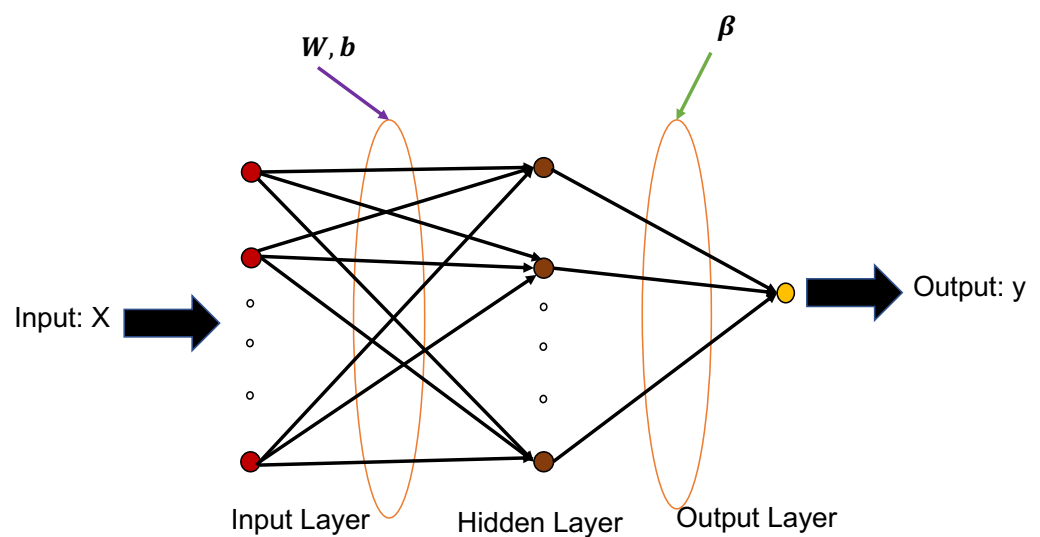
Figure 1. A single-layer feed-forward network.

The mathematical details of the ELM network are presented here. We consider that there is a dataset with N training pairs, with the training set $D_{train} = \{x_k, y_k\} : k = 1, \ldots, N\}$, where $x_k = [x_{k,1}, \ldots x_{k,D}]^T$ is a D dimensional vector of input data. In addition, y_k is the corresponding target output.

Suppose that there are L hidden nodes in the ELM network. Then, the output of the ELM network is given by

$$f(x) = \sum_{j=1}^{L} h_j(x)\beta_j, \tag{1}$$

where $\boldsymbol{\beta} = [\beta_1, \cdots, \beta_L]$, and β_j is the j-th output weight. Moreover, $h_j(\boldsymbol{x})$ is the activation function of the j-th node. In this paper, we consider the sigmoid activation function given by

$$h_j(\boldsymbol{x}) = \frac{1}{1 + exp^{-(xW_j + b_j)}}, \tag{2}$$

where W_j and b_j are the j-th component of $\boldsymbol{W}$ and $\boldsymbol{b}$, respectively. In the ELM concept, the objective function given by (3) is minimized to obtain the output weight β_j

$$
\begin{aligned}
J_{elm} &= \min_{\boldsymbol{\beta}} ||\boldsymbol{y} - \sum_{j=1}^{L} h_j(\boldsymbol{x})\beta_j||^2 + \lambda ||\boldsymbol{\beta}||^2 \\
&= \min_{\boldsymbol{\beta}} ||\boldsymbol{y} - \sum_{j=1}^{L} \boldsymbol{H}(\boldsymbol{x})\beta_j||^2 + \lambda ||\boldsymbol{\beta}||^2,
\end{aligned}
\tag{3}
$$

where λ is the regularizer parameter, $\boldsymbol{y} = [y_1, \cdots, y_N]$, $\boldsymbol{x} = [x_1, \cdots, x_N]$, and $\boldsymbol{H}$ is given by

$$
\boldsymbol{H} = \begin{bmatrix}
h_1(x_1) & h_1(x_2) & \cdots & h_1(x_N) \\
h_2(x_1) & h_2(x_2) & \cdots & h_2(x_N) \\
\vdots & \vdots & \ddots & \vdots \\
h_L(x_1) & h_L(x_2) & \cdots & h_L(x_N)
\end{bmatrix}^T .
$$

From (3) and (4), the output weight of the ELM network is given by

$$\text{Primal Space:} \quad \boldsymbol{\beta} = \left(\boldsymbol{H}^T\boldsymbol{H} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{H}^T\boldsymbol{y} \tag{4}$$

$$\text{Dual Space:} \quad \boldsymbol{\beta} = \boldsymbol{H}^T\left(\boldsymbol{H}\boldsymbol{H}^T + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{y}. \tag{5}$$

3.2. Autoencoder (AE)

An autoencoder (AE) is an unsupervised neural network. It can be used for dimension reduction or compression and feature extraction. Generally, the AE network consists of three parts namely an encoder, a code, and a decoder. There are several types of autoencoders based on how the AE network is trained. In this paper, we focus on an autoencoder based on an extreme learning machine. It is widely used in many applications because of its fast learning speed and good feature extraction ability [42]. Figure 2 shows the AE network. Unlike the ELM network (see Figure 1), AEs are trained to reconstruct their inputs instead of predicting the output $\boldsymbol{y}$. AEs are widely used in classification, regression, and other aspects of machine learning for feature extraction, dimension reduction, etc. Fusion of feature extractors and a classifier can increase the performance of a classifier [43,44].

For an AE network, in the training process, the construction error between the input and the output of the network is minimized. Figure 3 shows the typical process of an AE network. Furthermore, according to the AE concept, the input weight $\boldsymbol{W1}$ and bias $\boldsymbol{b1}$ of the encoder, shown in Figure 2, are randomly generated and fixed during the training. The output weight $\boldsymbol{W2}$ of the decoder is learned or trained to minimize the construction error given by

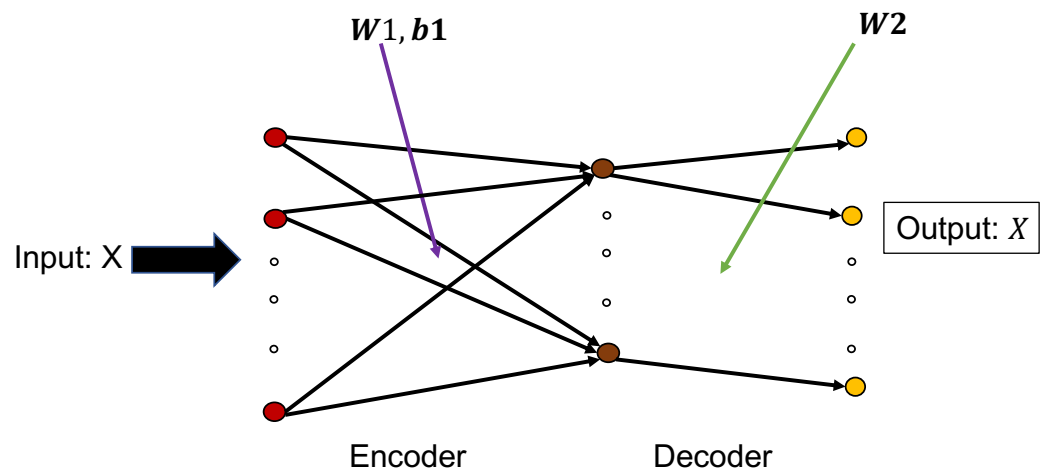$$E = ||\boldsymbol{X} - \hat{\boldsymbol{X}}||^2. \tag{6}$$
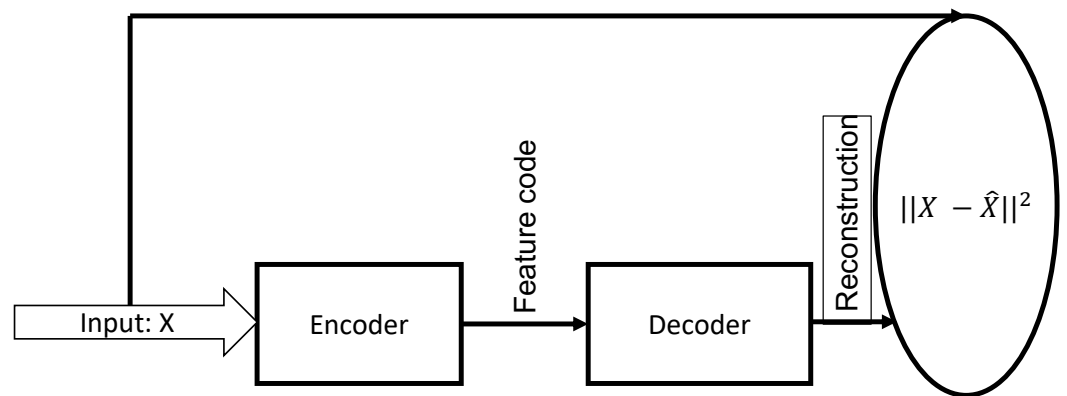
Figure 2. A single autoencoder network.

Figure 3. The general process of an AE network.

As shown in Figure 2, the input X is passed to the network and transformed into a hidden or latent space. In the ELM concept, the weight $W1$ and $b1$ are randomly generated, and the output of the hidden layer is computed. This output is given by

$$O_e = g(XW1 + b1), \tag{7}$$

where $g(.)$ is an activation function. The commonly used activation function is the sigmoid function. It is given by

$$\frac{1}{1 + exp(-x)}, \quad \text{where } x \text{ is the input.}$$

Thus, to compute the output $W2$, the error E (6) is minimized. Hence, the output weight of the AE can be computed, and it is given by

$$W2 = \left(O_e^T O_e\right)^{-1} O_e^T X. \tag{8}$$

Furthermore, in the ELM approach to the AE network, the features are encoded for the input X by using the computed output weight $W2$. This is given by

$$\hat{X} = g(XW2^T), \tag{9}$$

where $\hat{X}$ is the feature extracted. Hence, $\hat{X}$ can be fed to a regressor for predictions. In general, a deep/stack of AE can extract better features from the input data [45]. In the stacked AE or deep AE, the learned codes (features) from the previous AE are passed to

the next AE. The process continues in this manner until the maximum number of AEs is reached. Thus, the output of the final AE is passed to a regressor to make decisions.

3.3. Multilayer ELM (ML-ELM)

In this subsection, we present the proposed multilayer ELM (MLELM) for the prediction of annual building energy consumption. In the proposed approach, we used an ELM regressor. It had a fast learning speed and an excellent approximation capacity. It could approximate any function with sufficient hidden nodes. In our approach to modeling, we fused the AEs and the ELM network together. Moreover, to extract rich discriminative features from the input data X, we stacked a number of AEs together to form a stacked AE, and the output of the stacked AE was passed to the ELM regressor for the energy use prediction. Figure 4 illustrates the concept and approach proposed in this subsection.
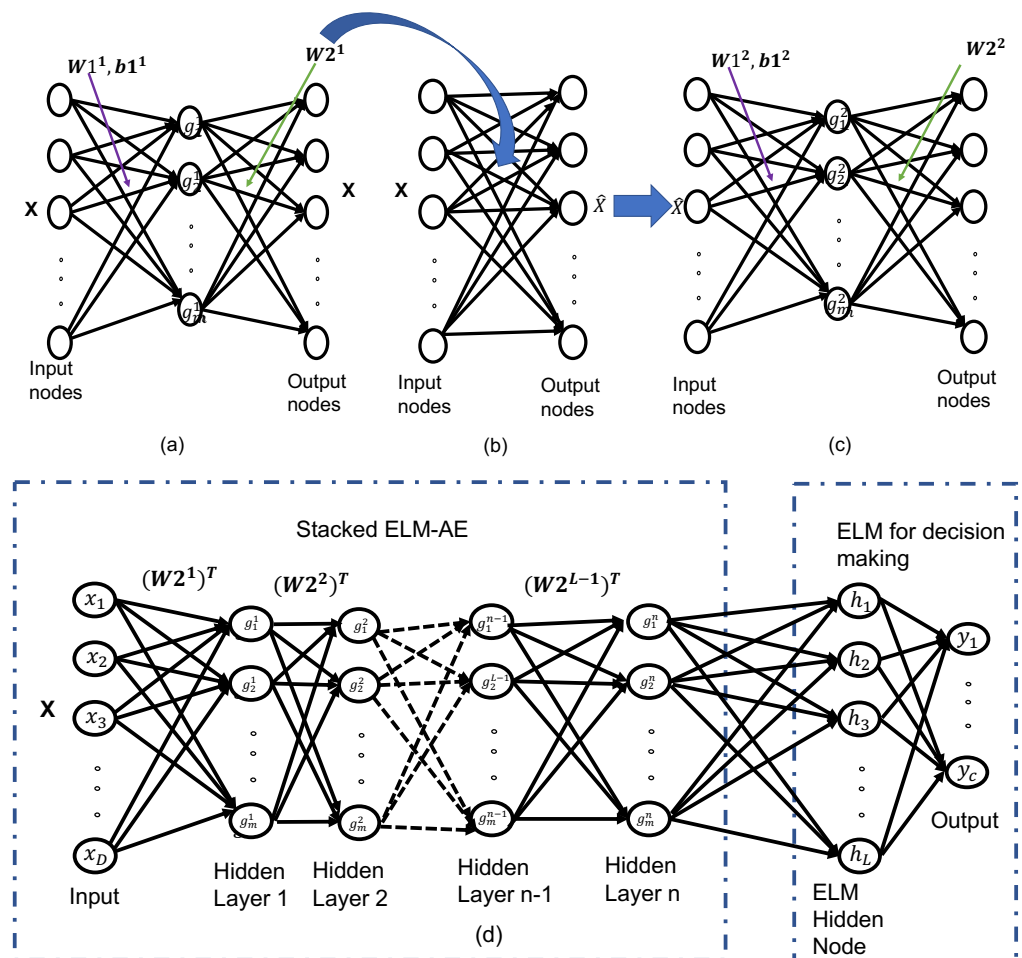
Figure 4. Architecture of the ML-ELM. It consists of feature encoding using stacks of AEs, a multilayer network with a randomized input weight and bias, and an ELM regressor for making the final decision. (**a**) is an autoencoder (**b**) extracts a new representation from input x, and (**c**) is a new autoencoder, it takes the new representation from (**b**) and extracts new features from it and (**d**) is the ML-ELM, it consist of stacks of AEs and ELM regressor for final decisions.

For instance, (a) is the single layer AE, and after learning its output weight $W2^1$ using (8), the learned output weight $W2^1$ was used to encode the input X. This was computed using (9), and the network is shown in (b). (c) shows the second AE, which took the learned code (features) from the first AE and learned a new output weight $W2^2$, and the next AE took the output of the previous AE as input, and the process continued. The output of the final AE was passed to the ELM regressor or classifier for the regression or

classification task. The proposed MLELM network, that is, the fusion of the stacked AEs and ELM regressor for decision making, is presented in (d). With (1) and (9), the output of the proposed MLELM network can be computed. The stacked AE was designed for feature encoding, and the ELM regressor was for regression.

4. Research Materials and Methodology

This study proposed a multilayer extreme learning machine for predicting annual energy consumption using a large dataset. The data collected for this research were from the United Kingdom (UK). In addition to the proposed machine learning method, we used other machine learning algorithms that are commonly used for prediction tasks. The algorithms implemented were multilayer perceptron (MLP), support vector machine (SVM), extreme learning machine (ELM), deep neural network (DNN), decision tree (DT), linear regressor (LR), K-nearest neighbor (KNN), stacking, and the proposed multilayer extreme learning machine (MLELM). The DNN has three layers, and it is trained using the backpropagation algorithm. In addition, the stacking method stacks a regularized linear regressor (LR) and support vector machine (SVM) together for a better performance. We developed each of the models using Python programming language on a Jupyter notebook.

In addition, the experiments were carried out using Python programming language. The experiments were performed using hardware with the following specifications: Apple MacBook Air with macOS Big Sur 11.7, 16 G RAM of an Apple MI Chip with 8 processing cores. First, exploration data analysis, such as data cleaning or preprocessing was carried out on the raw dataset. It was essential to remove NaN (not a number) values from the dataset, since a machine learning model cannot accept them. Further, the dataset was scaled between 0 and 1 in order to avoid heavy computations during model training.

In addition, in this study, the proposed energy consumption prediction framework had four steps. They were

- Data collection;
- Data preprocessing;
- Model development
  - feature extraction
  - training regressors;
- Model evaluation (testing).

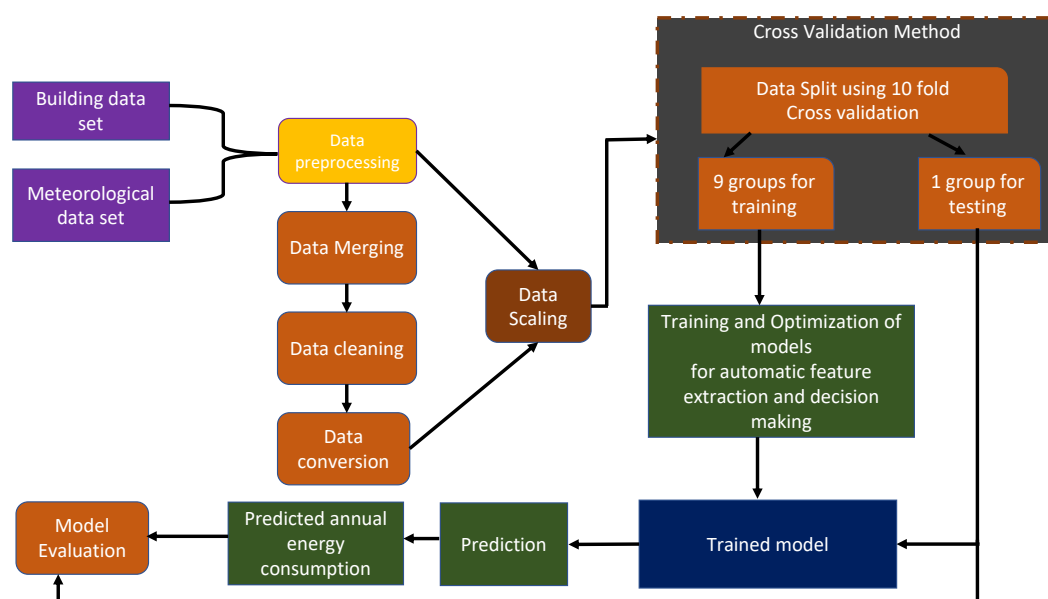Figure 5 shows the four sections of the proposed framework.

Figure 5. The flowchart of the proposed framework for annual energy prediction.

4.1. Data Collection

In this study, we used two types of data. They were meteorological data and building metadata. Building-related datasets were obtained from the Ministry of Housing Communities and Local Government (MHCLG). The collected building data contained energy consumption data of 5000 data points of residential building types in the UK. In addition, the data contained the metadata. Figure 6 shows the details of the building dataset.
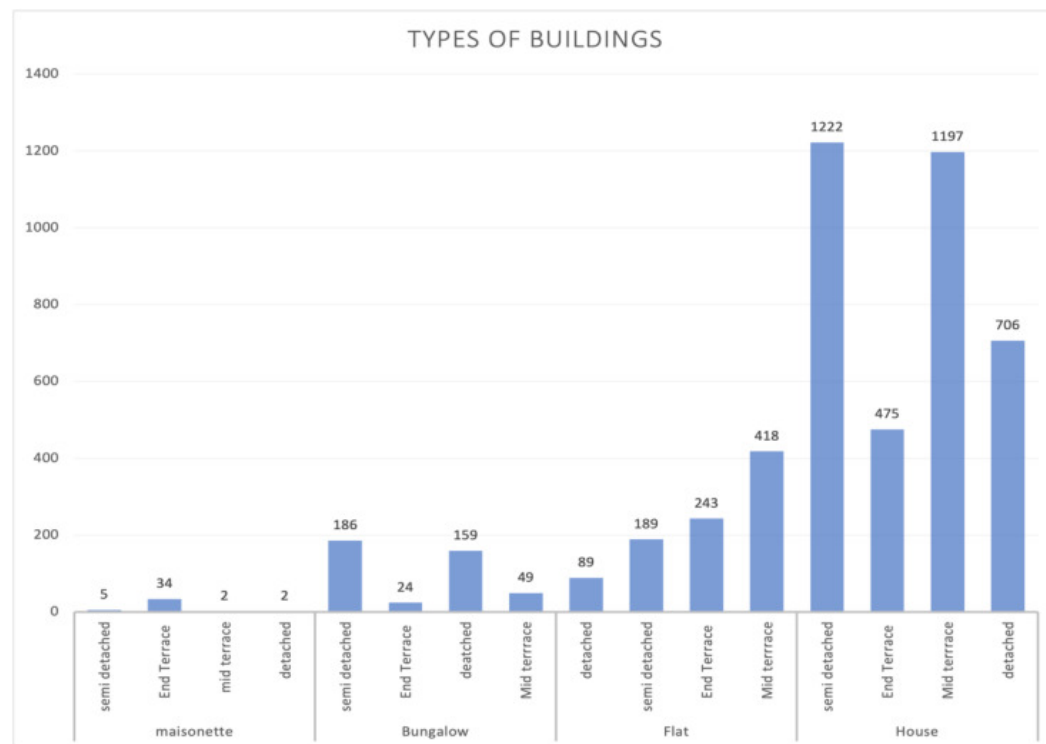
Figure 6. Different types of buildings from the building data utilized.

Building Metadata

Data on 5000 residential buildings within 10 UK area postcodes were collected. In other words, we utilized 500 residential buildings in each postcode. The collected data included the annual energy consumption for the year 2020 for each building. In addition, the metadata consisted of tunable parameters that could be altered during the design stage. The parameters included the wall description, floor level, number of habitable rooms, etc.

4.2. Meteorological Data

The collected meterological data contained weather features, namely wind speed, pressure, and temperature. The dataset was collected from the Meteostat repository. In [23], there is a detailed description of the dataset for the building metadata and meteorological data. A total of ten postcodes for residential buildings were analyzed. The meteorological data collected were the daily averages from 1 January 2020 to 31 December 2020. In consideration of its applicability beyond the UK, the meteorological data were averaged on a monthly basis rather than annually in accordance with the energy consumption data (to ensure the model worked in both high and moderate weather conditions).

4.3. Data Preprocessing

Machine learning requires good data. The initial process in a machine framework is data preprocessing. This is a set of processes where the data are prepared in a suitable form to be accepted by a machine learning model. This preliminary process gives the opportunity to detect inconsistencies in a dataset or invalid values in the data. In addition, it prevents inconsistency in the model comparison and performances during analysis [1,46].

4.3.1. Data Merging

Building a machine model with multiple datasets requires that the data be merged or combined. In other words, the meteorological dataset and the building dataset were merged. We used the postcode, a common variable from the two datasets to merge the building data and meteorological data. We utilized the pandas package to carry out the data merging.

4.3.2. Data Cleaning

Generally raw data contain invalid numbers or inconsistent values in the data samples. The process of removing the invalid data, such as outliers, and the treatment of invalid numbers, such as missing data, is called data cleaning. There were few missing values in the meteorological data. In [47], for the missing value, a replacement of the missing value using the mean of each feature column was used to resolve missing value problem. Hence, we followed the same concept; in this study, we replaced the missing data with the mean value of each column.

In the building dataset, there were 540 instances of missing values in the database. These missing values were deleted to prevent model development complexities and ambiguity in the training and testing phase.

4.3.3. Data Conversion

The collected dataset, specifically the building raw data, contained some categorical features, namely window energy efficiency, wall energy efficiency, and others. The feature or variables were assigned values to change the data into a form that was accepted by a machine learning model. The assigned values were: very good = highest value (5) to very poor = lowest value (0).

4.3.4. Data Scaling

In our exploration data analysis (EDA), we noticed that some of the data had very large values, e.g., between 1000 and 10,000, while some had small values between 0 and 5. These inconsistencies in data values can cause high computation complexity during the model training and testing. In addition, it may cause the machine learning algorithm to perform poorly. Hence, to prevent this, and for a fair comparison, this study adopted the common procedure of data preparation to eliminate these inconsistencies. Hence, the meteorological dataset and the building dataset were scaled from 0 to 1. We used the sklearn python package MinMaxScaler for the scaling. The sklearn Python package is a machine learning library that implements various algorithms including preprocessing techniques such as min–max scaling and standardization [48,49]. The formula of the MinMaxScaler used in the sklearn package to scale the data from 0 to 1 is given by

$$X_{min-max} = \frac{X - X_{min}}{X_{max} - X_{min}},$$

where X is the input data sample (the training dataset), X_{min} is the minimum value, and X_{max} is the maximum value of the input data sample.

4.4. Model Formulation

In our implementation, we utilized the cross-validation technique. Hence, the processed data were divided into a training set and test set. We used tenfold division and fifteenfold division methods. In the tenfold division, the data were randomly split into tenfold, and we ran the algorithms 10 times. In the first run, nine groups from the data were gathered as the training data and the other one group as the testing data. In the second run, we selected another one group for testing and the remaining nine groups for training the model. The process continued until the tenth run was reached. A similar process was followed for the fifteenfold division. The training data were fed into the proposed MLELM model and the other implemented algorithms, namely the artificial neural network (ANN),

support vector machine (SVM), linear regressor (LR), decision tree (DT), K-nearest neighbor (KNN), extreme learning machine (ELM), deep neural network (DNN), and stacking. The comparison algorithms were the state-of-the-art models commonly used for predictions.

The proposed MLELM model for the prediction of annual building energy consumption was described in Section 3.3. It consisted of feature extraction, the selection method, and a regressor for the final decision making. The stack of AEs was used to extract features from the processed dataset as described in Section 3.2.

In this study, DNN, ANN, KNN, SVM, LR, DT, ELM, and stacking were implemented using the sklearn machine learning package. However, we built the proposed MLELM from scratch because it was not available in the sklearn package. The MLELM contained two AEs with the ELM regressor. The ELM regressor was used for the final decision. Each AE had 100 hidden nodes, and the ELM regressor had 50 hidden nodes. Furthermore, in the development of the SVM, we utilized the standard parameters given in the sklearn package as follows: a 1.0 value of 'C', the radial basis function kernel, and a 0.1 value of epsilon. Similarly, for other comparison models, we used the standard parameters suggested in the sklearn machine learning package and [23].

4.5. Model Evaluation

In this research, we used the standard evaluation metrics commonly used in regression tasks to evaluate the performance of the models. The performance measures we employed were the root mean square error (RMSE), the mean absolute error (MAE), the mean square error (MSE), and the coefficient of determination (R^2-score). In addition, we present the training time of the compared algorithms. In [7,32] and the references therein, it shows that the most often used performance measures are the MSE and RMSE. For further comparsion, we compared the performance of the models using the R^2-score and the training time of the each model.

- The mean square error (MAE) is the absolute difference between the actual values and the predicted values at each point in a scatter plot. It is should be noted that the closer the MAE's value is to zero, the better the performance of the model. Furthermore, the higher the score, the worse the performance of the models. The average absolute difference between the actual and the predicted values is given by

$$MAE = \frac{1}{N} \sum_{k=1}^{N} |y_k - \hat{y}_k|, \tag{10}$$

  where y_k and $\hat{y}_k$ are the k-th value of the actual and the predicted values, respectively. N is the number of data points.

- The root mean square error (RMSE) is the root of the square of the average error between the actual values and the predicted values of the model. The RMSE is given by

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (y_k - \hat{y}_k)^2}, \tag{11}$$

  where N is the number of data points, and y_k and $\hat{y}_k$ are the k-th value of the actual and the predicted values, respectively.

- The mean square error (MSE) is the measure of the square error or variation between the actual values and the estimated values. It shows the quality of the prediction by the predictors (regressors). The closer the error is to zero, the better the performance of the model.

$$MSE = \frac{1}{N} \sum_{k=1}^{N} (y_k - \hat{y}_k)^2, \tag{12}$$

  where N is the number of data points, and y_k and $\hat{y}_k$ are the k-th value of the actual and the predicted values, respectively.

- The R square (R^2) is a metric commonly used to assess the performance of machine learning models in a prediction task. Basically, it determines how much of a difference in the target variable can be explained by the independent variables. The best outcome of the R^2 is 1.0, and the smaller the value of R^2 when compared to 1.0, the less desirable the result. Hence, for instance, an $R^2 = 0.65$ value is better than an $R^2 = 0.59$ value. It is also known as the coefficient of determination. The R square is calculated as follows:

$$R^2 = 1 - \frac{\sum_{k=1}^{N}(y_k - \hat{y}_k)^2}{\sum_{k=1}^{N}(y_k - \bar{y})^2},$$ (13)

where N is the number of data points, and y_k and $\hat{y}_k$ are the k-th value of the actual and the predicted values, respectively.

5. Result and Discussion

In this study, we validated the performance of the proposed MLELM using a large dataset. In addition, we compared the proposed MLELM with other state-of-the-art models commonly used for building energy consumption prediction. Eight other machine learning models were implemented and compared with the proposed MLELM. The comparison methods were support vector regression (SVR), decision tree (DT), linear regression (LR), K-nearest neighbor (KNN), deep neural network (DNN), multilayer neural network, ANN (MLP), stacking, extreme learning machine (ELM), and the proposed MLELM. In the experiment, for fair comparison, we utilized the popular cross-validation concept [50] to validate the performance of each method. The cross-validation method helps to evaluate the independence of the test dataset. Hence, it gives accurate details of the performances of each model.

In the the cross-validation concept, the tenfold evaluation method and fifteenfold evaluation method [50] were used. The cross-validation method is a standard method commonly used to evaluate the performance of a machine learning model on a test set.

Furthermore, we followed the standard sklearn package and the standard settings employed in [23]. For instance, for DNN, we utilized five layers, which included the input layer, the output layer, and three hidden layers, each containing 64 neurons. Other settings are available in [23].

5.1. Experiment 1: Tenfold Cross Validation

For the tenfold evaluation approach, the collected data were divided into 10 groups, and we ran the comparison algorithms 10 times. In the first run, we selected one group from the split data as a test set, while the rest of the data, that is, the remaining nine groups were used as the training set. In the second run, we chose another group as the test set, and the remaining nine groups were employed to train the models. This process continued until the tenth run was reached. The average of the evaluation metrics, namely the MSE, RMSE, MAE, and R square of each comparison model is reported in this study. Table 2 shows the detailed performance of the compared models for the tenfold evaluation method. The model that had the best performance is in bold as seen in Table 2. The proposed MLELM had the best performance in term of the MSE, MAE, RMSE, and R square. In other words, the average MSE, average MAE, average RMSE, and average R^2-score of the proposed MLELM model were lower than that of the compared machine learning models. Thus, the proposed MLELM was better able to predict the annual building energy consumption at the early stage of construction than the compared methods. As a result, it can lead to accurate energy policymaking. In addition, the standard deviation of the proposed method was much lower than that of the comparison methods. Furthermore, the R^2 of the proposed MLELM was 0.588990, and it was the highest value of all the models. Thus, the proposed MLELM outperformed the compared methods. This means that the proposed MLELM was more stable in terms of predicting the annual building energy consumption than other current machine learning models. In this way, construction costs can be managed, and

decisions can be made in a stable manner. For instance, in Table 2, from the MSE metric, the performance of MLELM had an average MSE of 0.0277, and it was small. Moreover, it had a standard deviation (STD) of 0.003463, while the values of the MSE of the compared methods were much higher than this with a larger standard deviation, as shown in Table 2. For instance, the average MSE of the DNN was 0.049172 with an STD of 0.008374, which was very large compared to the proposed MLELM.
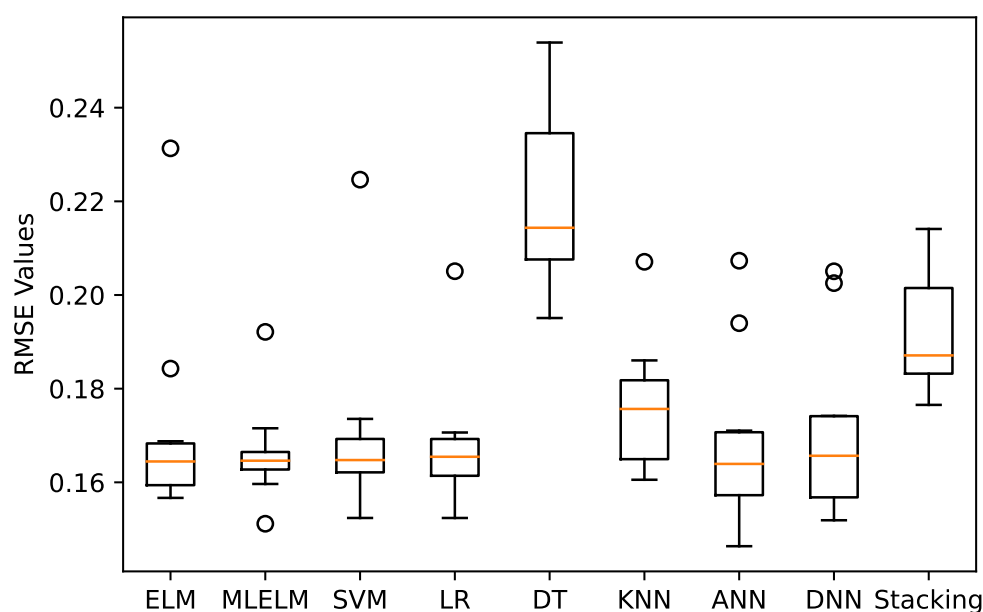
Table 2. The performance of various models for annual building energy consumption: tenfold division.

|	RMSE	STD-RMSE	MSE	STD-MSE	MAE	STD-MAE	R^2	STD-R^2
ELM	0.172858	0.008278	0.029880	0.008278	0.131924	0.016379	0.583827	**0.084258**
MLELM	**0.166541**	**0.003463**	**0.027736**	**0.003463**	**0.128263**	0.009046	**0.588990**	0.085919
SVM	0.171321	0.007268	0.029351	0.007268	0.130928	0.014499	0.581123	0.099753
LR	0.168418	0.004896	0.028364	0.004896	0.129586	0.009063	0.575575	0.120882
DT	0.221748	0.008374	0.049172	0.008374	0.168432	0.015450	0.468884	0.190318
KNN	0.176805	0.004831	0.031260	0.004831	0.134486	0.010274	0.512563	0.125415
ANN	0.169312	0.006397	0.028667	0.006397	0.130014	0.014149	0.571364	0.146048
DNN	0.171809	0.006493	0.029518	0.006493	0.129684	0.014702	0.553240	0.149189
Stacking	0.191935	0.004559	0.036839	0.004559	0.149320	**0.008277**	0.392181	0.108214

In addition, the value of the R square (R^2-score) of the MLELM was 0.588990, which was higher than the value of the R^2-score of the compared models. From the definition of R square in (Section 4.5), we concluded that the proposed MLELM had the best performance.

From the perspective of the interpretation of the results, they demonstrate that MLELM has the potential to accurately predict energy consumption during the early stages of construction. This information will help policymakers make more informed decisions. As a result, decisions can be made quickly and effectively, for instance, how much money the government could set aside for energy efficiency or upgrade projects. Using this accurate energy use prediction can also reduce energy taxation. Furthermore, the proposed model can be digitalized and commercialized, which means that software can be built around it. Therefore, the proposed model offers greater technological and economic benefits [51,52].
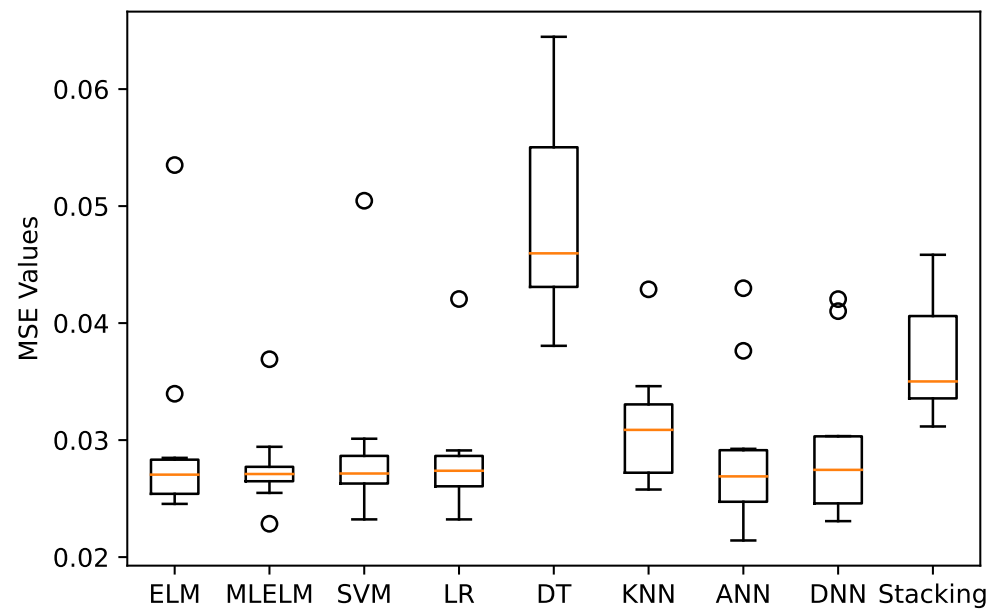
Furthermore, the boxplot method was employed to present the model's performances using the evaluation metrics as shown in Figures 7–10.

Figure 7. The performance of various models for annual building energy consumption: RMSE comparison.
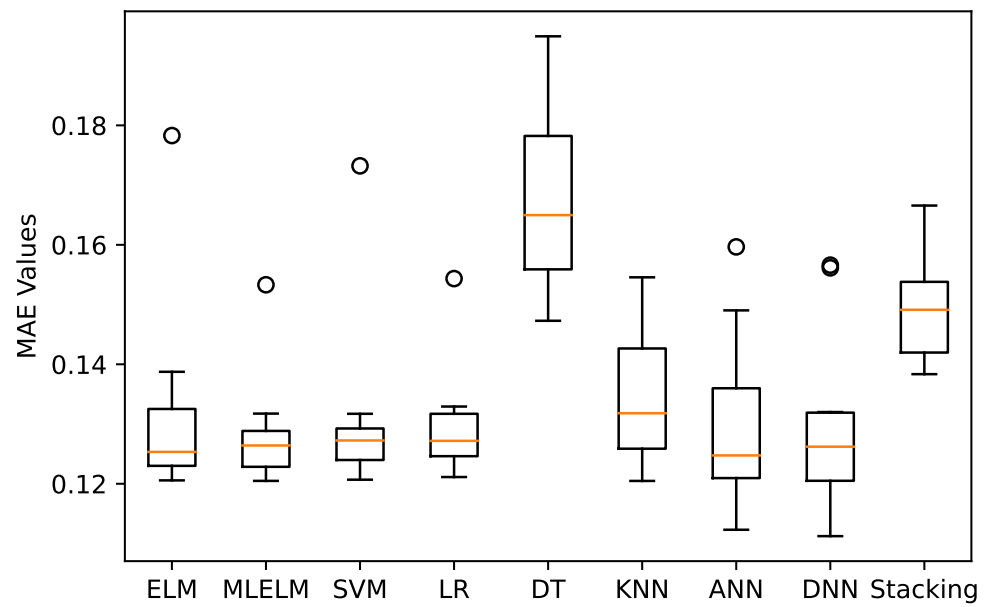
Figure 8. The performance of various models for annual building energy consumption: MAE comparison.
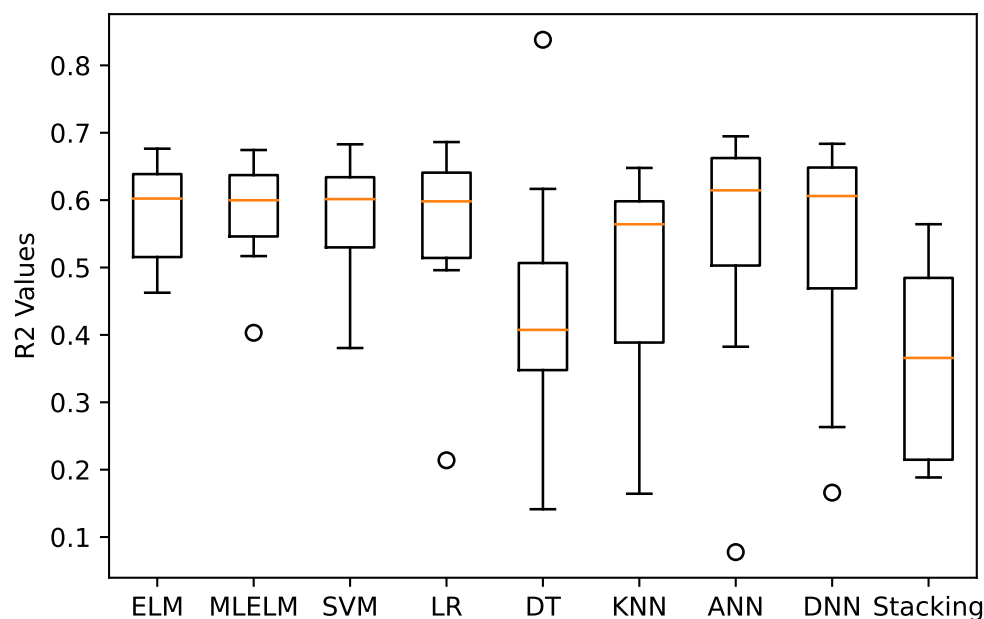

Figure 9. The performance of various models for annual building energy consumption: MSE comparison.

**Figure 10.** The performance of various models for annual building energy consumption: R^2 comparison.

Aside from the results in Table 2, a percentage improvement ratio (PIR) was performed to determine how much better the proposed MLELM was compared to the other models. The PIR was calculated, using the formula given by

$$\frac{P_c - P_p}{P_c} * 100\%, \tag{14}$$

where P_p and P_c are the performance of the proposed model and the performance of the compared model, respectively. The bar chart method was utilized to present the details of the PIR results. Figure 11 shows the PIR of the compared algorithms. The PIR shows that the MLELM had reasonable prediction performance in terms of the evaluation metrics, RMSE, MSE, MAE, and R^2-score. This supports using the MLELM model for annual building energy prediction at the early design phase of a building project.

From the figure, it is obvious that the DT had the worst performance in terms of the RMSE, MSE, and MAE. However, its performance was better than the stacking method in terms of the R^2-score, while the proposed MLELM had the best performance in all the cases considered. In addition, the MLELM outperformed the DNN, KNN, ANN, stacking, ELM, SVM, and LR. Based on the aforementioned results, the MLELM is suitable for the prediction of building energy consumption, especially at the early phase of construction. This will prevent or mitigate the construction of inefficient buildings with high energy use. Hence, it will minimize the cost of energy utilization and enhance decision making.
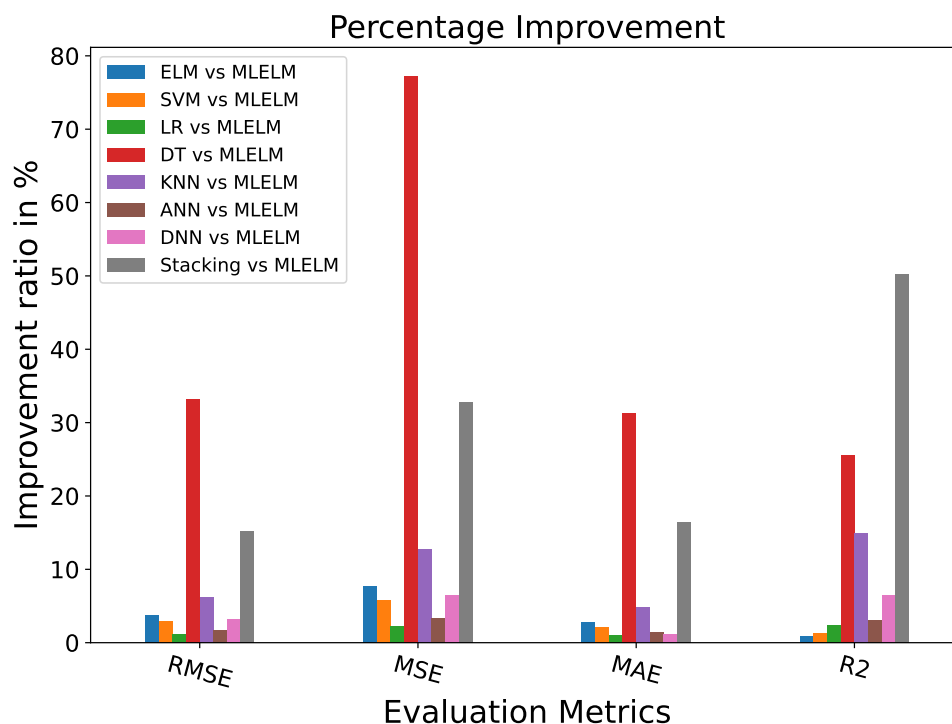
Figure 11. The Percentage Improvement Ratio (PIR) between the proposed MLELM and the compared methods.

5.2. Experiment 2: Fifteenfold Cross Validation

A further experiment was conducted to examine the performance of the models when more data segmentation was used. We used the fifteenfold cross-validation concept in this case. In other words, the dataset was divided into 15 groups, and we ran the algorithm 15 times. For each run, we took one group from the divided data as the test set and the remaining 14 groups as the training set. Hence, we trained the models with the data from the 14 groups and tested the models with a single group selected as the test set. In the second run, another group was taken as test set, and the remaining 14 groups were used as a training set. The process was repeated until the fifteenth run was reached. The average R^2-score, the average MSE, the average RMSE, and the average MAE are presented in Table 3. The model that had the best performance is in bold as seen in Table 3. As shown in the table, the proposed MLELM had a better result compared to the other methods. For instance, the MLELM had an average MSE, an average RMSE, and an average MAE of 0.026877, 0.163943, and 0.125931, respectively; whereas for the other methods, the average MSE, the average RMSE, and the average MAE were much higher. In the ANN method, we notice that the average MAE was 0.124810; however, it had a poor standard deviation (STD). The value of the STD of the ANN method was 0.010351. However, the value of STD of the proposed MLELM under the MAE metric was 0.007783, and it was very small compared to the ANN. This supports the conclusion that the proposed MLELM is more stable. With this stability, the proposed model will lead to stable decision making as well as facilitate design management prior to construction.

In addition, the proposed MLELM had the highest R^2-score and was better than its counterparts. The model that had the best performance is highlighted in bold in Table 3.

Furthermore, the average training time for the methods is presented in Table 4. From the table, the training time of the proposed MLELM was reasonable when compared with the commonly used machine learning methods, such as SVM, ANN, DNN, and stacking, for energy prediction. Hence, the proposed method can be used in production for the prediction of annual building energy consumption.

Table 3. The performance of various models for annual building energy consumption: fifteenfold division.

	RMSE	STD-RMSE	MSE	STD-MSE	MAE	STD-MAE	R^2	STD-R^2
ELM	0.165978	0.004063	0.027549	0.004063	0.127383	0.009553	0.591532	0.110926
MLELM	**0.163943**	**0.003467**	**0.026877**	**0.003467**	0.125931	**0.007783**	**0.604863**	0.098670
SVM	0.166296	0.003755	0.027654	0.003755	0.127400	0.008058	0.589093	0.108849
LR	0.165331	0.003584	0.027334	0.003584	0.127618	0.007845	0.595425	0.104244
DT	0.213160	0.007786	0.045437	0.007786	0.159670	0.015534	0.412233	0.167594
KNN	0.171204	0.004915	0.029311	0.004915	0.130788	0.011348	0.556586	0.121092
ANN	0.164569	0.004921	0.027083	0.004921	**0.124810**	0.010351	0.600091	0.133008
DNN	0.166838	0.003994	0.027835	0.003994	0.128700	0.010288	0.586552	0.105408
Stacking	0.185677	0.004164	0.034476	0.004164	0.144598	0.009565	0.435754	0.097321

Table 4. The average training time in seconds (s) for each method.

	ELM	MLELM	SVM	LR	DT	KNN	ANN	DNN	Stacking
Training Time (s)	0.174543	1.209907	7.169429	0.026579	0.335507	0.221448	3.642533	34.993043	5.044873

With the results presented in this subsection (Section 5.2) and subsection (Section 5.1), e.g., Tables 2 and 3, in terms of the MSE, RMSE, MAE, and R^2, we validated that the proposed MLELM method for the prediction of annual building energy consumption led to a better, more efficient, and more economic determination of building energy consumption compared to other methods. From an economic point of view [51], our method can optimize building construction time while building efficient buildings. Hence, it can save costs and time and enhance policy management. Furthermore, it can aid decision making on building construction. It can be said that, from the aforementioned explanations, the proposed method leads to a better techno-economic assessment [53,54], as it can lead to the construction of innovative designs, which can also encourage the design of new building materials, thus leading to the creation of new jobs and reducing the cost of energy consumption. The proposed method can prevent the unnecessary waste of energy. This can provide a comfortable, safe, and attractive living and work environment. In the long-term, having efficient buildings can lead to economic growth. The cost saved from constructing efficient buildings can be diverted to other avenues.

6. Conclusions

This paper proposed a multilayer extreme learning machine (ML-ELM) based on big data to predict annual building energy consumption at an early stage of construction. The paper also implemented eight state-of-the-art models. The implemented models were the ELM, SVM, LR, DT, KNN, ANN, DNN, stacking, and the proposed MLELM. The performances of these models were compared. In general, the MLELM model produced better results than the other models. Hence, with the good performance results generated by this study, building designers can predict energy consumption early in the design process of a building using the proposed MLELM. Thus, a better policy on building construction and cost savings can be proposed by policymakers.

The compared models were evaluated using various metrics, such as the MSE, MAE, RMSE, and R^2-score. In addition, we presented the training time of each model, which showed the computation time of each model. The fast training time of the proposed MLELM means a fast and accurate decision can be achieved by policymakers before construction.

In addition, the percentage improvement ratio (PIR) was calculated to determine the effectiveness of the proposed model in predicting the building energy consumption compared to other models.

This percentage improvement ratio (PIR) provided motivation for builders to employ the proposed MLELM model at an early stage of a building's construction to make a good decision that would lead to efficient building design and good construction management. In summary, the proposed model is of practical importance for the timely and accurate prediction of building energy consumption.

In addition to the aforementioned points, future studies should focus on employing a deeper stack of autoencoders (AEs) for a deeper MLELM model on a larger dataset of more than 60,000 for possible better performance. In addition, other variants of AEs can be explored for building energy prediction. As a result of the usefulness of open data regarding technologies, future studies can explore the use of some available open data [55,56]. Moreover, further optimization of the parameters for the proposed model can be explored using evolutionary optimization [57] or Bayesian optimization techniques [58,59].

Author Contributions: Conceptualization, M.A.; Methodology, M.A.; Writing—original draft, M.A.; Writing—review & editing, A.H. and S.A.; Visualization, R.O.-A.; Project administration, S.A.; Funding acquisition, A.H. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The data presented in this study are available on request.

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

Abbreviations

The following abbreviations are used in this manuscript:

ANN	Artificial Neural Network
MLP	Multilayer Perceptron
ELM	Extreme Learning Machine
MLELM	Multilayer Extreme Learning Machine
SVM	Support Vector Machine
DT	Decision Tree
RF	Random Forest
KNN	K-Nearest Neighbor
DNN	Deep Neural Network
LR	Linear Regression
AE	AutoEncoder
AEs	AutoEncoders
EDA	Exploration Data Analysis
PIR	Percentage Improvement Ratio
MAE	Mean Average Error
MSE	Mean Square Error
RMSE	Root Mean Square Error
R^2	Coefficient of Determination
STD	Standard Deviation
$\bar{y}$	Mean of the output
N	Number of data points

References

1. Amasyali, K.; El-Gohary, N.M. A review of data-driven building energy consumption prediction studies. *Renew. Sustain. Energy Rev.* **2018**, *81*, 1192–1205. [CrossRef]
2. Allouhi, A.; El Fouih, Y.; Kousksou, T.; Jamil, A.; Zeraouli, Y.; Mourad, Y. Energy consumption and efficiency in buildings: Current status and future trends. *J. Clean. Prod.* **2015**, *109*, 118–130. [CrossRef]
3. Guo, Y.; Wang, J.; Chen, H.; Li, G.; Liu, J.; Xu, C.; Huang, R.; Huang, Y. Machine learning-based thermal response time ahead energy demand prediction for building heating systems. *Appl. Energy* **2018**, *221*, 16–27. [CrossRef]
4. Killian, M.; Kozek, M. Ten questions concerning model predictive control for energy efficient buildings. *Build. Environ.* **2016**, *105*, 403–412. [CrossRef]
5. Serale, G.; Fiorentini, M.; Capozzoli, A.; Bernardini, D.; Bemporad, A. Model predictive control (MPC) for enhancing building and HVAC system energy efficiency: Problem formulation, applications and opportunities. *Energies* **2018**, *11*, 631. [CrossRef]
6. Pham, A.D.; Ngo, N.T.; Truong, T.T.H.; Huynh, N.T.; Truong, N.S. Predicting energy consumption in multiple buildings using machine learning for improving energy efficiency and sustainability. *J. Clean. Prod.* **2020**, *260*, 121082. [CrossRef]
7. Dong, B.; Cao, C.; Lee, S.E. Applying support vector machines to predict building energy consumption in tropical region. *Energy Build.* **2005**, *37*, 545–553. [CrossRef]

8. Paudel, S.; Elmitri, M.; Couturier, S.; Nguyen, P.H.; Kamphuis, R.; Lacarrière, B.; Le Corre, O. A relevant data selection method for energy consumption prediction of low energy building based on support vector machine. *Energy Build.* **2017**, *138*, 240–256. [CrossRef]

9. Wu, X.; Kumar, V.; Ross Quinlan, J.; Ghosh, J.; Yang, Q.; Motoda, H.; McLachlan, G.J.; Ng, A.; Liu, B.; Yu, P.S.; et al. Top 10 algorithms in data mining. *Knowl. Inf. Syst.* **2008**, *14*, 1–37. [CrossRef]

10. Ramos, D.; Faria, P.; Morais, A.; Vale, Z. Using decision tree to select forecasting algorithms in distinct electricity consumption context of an office building. *Energy Rep.* **2022**, *8*, 417–422. [CrossRef]

11. Shcherbakov, M.; Kamaev, V.; Shcherbakova, N. Automated electric energy consumption forecasting system based on decision tree approach. *IFAC Proc. Vol.* **2013**, *46*, 1027–1032. [CrossRef]

12. Wang, Z.; Wang, Y.; Zeng, R.; Srinivasan, R.S.; Ahrentzen, S. Random Forest based hourly building energy prediction. *Energy Build.* **2018**, *171*, 11–25. [CrossRef]

13. Cáceres, L.; Merino, J.I.; Díaz-Díaz, N. A computational intelligence approach to predict energy demand using random forest in a cloudera cluster. *Appl. Sci.* **2021**, *11*, 8635. [CrossRef]

14. Chen, Y.T.; Piedad, E., Jr.; Kuo, C.C. Energy consumption load forecasting using a level-based random forest classifier. *Symmetry* **2019**, *11*, 956. [CrossRef]

15. Ahmad, A.S.; Hassan, M.Y.; Abdullah, M.P.; Rahman, H.A.; Hussin, F.; Abdullah, H.; Saidur, R. A review on applications of ANN and SVM for building electrical energy consumption forecasting. *Renew. Sustain. Energy Rev.* **2014**, *33*, 102–109. [CrossRef]

16. Li, Z.; Dai, J.; Chen, H.; Lin, B. August. An ANN-based fast building energy consumption prediction method for complex architectural form at the early design stage. In *Building Simulation*; Tsinghua University Press: Beijing, China, 2019; Volume 12, pp. 665–681.

17. Runge, J.; Zmeureanu, R. Forecasting energy use in buildings using artificial neural networks: A review. *Energies* **2019**, *12*, 3254. [CrossRef]

18. Li, K.; Hu, C.; Liu, G.; Xue, W. Building's electricity consumption prediction using optimized artificial neural networks and principal component analysis. *Energy Build.* **2015**, *108*, 106–113. [CrossRef]

19. Fu, X.; Zhou, Y.; Yang, F.; Ma, L.; Long, H.; Zhong, Y.; Ni, P. A review of key technologies and trends in the development of integrated heating and power systems in agriculture. *Entropy* **2021**, *23*, 260. [CrossRef]

20. Xie, Q.; Ni, J.Q.; Bao, J.; Su, Z. A thermal environmental model for indoor air temperature prediction and energy consumption in pig building. *Build. Environ.* **2019**, *161*, 106238. [CrossRef]

21. Jallal, M.A.; González-Vidal, A.; Skarmeta, A.F.; Chabaa, S.; Zeroual, A. A hybrid neuro-fuzzy inference system-based algorithm for time series forecasting applied to energy consumption prediction. *Appl. Energy* **2020**, *268*, 114977. [CrossRef]

22. McNeil, M.A.; Karali, N.; Letschert, V. Forecasting Indonesia's electricity load through 2030 and peak demand reductions from appliance and lighting efficiency. *Energy Sustain. Dev.* **2019**, *49*, 65–77. [CrossRef]

23. Olu-Ajayi, R.; Alaka, H.; Sulaimon, I.; Sunmola, F.; Ajayi, S. Building energy consumption prediction for residential buildings using deep learning and other machine learning techniques. *J. Build. Eng.* **2022**, *45*, 103406. [CrossRef]

24. Jassar, S.; Liao, Z.; Zhao, L. Impact of data quality on predictive accuracy of ANFIS based soft sensor models. In Proceedings of the World Congress on Engineering and Computer Science, San Francisco, CA, USA, 20–22 October 2009; Volume 2, pp. 1001–1006.

25. Demsar, J. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **2006**, *7*, 1–30.

26. Huang, G.; Huang, G.B.; Song, S.; You, K. Trends in extreme learning machines: A review. *Neural Netw.* **2015**, *61*, 32–48. [CrossRef] [PubMed]

27. Amasyali, K.; El-Gohary, N. Deep learning for building energy consumption prediction. In Proceedings of the 6th CSCE-CRC International Construction Specialty Conference 2017-Held as Part of the Canadian Society for Civil Engineering Annual Conference and General Meeting 2017, Vancouver, BC, Canada, 31 May–3 June 2017; Canadian Society for Civil Engineering: Montreal, QC, Canada, 2017; pp. 466–474.

28. Shao, M.; Wang, X.; Bu, Z.; Chen, X.; Wang, Y. Prediction of energy consumption in hotel buildings via support vector machines. *Sustain. Cities Soc.* **2020**, *57*, 102128. [CrossRef]

29. Zhao, H.X.; Magoulès, F. A review on the prediction of building energy consumption. *Renew. Sustain. Energy Rev.* **2012**, *16*, 3586–3592. [CrossRef]

30. Dong, Z.; Liu, J.; Liu, B.; Li, K.; Li, X. Hourly energy consumption prediction of an office building based on ensemble learning and energy consumption pattern classification. *Energy Build.* **2021**, *241*, 110929. [CrossRef]

31. Li, Q.; Meng, Q.; Cai, J.; Yoshino, H.; Mochida, A. Applying support vector machine to predict hourly cooling load in the building. *Appl. Energy* **2009**, *86*, 2249–2256. [CrossRef]

32. Li, Q.; Meng, Q.; Cai, J.; Yoshino, H.; Mochida, A. Predicting hourly cooling load in the building: A comparison of support vector machine and different artificial neural networks. *Energy Convers. Manag.* **2009**, *50*, 90–96. [CrossRef]

33. Wang, R.; Lu, S.; Feng, W. A novel improved model for building energy consumption prediction based on model integration. *Appl. Energy* **2020**, *262*, 114561. [CrossRef]

34. Chou, J.S.; Bui, D.K. Modeling heating and cooling loads by artificial intelligence for energy-efficient building design. *Energy Build.* **2014**, *82*, 437–446. [CrossRef]

35. Li, X.; Wen, J. Review of building energy modeling for control and operation. *Renew. Sustain. Energy Rev.* **2014**, *37*, 517–537. [CrossRef]

36. Abd Elaziz, M.; Thanikanti, S.B.; Ibrahim, I.A.; Lu, S.; Nastasi, B.; Alotaibi, M.A.; Hossain, M.A.; Yousri, D. Enhanced marine predators algorithm for identifying static and dynamic photovoltaic models parameters. *Energy Convers. Manag.* **2021**, *236*, 113971. [CrossRef]
37. Singh, R.; Balasundaram, S. Application of extreme learning machine method for time series analysis. *Int. J. Comput. Inf. Eng.* **2007**, *1*, 3407–3413.
38. Şahin, M.; Kaya, Y.; Uyar, M.; Yıldırım, S. Application of extreme learning machine for estimating solar radiation from satellite data. *Int. J. Energy Res.* **2014**, *38*, 205–212. [CrossRef]
39. Yahia, S.; Said, S.; Zaied, M. Wavelet extreme learning machine and deep learning for data classification. *Neurocomputing* **2022**, *470*, 280–289. [CrossRef]
40. Liu, X.; Xu, L. The universal consistency of extreme learning machine. *Neurocomputing* **2018**, *311*, 176–182. [CrossRef]
41. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [CrossRef]
42. Sun, K.; Zhang, J.; Zhang, C.; Hu, J. Generalized extreme learning machine autoencoder and a new deep neural network. *Neurocomputing* **2017**, *230*, 374–381. [CrossRef]
43. Katuwal, R.; Suganthan, P.N. Stacked autoencoder based deep random vector functional link neural network for classification. *Appl. Soft Comput.* **2019**, *85*, 105854. [CrossRef]
44. Hu, X.; Xiao, Z.; Liu, D.; Tang, Y.; Malik, O.P.; Xia, X. KPCA and AE based local-global feature extraction method for vibration signals of rotating machinery. *Math. Probl. Eng.* **2020**, *2020*, 5804509. [CrossRef]
45. Li, R.; Wang, X.; Lei, L.; Wu, C. Representation learning by hierarchical ELM auto-encoder with double random hidden layers. *IET Comput. Vis.* **2019**, *13*, 411–419. [CrossRef]
46. Bourdeau, M.; Zhai, X.Q.; Nefzaoui, E.; Guo, X.; Chatellier, P. Modeling and forecasting building energy consumption: A review of data-driven techniques. *Sustain. Cities Soc.* **2019**, *48*, 101533. [CrossRef]
47. Lin, W.C.; Tsai, C.F. Missing value imputation: A review and analysis of the literature (2006–2017). *Artif. Intell. Rev.* **2020**, *53*, 1487–1509. [CrossRef]
48. Hao, J.; Ho, T.K. Machine learning made easy: A review of scikit-learn package in python programming language. *J. Educ. Behav. Stat.* **2019**, *44*, 348–361. [CrossRef]
49. Komer, B.; Bergstra, J.; Eliasmith, C. Hyperopt-sklearn. In *Automated Machine Learning*; Springer: Cham, Switzerland, 2019; pp. 97–111.
50. Refaeilzadeh, P.; Tang, L.; Liu, H. Cross-validation. *Encycl. Database Syst.* **2009**, *5*, 532–538.
51. Tronchin, L.; Manfren, M.; Nastasi, B. Energy efficiency, demand side management and energy storage technologies—A critical analysis of possible paths of integration in the built environment. *Renew. Sustain. Energy Rev.* **2018**, *95*, 341–353. [CrossRef]
52. Manfren, M.; Nastasi, B.; Tronchin, L.; Groppi, D.; Garcia, D.A. Techno-economic analysis and energy modelling as a key enablers for smart energy services and technologies in buildings. *Renew. Sustain. Energy Rev.* **2021**, *150*, 111490. [CrossRef]
53. Maltese, S.; Tagliabue, L.C.; Cecconi, F.R.; Pasini, D.; Manfren, M.; Ciribini, A.L. Sustainability assessment through green BIM for environmental, social and economic efficiency. *Procedia Eng.* **2017**, *180*, 520–530. [CrossRef]
54. Bora, R.R.; Lei, M.; Tester, J.W.; Lehmann, J.; You, F. Life cycle assessment and technoeconomic analysis of thermochemical conversion technologies applied to poultry litter with energy and nutrient recovery. *ACS Sustain. Chem. Eng.* **2020**, *8*, 8436–8447. [CrossRef]
55. Available online: https://datamillnorth.org/dataset (accessed on 5 December 2022).
56. ECUK. Available online: https://www.data.gov.uk/dataset/26afb14b-be9a-4722-916e-10655d0edc38/energy-consumption-in-the-uk (accessed on 6 December 2022).
57. Poli, R.; Kennedy, J.; Blackwell, T. Particle swarm optimization. *Swarm Intell.* **2007**, *1*, 33–57. [CrossRef]
58. Zhang, Y.M.; Wang, H.; Mao, J.X.; Xu, Z.D.; Zhang, Y.F. Probabilistic framework with bayesian optimization for predicting typhoon-induced dynamic responses of a long-span bridge. *J. Struct. Eng.* **2021**, *147*, 04020297. [CrossRef]
59. Liang, X. Image-based post-disaster inspection of reinforced concrete bridge systems using deep learning with Bayesian optimization. *Comput.-Aided Civ. Infrastruct. Eng.* **2019**, *34*, 415–430. [CrossRef]