Article

Towards Efficient Building Designing: Heating and Cooling Load Prediction via Multi-Output Model

**Muhammad Sajjad [1], Samee Ullah Khan [2], Noman Khan [2], Ijaz Ul Haq [2] , Amin Ullah [2] ,
Mi Young Lee [2] and Sung Wook Baik [2,***

[1]   Digital Image Processing Laboratory, Islamia College Peshawar, Peshawar 25120, Pakistan;
       Muhammad.sajjad@icp.edu.pk

[2]   Sejong University, Seoul 143-747, Korea; samee@sju.ac.kr (S.U.K.); noman0869@sju.ac.kr (N.K.);
       ijazulhaq@sju.ac.kr (I.U.H.); qamin3797@gmail.com (A.U.); miylee@sejong.ac.kr (M.Y.L.)

*   Correspondence: sbaik@sejong.ac.kr

check for
updates

Abstract: In the current technological era, energy-efficient buildings have a significant research body
due to increasing concerns about energy consumption and its environmental impact. Designing an
appropriate energy-efficient building depends on its layout, such as relative compactness, overall area,
height, orientation, and distribution of the glazing area. These factors directly influence the cooling
load (CL) and heating load (HL) of residential buildings. An accurate prediction of these load facilitates
a better management of energy consumption and enhances the living standards of inhabitants. Most of
the traditional machine learning (ML)-based approaches are designed for single-output (SO) prediction,
which is a tedious task due to separate training processes for each output with low performance.
In addition, these approaches have a high level of nonlinearity between input and output, which need
more enhancement in terms of robustness, predictability, and generalization. To tackle these issues,
we propose a novel framework based on gated recurrent unit (GRU) that reliably predicts the CL
and HL concurrently. To the best of our knowledge, we are the first to propose a multi-output
(MO) sequential learning model followed by utility preprocessing under the umbrella of a unified
framework. A comprehensive set of ablation studies on ML and deep learning (DL) techniques is
done over an energy efficiency dataset, where the proposed model reveals an incredible performance
as compared to other existing models.

Keywords: cooling load; energy consumption; energy efficient building; GRU; heating load

1. Introduction to Residential Building Energy

Due to the rapid growth in economy and population during last few decades, the consumption
of electrical energy has been rapidly increasing day by day [1]. In 2018, the International Energy
Agency (IEA) reported that most of the electrical energy is spent in residential buildings, and the
demand for energy is rising every year due to excess usage of energy appliances as shown in Figure 1.
Existing studies reveal that residential buildings consumed more energy in summer and winter seasons,
which totally depends on the building architecture and occupied area [2].

Internal and external environment temperatures also have an effect on the total energy consumption
in a building [3]. Therefore, precise prediction of HL and CL is important in order to provide a luxurious
life for occupants [4,5]. HL is described as the total amount of required heat energy to keep the room
temperature normal, while CL is the sum of thermal energy necessary to be eliminated from a cooling
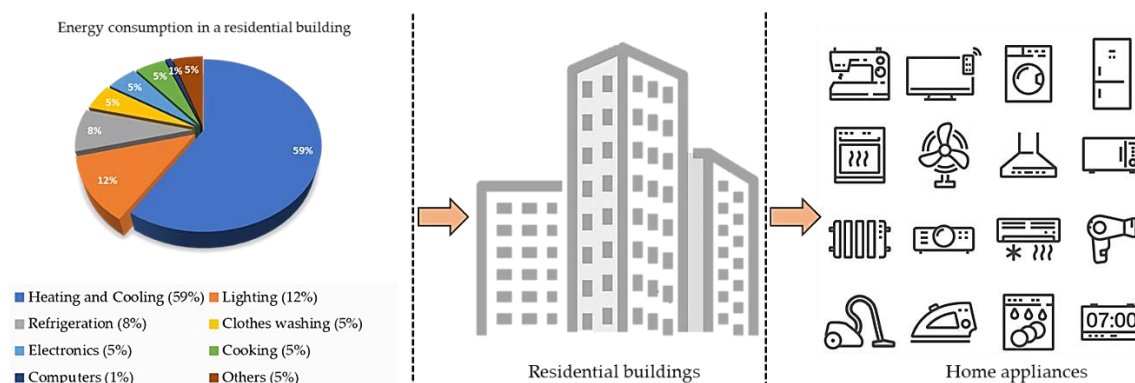area in order to keep the temperature at an appropriate level [2].

Figure 1. Massive amount of energy is consumed in the residential sector because various electrical appliances are installed.

Before evaluating the thermal load, it is important to know the infrastructure of buildings, because energy consumption is reliant on their physical attributes. Basically, four tools are employed to predict the CL and HL of buildings: simulation modeling, engineering calculations, statistical models, and ML models [6]. The simulation model is commonly used to simulate energy efficiency based on prior information, but it is a very difficult and time-consuming model because it requires more skill to operate. For instance, Bagheri et al. [7] considered the simulation methods in terms of its applications and limitations in the domain of energy performance. The second tool utilized complicated mathematical formulas according to its principles to efficiently predict energy load. Next, a statistical tool is used to evaluate linear regression models for residential energy consumption prediction, and later, the performance of the model is enhanced by modifying different parameters. The final tool is ML, which is a subset of statistical techniques, but it has the potential to learn from real data and predict the desired outputs. Further, it assists civil engineers in evaluating the ingredients used in the building design. For instance, support vector regression (SVR), clustering, and Gaussian-based regression are active ML approaches in energy predictions [6].

ML algorithms can be broadly categorized into two main groups (i.e., supervised and unsupervised) based on diverse learning style. The predicted output variables are available in the case of supervised learning, while the unique labeled output does not occur in an unsupervised learning strategy. The current study focuses on a supervised learning approach because an energy efficiency dataset has labeled data. Artificial neural networks (ANNs) have gained attention among supervised learning techniques due to nonlinear relationships within the data. Moreover, the activation function of ANN can predicted the desired outputs, which indicate the nonlinearity with various input attributes [6]. Numerous ANN architectures, including recurrent networks, radial basis function, and feedforward, are used for energy prediction. Besides ANN, researchers have mostly implemented the multilayer perceptron (MLP) model, where information flows in a single direction with multiple layers. The MLP model comprises three basic layers (input, hidden, output) consisting of neurons with weighted functions. In case of complex data processing, the existing model is altered by increasing the number of neurons and hidden layers.

Managing huge and complicated energy consumption data is formidable for ANN, while researchers have criticized this network due to low transparency in the model [8]. Sensitivity analysis (SA) is broadly applied to analyze the relationship between variables. For precise energy forecasting, ANN shows better performance if irrelevant inputs are removed [9]. Sensitivity analysis about the mean (SAAM) is one of the conventional strategies, where changes of dependent variables are recorded while independent variables are kept in a specified range by computing the mean [10]. The key benefits of SAAM are simple interpretation, easy implementation, and application, along with statistical analysis [11]. In addition, state-based sensitivity analysis is a global SA method in which separate variables are varied independently and the rest of the variables are changed concurrently to obtain the reliant attributes [8].

In residential buildings, there are various factors that influence energy consumption, such as consumer's behavior and building architecture. Therefore, building-structure-related data play a key role in developing an efficient energy model. Moreover, the height of buildings, construction materials, and areas such as wall, roof, and glazing are the main attributes in the current research. The simulated method in [12] performs a pivotal part in improving building constructions, and it can also accurately depict real assessments of different building designs to predict HL and CL [13]. On the other hand, most of the researchers get full advantages by applying DL models on different domains, such as movie and video summarization [14,15], energy forecasting [16], biological data analysis [17], violence detection [18,19], and action recognition [20]. In this study, we explored numerous ML and DL models for the prediction of HL and CL using an energy efficiency dataset. The potential of sequential models for this dataset has not been thoroughly explored till date. Therefore, GRU has an optimal preference to predict HL and CL as there exists an intensely independent relationship between data. We conduct two types of experiments. First, we enhance the existing performance in which HL and CL are predicted separately. Second, a multi-output prediction is performed through the same architecture. The relevance of this work can enable engineers to solve major structural issues when designing an energy-efficient building.

There is no existing work that utilized GRU for this dataset till date. Therefore, in the current study, we utilize the sequence learning model GRU for non-sequential data by examining various parameters. The second limitation is the unavailability of preprocessing methods, including polynomial and min–max normalization for HL and CL. In this study, the simulation data first pass through a preprocessing step where outliers are removed, scattered data is normalized in a specific range, and increase the number of features. Next, the refined data are fed into the GRU network to extract silent hidden patterns. Finally, we evaluate the error in different metrics, such as mean absolute error (MAE), relative mean absolute error (rMAE), mean square error (MSE), relative mean square error (rMSE), root mean square error (RMSE), relative root mean square error (rRMSE), mean average percentage error (MAPE), and relative mean average percentage error (rMAPE). The major contributions of this study are summarized below:

- It is a common fact that the performance of a deep model is directly depends on the input data. In this study, energy efficiency dataset is used that contains a limited number of attributes with values in a different range, which cause overfitting and take extra time to converge. To address these issues, first, we pass the input data through a preprocessing layer where the number of features increased using a polynomial equation and min–max normalization process is applied to remove outliers and normalize the data in a particular range.

- Existing models in the literature are trained separately for HL and CL prediction, which requires a tedious and time-consuming job. In contrast, the proposed framework has a generalized ability in which the same architecture can be used for both SO and MO that predict HL and CL concurrently.

- DL models always reveal a convincing performance compared with traditional ML models. Therefore, we propose a sequence learning model GRU, which learns discriminative features and efficiently predicts the HL and CL. We also conduct a comparative study between ML and DL techniques to show the superiority of DL models.

- We verify experimentally that the proposed framework outperforms state-of-the-art techniques using the hold-out and 10-fold methods. To check the effectiveness of the proposed framework, we evaluateit on various metrics, such as MAE, rMAE, MSE, rMSE, RMSE, and rRMSE.

The rest of the paper is categorized into four main sections. Section 2 briefly discusses the literature study about HL and CL prediction. Section 3 explains the proposed methodology, followed by comprehensive experiments in Section 4. Section 5 concludes this study with future research direction.

2. Literature Review of HL and CL Prediction

The literature study for HL and CL prediction in buildings is mainly divided into four major classes: residential, educational, commercial, and mixed. According to statistics in [21], 30% of the literature is based on residential building energy. Through Ecotect software, Tsanas and Xifara [12] simulated 12 distinct building structures to predict HL and CL. After considering all the various permutations of input variables, 768 building designs were generated. During the simulation of building designs, heating, ventilation, and air-conditioning HVAC rules were pursued. Through numerous ML techniques, various researchers analyze these data for precise prediction. Based on the prominent contribution of Tsanas and Xifara [12], the existing literature is summarized in Table 1. Although the dataset has been prepared via a simulated tool, but there is lack of data related to building infrastructure and materials. The dataset used in this study is publicly accessible and extensively used for research study by exploring its applications related to energy. Simulated data play a significant role when designing the architectures of a building. The terms used in the existing studies are listed in Table 1.

Table 1. Existing approaches for the prediction of HL and CL using energy efficiency dataset.

Reference	Learning Strategy	Feature Selection	Evaluation Metrics
Tsanas and Xifara [12]	RF, iteratively reweighted last squares (IRLS)	Mutual information, Spearman rank correlation coefficient, and *p*-value	MSE, MAE, MRE
Chou and Bui [22]	Fusion method (SVR + ANN), RF, SVR, CART, GLR, CHAID	-	RMSE, MAE, MAPE, R, SI
Cheng and Cao [23]	Evolutionary multivariate adaptive regression splines (EMARS)	MARS	RMSE, MAPE, MAE, R^2
Ahmed et al. [24]	ANN and *k*-means cluster	-	Silhouette score
Sonmez et al. [25]	KNN and ANN (ABC, GA)	-	MAE, standard deviation
Alam et al. [26]	ANN	ANOVA	RMSE
Fei et al. [27]	ANN	-	MSE
Regina and Capriles [28]	DT, MLP, RF, SVR	-	MAE, RMSE, MRE, R^2
Naji et al. [29]	ANFIS	-	RMSE, R, R^2
Naji et al. [30]	ELM	-	RMSE, R, R^2
Nilashi et al. [31]	EM and ANFIS	PCA	MAE, MAPE, RMSE
Nwulu [32]	ANN	-	RMSE, RRSE, MAE, RAE, R^2
Duarte et al. [33]	DT, MLP, RF, SVM	-	MAE, RMSE, MAPE, R^2
Roy et al. [2]	Multivariate adaptive regression splines, ELM, a hybrid model of MARS and ELM	MARS	RMSE, MAPE, MAE, R^2, WMAPE, Time
Kavaklioglu [34]	OLS, PLS	-	RMSE, R^2,
Kumar et al. [35]	ELM, online sequential ELM, bidirectional ELM	-	MAE, RMSE
Al-Rakhami et al. [36]	Ensemble learning applying XGBoost	-	RMSE, R^2, MAE, MAPE
Sekhar et al. [4]	DNN, GRP, MPMR	-	VAF, RAAE, RMAE, R^2, MAPE, NS, RMSE, WMAPE

Tsanas and Xifara [12] conducted a detailed statistical study of density and scatterplots. The performance outcomes of the statistical analysis approach are mainly used for nonlinear problems. From Table 1, it can be observed that few studies have applied ANN on energy efficiency dataset [24,32], although others follow the ensemble strategy by integrating different methods [22,25,29,31]. To the best of our knowledge, only one article exists that applied a deep neural network (DNN) to predict HL and CL, presented by Sekha et al. [4]. The efficiency of DNN is better as compared with other traditional algorithms, such as Minimax Probability Machine Regression (MPMR) and Gaussian Process Regression (GPR). Moreover, the traditional approaches did not mention the model parameters,

such as processing elements, activation functions, and numbers of layers. To achieve a remarkable performance on any models, analysis of data is essential to identify the significant and insignificant inputs. In this regard, Roy et al. [2] proposed a nonparametric regression model known as Multivariate Adaptive Regression Splines (MARS) that splits the data and fit each interval into a basis function. Principal component analysis (PCA) is also applied for ideal features selection and dimensionality reduction, which eradicates the multilinearity problem. Nilashi et al. [31] reported that PCA targets four main aspects: retrieving essential information, reducing the dimension of data, simplifying the information, and analyzing architecture-related observations. Most of the articles did not utilize the SA approach for the prediction of HL and CL as shown in Table 1.

The techniques for quantitative SA are classified into local and global [37]. For instance, the input instances were impartial with each other; therefore, Ardjmand et al. [8] defined a regression-based strategy in which conventional SA techniques, such as sampling, regression-based, and variance-based, are expanded to state-based sensitivity analysis (SBSA). This means that modifying one variable value would influence the others; therefore, it is not realistic for fixed values for certain inputs in local SAs. On the other hand, the global SA adjusts the ideal input value, while in the case of multidimensionality, it takes the average number of variable inputs [38].

The efficiency of a mathematical model is also influenced by many assumptions in order to predict energy HL and CL separately. In majority of the works, the HL and CL are predicted in an SO fashion; however, we develop such a model that can be utilized for both SO and MO. Another primary consideration for enhancing the efficiency of a predictive model is preprocessing of data. Therefore, Kumar et al. [35] followed the ensemble technique with a proper attribute selection and preprocessing method to efficiently predict energy in real time. To boost the model efficiency, it is necessary to pass the data through the preprocessing stage. Notably, MSE, MAE, RMSE, and MAPE were common evaluation metrics used by researchers for model assessment, but in this research, we also use extra metrics for evaluation, including rMAE, rMSE, rRMSE, rMAPE.

3. Methodology for HL and CL Prediction

The primary goal of the proposed framework is to efficiently predict the HL and CL that will assist engineers in building energy-efficient buildings. A comprehensive set of experiments are conducted on ML and DL using hold-out and cross-validation methods. The proposed framework is mainly categorized into three steps, as shown in Figure 2. In the first step, we perform the preprocessing on raw data related to the building. In the second step, the polished data are redirected to GRU, where they learn hidden patterns in both forward and backward fashion. In the third step, the softmax layer generates the desired output and evaluates the performance of the model through various evaluation metrics. All the acronyms used throughout the paper are defined in Table 2.

Table 2. Parameters and their abbreviations.

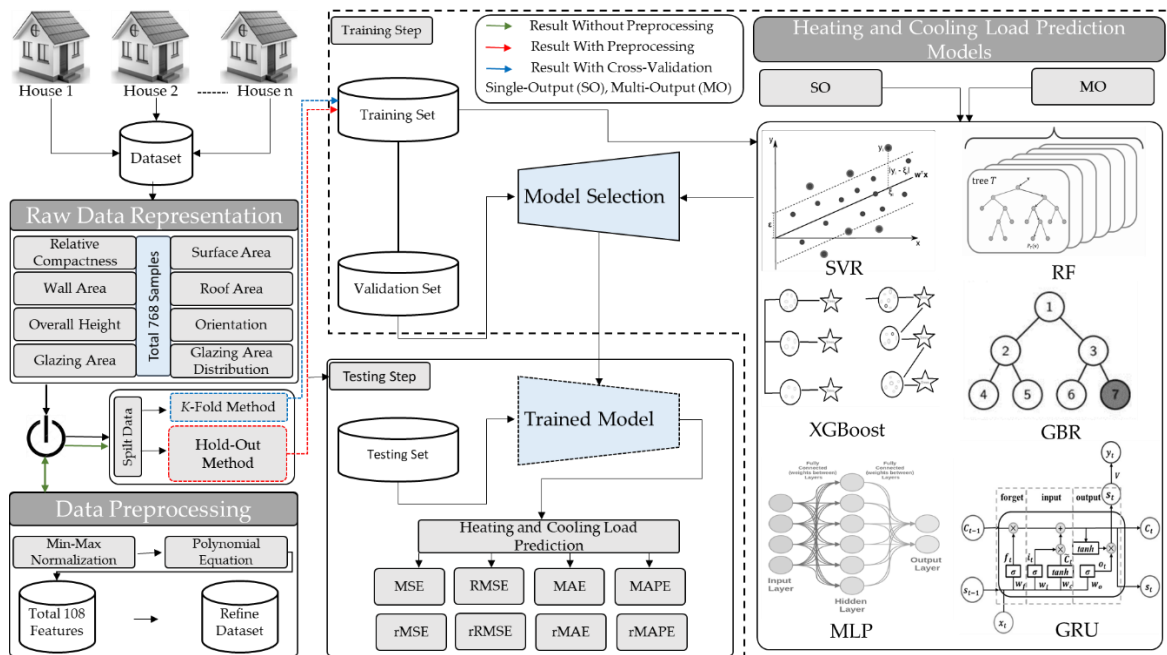| Acronyms	Description	Acronyms	Description
CL	Cooling load	MAE	Mean absolute error
HL	Heating load	RMSE	Root mean square error
GRU	Gated recurrent unit	SA	Sensitivity analysis
SVR	Support vector regression	SVM	Support vector machine
ANN	Artificial neural network	PCA	Principal component analysis
MLP	Multilayer perceptron	DNN	Deep neural network
ML	Machine learning	SO	Single-output
DL	Deep learning	MO	Multi-output
RF	Random forest	GPR	Gaussian process regression
MSE	Mean square error	GBR	Gradient boost regressor
SVM	Support vector machine	DMTs	Decision-making trees
rMSE	Relative mean square error	rRMSE	Relative root mean square error

Figure 2. The proposed framework for precise prediction of HL and CL through energy efficiency data using sequential learning model.

3.1. Data Acquisition and Preprocessing

Existing simulated data comprise various attributes related to building structure, such as relative compactness, overall area, height, orientation, and distribution of the glazing area. The performance of deep models depends on the characteristics of the data for training. For instance, if the input data are well organized, then they assist in efficient performance. In this study, we employ two techniques in the preprocessing step to refine the simulation because each of the attribute data in the dataset is scattered with each other that trigger the outlier issue. Therefore, we apply min–max normalization to remove outliers and arrange all the values in the range of 0 and 1. The alteration effect on data before and after normalization is visualized in Figure 3. Similarly, samples of the dataset are very few, and prior researchers always tried to tune a model for enhancing the performance of the model over limited data. From various computer vision problems, it has been proved that the DL model depicts a remarkable performance on a massive number of data [39]. However, in our case the total samples in the dataset are fewer; therefore, we apply a polynomial equation to increase the number of features. In this perspective, we also check the generalized capability of the model and decrease the overfitting probabilities during training. In addition, such model can be applicable for new data predictions in the future.

Before applying the polynomial equation, first, we made four sets of couples from the given eight attributes, where each couple is expanding up to the sixth power with various possible combinations of pairs. Equation (1) presents a polynomial process applied to a single pair of attributes.

$$Fv = \left(x, \ y, \ x^2, xy, \ y^2, \ x^3 \ldots \ldots \ldots xy^5, \ y^6\right). \tag{1}$$

where Fv represents the polynomial feature vector and x, y denotes two different attributes, where from each pair, 27 diverse feature vectors are produced, while in our case, we have 8 pairs, so the total possible feature vectors become ($27 \times 8 = 108$), which is our desired output.
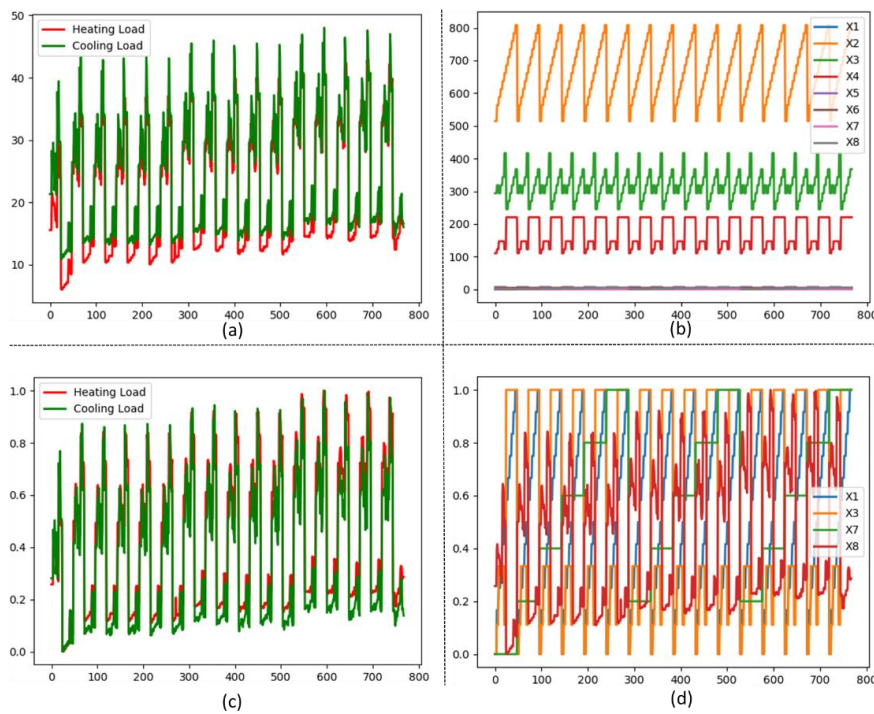
Figure 3. (**a**) Visual representation of actual HL and CL data where *x*-axis shows the number of samples and *y*-axis illustrates the range of samples; (**b**) overall attributes in the dataset; (**c**) normalized sample value of HL and CL; (**d**) normalized value of 4 attributes.

3.2. Multi-Output (MO) Regression Model

Various traditional approaches have been developed to efficiently predict HL and CL in an SO manner, where a model is trained two times for each prediction. The energy efficiency dataset contains a total of eight features (relative compactness, surface area, wall area, roof area, overall height, orientation, glazing area, and glazing area distribution) and two labels (HL and CL). Hence, there are no separate attributes and samples for HL and CL prediction; therefore, these models were first utilized for CL data prediction and then trained again for HL, which is a laborious and headache job.

In this paper, we propose an efficient model for both SO and MO with static parameters. First, we improve the performance of the model in SO, and then extra experiments are performed to evaluate the error rates of HL and CL in an MO fashion to save time. To the best of our knowledge, there is no existing model that can generate MO through a sequential learning approach. We performed comprehensive experiments on both SO and MO using various ML and DL techniques to find the optimal model that predicts the HL and CL in an efficient way (Figure 4).
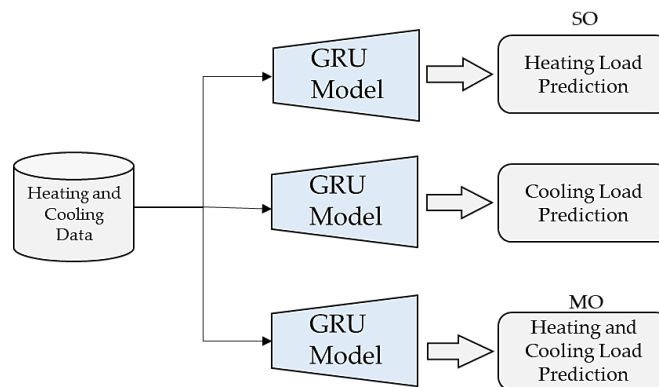

Figure 4. Difference between SO and MO in the prediction of HL and CL using the GRU model.

3.3. Machine Learning (ML)

3.3.1. Support Vector Regression (SVR)

Support vector machine (SVM) is a supervised ML approach mainly used for classification and regression problems. It has been proved from earlier studies that SVM shows incredibly better performance than other supervised learning algorithms because its formulation is based on structural risk minimization instead of empirical. In this paper, we deal with CL and HL prediction as a regression problem; therefore, we use SVR. The main goal of SVR is convex means that global optimum is always converge in this approach, first, the input data x is mapped via nonlinear mapping into n-dimensional feature space. In the consequent feature space, we build a linear model for the prediction of CL and HL. The mathematical background of the linear model is presented in Equation (2).

$$M_{(x,w)} = \left(\sum_{i=1}^{n} w_i \, h_i \, (x) + \, b \right). \tag{2}$$

where nonlinear transformations are denoted by $h \, (i) \, (x)$, $i = (1, \dots \dots \, n)$ as most of the time the bias "b" term is considered zero; therefore, it is ignored. The SVR model is totally dependent on the training data; therefore, it always tries to minimize the error rate and model complexity by reducing $\|w\|^2$, which is the main objective.

$$\left(w\text{in} 1/2\|w\|^2 + C \sum_{j=1}^{m} \xi_j + \, \xi_j^* \right) \tag{3}$$

$$\text{Such that } Z_j - M_{(x_j,w)} \le \varepsilon + \xi_j^*$$

$$M_{(x_i,w)} - z_i \le \varepsilon + \xi_j.$$

$$\xi_j, \xi_j^* \ge 0, \quad j = (1, \dots, m).$$

In Equation (3), ε denotes the loss function, while ξ_j and ξ_j^* are the slack non-negative variables that determine the training sample deviation. The parameter $C > 0$ is used to measure the fitting of the samples, and w illustrates the regression weights of the SVR model. Generally, the optimization task can be easily handled if converted into a dual problem by applying the dualization and Lagrange multipliers techniques. These methods are mathematically elaborated in Equation (4).

$$L = \left(\begin{array}{c} \frac{1}{2}\|w\|^2 + C\sum_{j=1}^{m}(\xi_j + \, \xi_j^*) - \sum_{j=1}^{m} \sigma_j^* \, (\varepsilon + \, \xi_j^* \, - Z_j + \, M_{(x_j,w)}) - \\ \sum_{j=1}^{m} \sigma_j \, (\varepsilon + \, \xi^* - Z_j + \, M_{(x_j,w)}) - \sum_{j=1}^{m} \, (\lambda_j \varepsilon + \, \lambda_j^* \xi_j^* \,), \end{array} \right) \tag{4}$$

In the Lagrange L equation, σ_j, σ_j^*, λ_j, $\lambda_j^* \ge 0$ refer to Lagrange multipliers. According to the consideration saddle point situation, the partial derivatives of the Lagrange variable (w, b, ξ_j, ξ_j^*) will disappear during optimality. Finally, in Equation (5) is the dual optimization along with the preceding steps.

$$g(x) = \sum_{j=1}^{Nsv} \left(\sigma_j - \sigma_j^* \right) K(x_j - \, x \,) \tag{5}$$

In the dual optimization procedure, K depicts the number of vectors in feature space and kernel function. The dot product of two vectors with kernel is represented in Equation (6).

$$K\left(x, x_j\right) = \sum_{j=1}^{n} \, g_j(x) \, g_j\left(x_j\right) \tag{6}$$

In this study, we use a Gaussian radial basis function (RBF) with its kernel parameters 'Y' in order to manage the nonlinearities between the input data and their perceptive class.

3.3.2. Random Forest (RF)

Random forest (RF) is a supervised learning algorithm and ensemble approach of randomized decision-making trees (DMTs). A DMT is a non-parametric ML algorithm that establishes a model in the form of a tree structure and divides the given records into smaller chunks until only one record remains in the subgroup. The final and internal sets are referred to as leaf and root nodes, respectively. The particular DMT is utilized for an unstable system that is totally dependent on data, so limited data affect the entire structure. To tackle this problem, a collection of DMTs is utilized to select target values based on average predicted values for all individual trees. Typically, RF follows the bags and boosts strategy in which they integrate different models sharing common information in order to produce several individual trees [40]. Multiple hyper parameters are required to tune the RF, but the number key parameter is the number of independent trees in the forest. To find an efficient model in terms of accuracy and time complexity, it is necessary to tune the model on different parameters.

3.3.3. XGBoost

XGBoost is a supervised learning approach that is applicable to both regression and classification problems. Here, we use multiple features X_i for training data to predict the target variable Y_i. XGBoost stands for "Extreme Gradient Boosting", which follows an ensemble learning strategy, including regression tree and classification mechanism. Let us assume $\{T_1 \ldots \ldots T_n \ (x_i, y_i)$ where, $i = 1, \ldots, n\}$, where x_i and y_i represent the training samples and the appropriate class labels. To get the final predicted score, all the individual scores are combined via additive function A as depicted in Equation (7).

$$y_i^* = \sum_{a=1}^{A} f_a(X_i), \ f_a \varepsilon \ F. \tag{7}$$

where f_a is the number of gradient boosting trees and functional spaces for all trees. The two additional functions (i.e., training and regularization term) are illustrated in Equation (8):

$$O_\theta = \sum_{i}^{m} 1\left(y_i, y_i^*\right) + \sum_{a=1}^{A} \psi(f_a). \tag{8}$$

where the measurement of the loss function between target and predicted class is illustrated as 1 and ψ is represent the regularization term used for handling the overfitting issue.

In XGBoost for each level '*t*', the additive training approach is followed for the prediction of each class label y_i^*. Mathematically, it can be shown in Equation (9).

$$y_i^{*t} = \sum_{a=1}^{A} f_a(x_i) = y_i^{*t-1} + f_t(x_i). \tag{9}$$

After applying the tree boosting method, Equation (8) becomes

$$O_\theta{}^t = \sum_{i}^{m} 1\left(y_i, y_i^{*t-1} + f_t(x_i)\right) + \psi(f_t). \tag{10}$$

Equation (10) is obtained by dividing the leaf node to get the final score after a series of evaluations.

3.4. Deep Learning (DL)

3.4.1. Multilayer Perceptron (MLP)

Inspired by the human brain, McCulloch and Pitts proposed the concept of ANN [41], which has been widely used in various research domains [42]. One of its fundamental potentials is generating the nonlinear relation map between the input and the output. In the current study, we choose MLP among various types of ANNs because it is one of the trustworthy methods designed for prediction problems. Our MLP architecture comprises three layers along with computational units called neurons, as shown in Figure 5. To achieve the actual and predicted outputs precisely, it is necessary to design an efficient

model. Moreover, let us assume the input vectors are denoted by I; then the final output for the *n*-th neuron is formulated by Equation (11):

$$O_k = A\left(\sum_{i=1}^{j} I_i W_{ik} + b_k\right). \tag{11}$$

where *I* illustrate the input samples, and *W*, *b*, and *A* represent the weights, bias, and activation function, respectively.
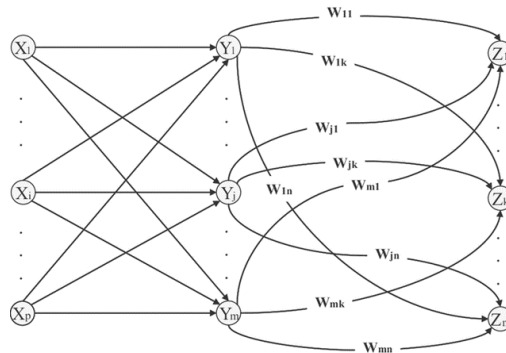

Figure 5. Backpropagation of MLP architecture with one hidden layer.

3.4.2. Gated Recurrent Unit (GRU)

GRU is the most common sequence learning model of Recurrent Neural Network (RNN), which is basically used to overcome the vanishing gradient issue [43]. GRU contains two main gates (i.e., update and reset gate), that determine which information is to be transferred to the output layer as shown in Figure 6. The unique capability of these two gates is that they can store information for the long term and they do not remove irrelevant information related to the prediction. They can be considered as a subset of Long Shot-Term Memory (LSTM) due to similarity in architecture and performance. In the current study, we use GRU because it gives a remarkable performance on smaller numbers of sample data as well as on more samples. Furthermore, it trains faster due to a smaller number of parameters. The mathematics behind this network is shown in Equations (12)–(15).

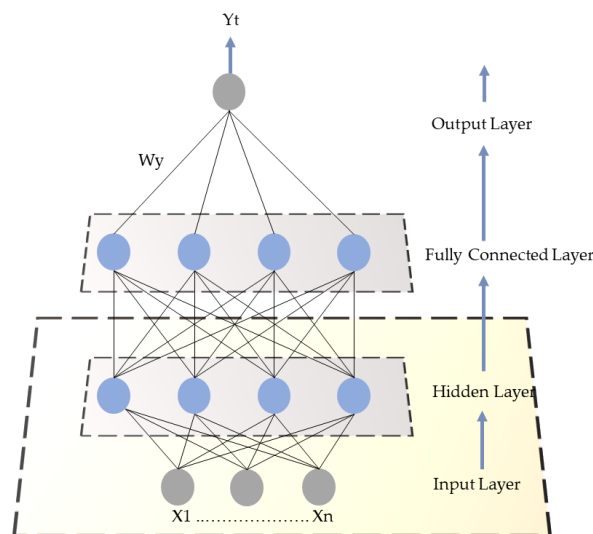$$Z_t = \Theta(W_{(z)} X_t + U_{(z)} h_{(t-1)}). \tag{12}$$

Figure 6. The GRU architecture for HL and CL prediction.

In the update gate, as the input X_t is tied to the network node, it is multiplied by its weights $W_{(z)}$. Similarly, $h_{(t-1)}$ retains the information of the earlier cell units and is multiplied by its particular weights $U_{(z)}$. The outcomes of both are integrated and flatten the result between the range of 0 and 1 through sigmoid activation function.

$$r_t = \Theta(W_{(r)} X_t + U_{(r)} h_{(t-1)}). \tag{13}$$

The reset gate is almost similar to the update gate, but the difference is only in weights and functionality. This gate is basically used to decide how much previous information is to be forgotten. As $h_{(t-1)}$ and X_t are multiplied with their subsequent weights. After that, the sigmoid function is applied to the merged results.

$$\hat{h}_t = \tan h(Wx_t + r_t \bullet Uh_{t-1}). \tag{14}$$

The reset of the gate memory content is used to hold the relevant pattern information from the earlier cell gate. Here, two operations are performed: first, the input sequence X_t is multiplied with its corresponding weights W, and second, an element-wise operation is performed between r_t and $Uh_{(t-1)}$, which will take a decision in removing the information from the earlier time steps. Furthermore, it sums up both the generated output and the employed nonlinear tan*h* activation function.

$$h_t = \left(z_t \bullet h_{t-1} + (1 - z_t) \bullet \hat{h}_t\right). \tag{15}$$

Finally, the network calculates the h_t vector, and the last memory of the current time step holds the current unit information and forwards it to the next layer of the network in order to update the gate. Furthermore, it contains information about the current and previous steps. In this unit cell, three basis operations are performed, first, to update the gate element-wise product operated between z_t and $h_{(t-1)}$. Second, again element-by-element multiplication is done between $(1 - zt)$ and $\hat{h}t$. In the last operation, both operations 1 and 2 are incorporated, and the final output is produced.

4. Experimental Results

4.1. System Configuration

We substantiate the effectiveness of the proposed GRU model using an energy efficiency dataset that is publicly available on the University of California Irvine (UCI) repository. The model is trained over TITAN X (Pascal)/PCLe/SSE2 GPU with Intel Core i5-6600 processor, 64 GB RAM, and Windows 10 operating system. The implementation is performed in Python with Keras DL framework with TensorFlow at the backend and Adam optimizer with 100 epochs, 0.0001 learning rate, 0.9 momentum, 16 batch size, and 20 units in GRU. Two types of experiments are conducted, including hold-out and cross-validation on the energy efficiency dataset, which are further categorized into SO and MO with and without preprocessing. Furthermore, in the hold-out method, the data are divided into training set 80% and testing 20%, and from the training set, we select 10% data as a validation set. On the other hand, in the cross-validation process, the entire data are divided into 10 equal parts. After that, one chunk is used for the testing set, and the remaining ten chunks are considered as training set. This process is repeated until each chunk is tested.

4.2. Dataset Description

The dataset used for this study is proposed by [12] with no additional characteristics. Through an elementary cubes process, 12 residential building shapes were simulated, each comprising 18 elements, and the total volume of the buildings was 771.75 m^3. The building relative compactness is computed by evaluating the total areas of the building structure and position structure when the total volume of the building is equal to the position structure. The selection was made by the newest and most common materials used during constructing building industry and by the lowest U-value. Building structure

features and their related (U-values appear in parenthesis): walls (1.780), floors (0.860), roofs (0.500), and windows (2.260). The simulated data is assumes that the actual buildings are located in Athens, Greece. The interior layout was set as clothing: 0.6 clo; room temperature: 21 °C; thermal insulation that is suitable for a particular weather condition; humidity rate: 60%; air velocity: 0.3 m/s; and a total of 300 lux lights. The heating properties have shown 95% efficiency with a thermostat range of 19–24 °C that is working 15–20 h on weekdays and 10–20 h on weekends. Three different percentages were utilized for the glazing area of a building: 10%, 25%, and 40%. Furthermore, the glazing area was categorized into five various scenarios: uniform, 25%; north, 55%; east, 55%; south, 55%; and west, 55%. In addition, four directions were indicated by 2, 3, 4, and 5, which represent north facing, south facing, east facing, and west facing, respectively.

The energy efficiency dataset contains 12 shapes of a building along with 3 glazing areas, 5 different distribution scenarios for each glazing areas, and 4 directions, which are associated with 720 samples. However, if we enter the structure of the 12 buildings, which have no glazing area along 4 orientations, then the total becomes 768 buildings with respective values of HL and CL. The key characteristics of inputs and outputs are given in Table 3.

Table 3. Detailed description of energy efficiency dataset.

Variable	Building Information	Attribute	Total Values	Data Type	Units
Input	Relative compactness	X_1	12	Real	None
	Surface area	X_2	12	Real	m^2
	Wall area	X_3	07	Real	m^2
	Roof area	X_4	04	Real	m^2
	Overall height	X_5	02	Real	M
	Orientation	X_6	04	Integer	None
	Glazing area	X_7	04	Real	None
	Glazing area distribution	X_8	6	Integer	None
Output	Heating load	Y_1	586	Real	kWh/m^2
	Cooling load	Y_2	636	Real	kWh/m^2

4.3. Evaluation Metrics

To assess the variation between the actual and predicted heating and cooling load, it is necessary to evaluate the effectiveness of the regression model via various evaluation metrics, such as MSE, RMSE, MAE, and MAPE. Basically, MSE calculates the average square value of the difference between the target and predicted values via the regression model. RMSE is commonly used for a regression problem, which is the root squared difference between the actual and predicted values, and MAE is the linear score in which the individual weighted differences are considered equally. Finally, the MAPE metric computes the prediction accuracy in percentage. The mathematical representation of all these metrics is depicted in Equations (16)–(19).

$$\text{MSE} = \frac{1}{n} \sum_{1}^{n} (y - \hat{y})^2 \tag{16}$$

$$\text{MAE} = \frac{1}{n} \sum_{1}^{n} |y - \hat{y}| \tag{17}$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{1}^{n} (y - \hat{y})^2} \tag{18}$$

$$\text{MAPE} = \frac{100\%}{n} \sum_{1}^{n} \frac{|y - \hat{y}|}{y} \tag{19}$$

4.4. Performance of ML and DL Methods for SO

In this study, we performed various experiments on ML and DL methods to select the optimal model for SO. The experiments were performed with and without preprocessing followed by two strategies: hold-out and cross-validation. In the ML models, XGBoost beat all other ML algorithms in both cases because it utilized the ensemble learning strategy. In HL and CL prediction, it showed remarkable performance as depicted in Table 4. Further, we checked the prediction error in various kernels in SVR as shown in Figure 7.

Table 4. Experimental results of various ML and DL models for SO prediction using the hold-out method.

Method	Hold-Out without Preprocessing							
	HL				CL			
	MAE	MSE	RMSE	MAPE	MAE	MSE	RMSE	MAPE
SVR	1.9532	1.5241	1.2345	1.3913	2.2143	1.6241	1.2744	1.7471
RF	2.4310	1.8701	1.3675	1.6714	2.4197	1.9875	1.4097	1.9032
XGBoost	1.8236	1.4797	1.2164	1.5941	2.1027	1.5579	1.2481	1.6179
GBR	2.3142	1.6091	1.2685	1.6721	2.3471	1.7928	1.3389	1.8932
MLP	1.7613	0.9781	0.9889	1.1741	1.9897	1.0899	1.0439	1.4869
GRU	1.3691	0.7215	0.8494	0.9315	1.4027	0.9791	0.9894	1.0132

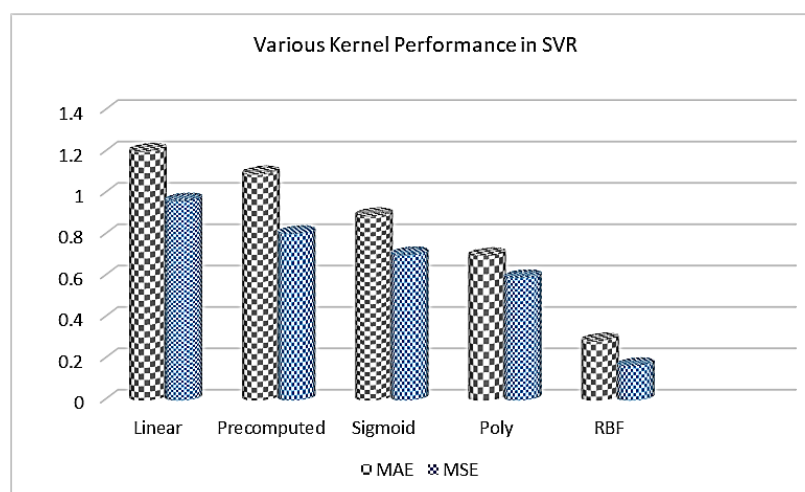| Method	Hold-Out with Preprocessing							
	HL				CL			
	MAE	MSE	RMSE	MAPE	MAE	MSE	RMSE	MAPE
SVR	0.2855	0.1658	0.4072	0.5833	0.5662	0.6851	0.8277	0.9428
RF	0.3225	0.1924	0.4386	0.5312	1.0212	2.3355	3.7084	3.8192
XGBoost	0.2130	0.0911	0.3018	0.4120	0.4167	0.3566	0.5971	0.6580
GBR	0.3048	0.1467	0.3830	0.5269	0.9311	0.5971	2.7084	2.8149
MLP	0.0853	0.0075	0.0867	0.0988	0.0838	0.0074	0.0858	0.0897
GRU	0.0102	0.0003	0.0166	0.0284	0.0167	0.0006	0.0247	0.0368

Figure 7. Numerous kernel performances of SVR in the prediction of HL and CL.

In contrast to ML algorithms, GRU significantly predicted the HL and CL in both processed and unprocessed data in DL algorithms. In refined data, GRU achieved 0.0102, 0.0003, 0.0166, and 0.0284, for MAE, MSE, RMSE, and MAPE in HL prediction and 0.0167, 0.0006, 0.0247, and 0.0368 for MAE, MSE, RMSE, and MAPE in CL prediction, respectively. The rest of the algorithm's performance are illustrated in Table 4.

It is clear from Table 4 that our preprocessing strategy significantly reduced the error rate in all methods. Besides hold-out, we also did the cross-validation for the prediction of HL and CL, where again XGBoost showed better performance than other ML methods, while GRU dominated all methods, including XGBoost and MLP, because it learns more from data in an efficient way. The comprehensive experiments are shown in Table 5.

Table 5. Experimental results of various ML and DL models for SO prediction using the 10-fold method.

Method	Cross-Validation without Preprocessing							
	HL				CL			
	MAE	MSE	RMSE	MAPE	MAE	MSE	RMSE	MAPE
SVR	2.0978	1.6463	1.2830	1.4192	2.2089	1.7574	1.3256	1.5303
RF	2.5421	1.9943	1.4121	1.6971	2.6532	2.0215	1.4217	1.7082
XGBoost	1.9347	1.5998	1.2648	1.4023	2.0458	1.7110	1.3080	1.5134
GBR	2.4235	1.7497	1.3227	1.5932	2.5346	1.8608	1.3641	1.7043
MLP	1.8724	1.4996	1.2245	1.4932	1.9835	1.6107	1.2691	1.6043
GRU	1.4802	0.9871	0.9935	1.0210	1.5913	0.8920	0.9444	1.1031
Method	Cross-Validation with Preprocessing							
	HL				CL			
	MAE	MSE	RMSE	MAPE	MAE	MSE	RMSE	MAPE
SVR	0.1941	0.0431	0.2076	0.3712	0.1830	0.0320	0.1788	0.2823
RF	0.2916	0.0981	0.3132	0.4312	0.2805	0.0870	0.2949	0.5024
XGBoost	0.1813	0.0334	0.1827	0.2715	0.1701	0.0231	0.1519	0.2529
GBR	0.2712	0.0849	0.2913	0.3108	0.2601	0.0738	0.2716	0.3914
MLP	0.0191	0.0091	0.0953	0.1076	0.0189	0.0080	0.0894	0.1289
GRU	0.0092	0.0001	0.0100	0.0391	0.0021	0.0001	0.0100	0.0282

4.5. Performance of ML and DL Methods for MO

In this section, we conducted numerous experiments on ML and DL models in order to pick the optimal model on the basis of their performance. State-of-the-art models generated output in a SO fashion that required more time for the prediction of HL and CL. There is no existing work that employed a sequence learning model to generate the desired output in MO style. The same set of experiments for hold-out and 10-fold cross-validation with and without refining data are conducted for MO model assessment. In the ML category, XGBoost showed a convincing performance, while the proposed model (GRU) showed overall dominancy in both hold-out and 10-fold cross-validation because it only keeps and learns the most prominent information to make HL and CL predictions and eliminates irrelevant information. Detailed results are given in Table 6.

Table 6. Experimental results of various ML and DL models for MO prediction using the hold-out method.

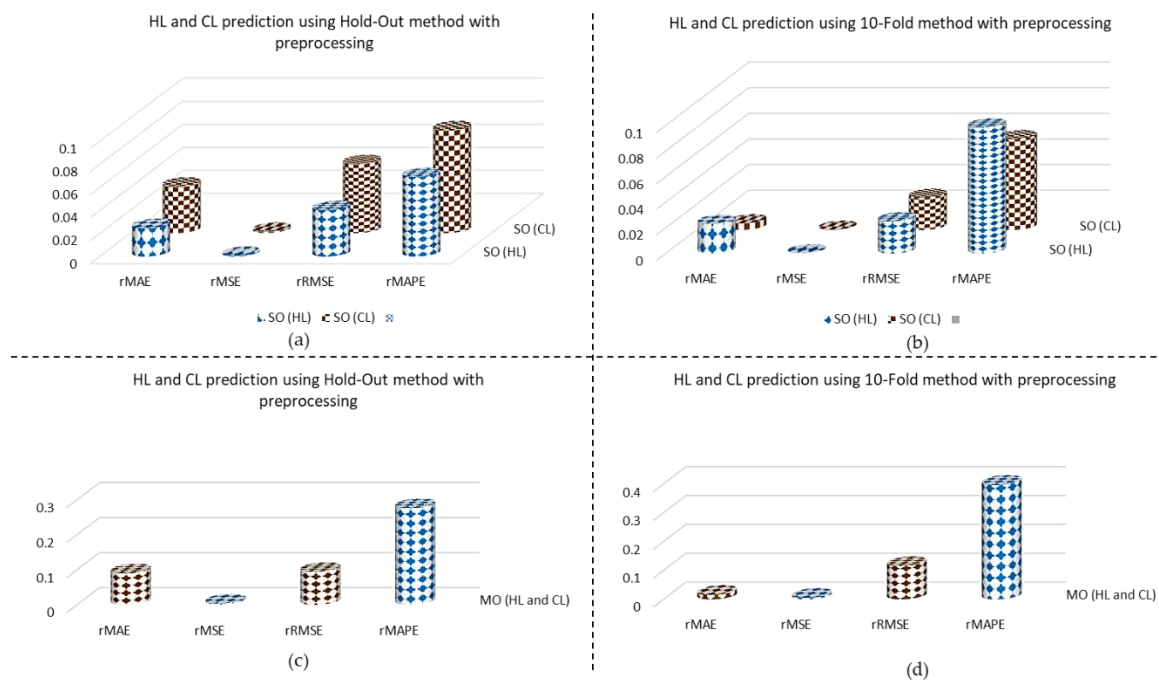| Method	Hold-Out with Preprocessing				Hold-Out without Preprocessing			
	HL and CL				HL and CL			
	MAE	MSE	RMSE	MAPE	MAE	MSE	RMSE	MAPE
SVR	0.7831	0.5479	0.7402	0.8922	3.5347	2.3701	1.5395	2.6368
RF	0.9867	0.7863	0.8867	0.9647	3.9375	2.5561	1.5987	2.8059
XGBoost	0.5182	0.4841	0.6957	0.7328	3.2439	2.1253	1.4578	2.5278
GBR	0.6798	0.6531	0.8081	0.9781	3.7294	2.4321	1.5595	2.7053
MLP	0.0953	0.0189	0.1374	0.2579	2.9124	1.9760	1.4057	1.9979
GRU	0.0368	0.0015	0.0387	0.1134	1.7519	1.0217	1.0107	1.0901
	Cross-Validation with Preprocessing				Cross-Validation without Preprocessing			
SVR	0.6975	0.4043	0.6358	0.7098	3.4438	2.2903	1.5133	2.8186
RF	0.8790	0.6901	0.8307	0.9767	3.8466	2.4650	1.5700	3.0077
XGBoost	0.4791	0.3765	0.6135	0.7452	3.1529	2.0344	1.4263	2.7096
GBR	0.5170	0.5536	0.7440	0.8062	3.6385	2.3412	1.5300	2.9071
MLP	0.3732	0.1932	0.4395	0.5690	2.8215	1.8851	1.3729	2.0707
GRU	0.0062	0.0021	0.0458	0.1574	1.6608	1.0308	1.0152	1.0724

Figure 8. Experimental results of the proposed model (GRU) for SO and MO prediction using the hold-out and 10-fold methods.

For more satisfaction of the proposed model, we also visualized actual and predicted load on both SO and MO as illustrated in Figure 9.
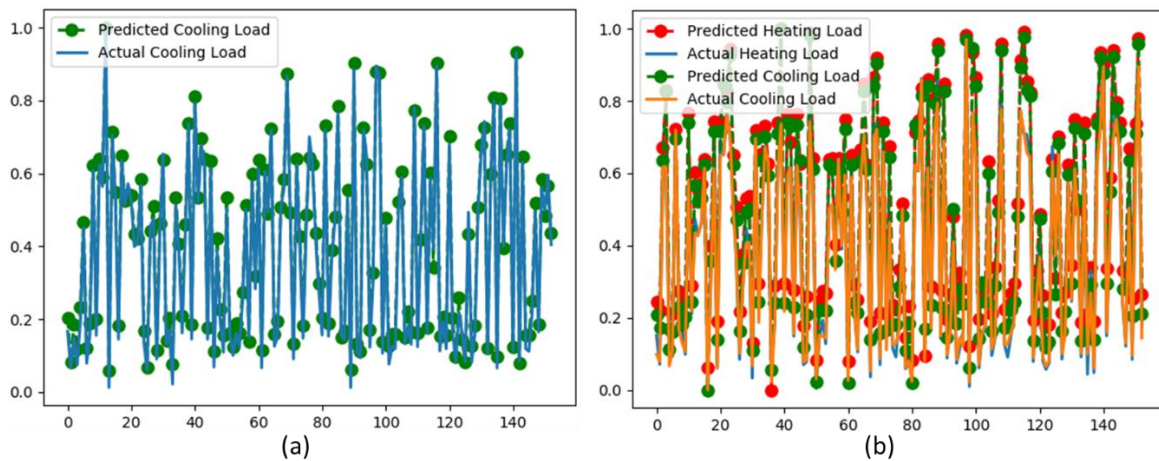

Figure 9. Visualization of prediction results obtained via the proposed model (GRU), where *x*-axis indicates the number of samples while *y*-axis represents the actual and predicted load; (**a**) the actual and predicted outputs of CL using the SO strategy; (**b**) the actual and predicted outputs of HL and CL using the MO strategy.

4.6. Comparison with State-Of-The-Art Models

In this section, we compared the results achieved through the proposed model (GRU) with existing approaches over energy efficiency dataset. The proposed model showed a remarkable performance on both hold-out and 10-fold cross-validation as compared to state-of-the-art models. Most of the researchers developed ML and ANN methods to evaluate HL and CL without utilizing a preprocessing technique, which sometimes generated false prediction. Furthermore, they repeated the training process to obtain the HL and CL values individually, which is a very tedious and time consuming

job. In a such way, most of time the performance of the HL improved, but, the accuracy of the CL prediction decrease. In contrast, our proposed model is better than the others existing baseline models because, we employed a sequential learning model for nonsequential data which improved the SO and MO performances on both hold-out and 10-fold. Table 7 presents the SO results based on the hold-out technique with recent state-of the-art models [2,4,12,22,23,25,26,28,31–36,44]. For HL prediction, the proposed model (GRU) achieved the least error rates for MAE (0.0102), MSE (0.0003), and RMSE (0.0166). Similarly, the proposed model (GRU) achieved an incredible performance and attained the best results for CL (i.e., 0.0167, 0.0006, and 0.0247 for MAE, MSE, and RMSE, respectively).

Table 7. Comparison of the proposed model (GRU) for HL and CL prediction with state-of-the-art models.

Method	HL			CL		
	MAE	MSE	RMSE	MAE	MSE	RMSE
Tsanas and Xifara [12]	0.51	-	-	1.42	-	-
Chou and Bui [22]	0.236	-	0.346	0.89	-	1.566
Cheng and Cao [23]	0.35	-	0.47	0.71	-	1
Sonmez et al. [25]	0.61	-	-	1.25	-	-
Alam et al. [26]	-	-	0.19	-	-	1.42
Regina and Capriles [28]	0.246	-	1.094	0.39	-	1.284
Nilashi et al. [31]	0.16	-	0.26	0.52	-	0.81
Nwulu [32]	0.977	-	1.228	1.654	-	2.111
Duarte et al. [33]	0.315	-	0.223	0.565	-	0.837
Roy et al. [2]	0.037	-	0.053	0.127	-	0.195
Kavaklioglu [34]	-	-	3.16	-	-	3.122
Kumar et al. [35]	0.138		0.321	0.134	-	0.646
Al-Rakhami et al. [36]	0.175	-	0.265	0.307	-	0.47
Sekhar et al. [4]	-	-	0.059		-	0.079
Sadeghi et al. [44]	0.2	-	0.263	0.485	-	0.69
Proposed (hold-out)	0.0102	0.0003	0.0166	0.0167	0.0006	0.0247
Proposed (10-fold)	0.0092	0.0001	0.0100	0.0021	0.0001	0.0100

5. Conclusions and Future Research Direction

In this study, we proposed an intelligent framework for HL and CL prediction via a sequential learning model (GRU). First, we applied min-max normalization and polynomial equation in order to remove outliers, normalize all the sample values in specific range, and increase the number of features, respectively. Next, we conducted comprehensive set of experiments over ML and DL methods using hold-out and 10-fold cross-validation to choose the most favorable model in terms of accuracy. Finally, to validate the performance of the proposed model we evaluated it on numerous metrics such as MAE, rMAE, MSE, rMSE, RMSE, rRMSE. In the future, we aim to utilize evolutionary algorithms on different datasets and improve the existing performance of the model by utilizing some advanced preprocessing strategies.

Author Contributions: Conceptualization, M.S., S.U.K. and A.U.; Data curation, I.U.H.; Formal analysis, M.Y.L.; Funding acquisition, S.W.B.; Methodology, S.U.K. and N.K.; Project administration, M.Y.L. and S.W.B.; Software, N.K.; Supervision, and S.W.B.; Validation, M.S. and N.K.; Visualization, I.U.H.; Writing–original draft, M.S. and S.U.K.; Writing–review & editing, I.U.H. and A.U. All authors have read and agreed to the published version of the manuscript.

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

References

1. Li, C.; Ding, Z.; Zhao, D.; Yi, J.; Zhang, G. Building energy consumption prediction: An extreme deep learning approach. *Energies* **2017**, *10*, 1525. [CrossRef]

2. Roy, S.S.; Roy, R.; Balas, V.E. Estimating heating load in buildings using multivariate adaptive regression splines, extreme learning machine, a hybrid model of MARS and ELM. *Renew. Sustain. Energy Rev.* **2018**, *82*, 4256–4268.

3. Yu, W.; Li, B.; Lei, Y.; Liu, M. Analysis of a residential building energy consumption demand model. *Energies* **2011**, *4*, 475–487. [CrossRef]

4. Roy, S.S.; Samui, P.; Nagtode, I.; Jain, H.; Shivaramakrishnan, V.; Mohammadi-Ivatloo, B. Forecasting heating and cooling loads of buildings: A comparative performance analysis. *J. Ambient Intell. Humaniz. Comput.* **2020**, *11*, 1253–1264. [CrossRef]

5. Bouktif, S.; Fiaz, A.; Ouni, A.; Serhani, M.A. Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. *Energies* **2018**, *11*, 1636. [CrossRef]

6. Seyedzadeh, S.; Rahimian, F.P.; Glesk, I.; Roper, M. Machine learning for estimation of building energy consumption and performance: A review. *Vis. Eng.* **2018**, *6*, 5. [CrossRef]

7. Bagheri, A.; Feldheim, V.; Ioakimidis, C.S. On the evolution and application of the thermal network method for energy assessments in buildings. *Energies* **2018**, *11*, 890. [CrossRef]

8. Ardjmand, E.; Millie, D.F.; Ghalehkhondabi, I.; Young, I.; William, A.; Weckman, G.R. A state-based sensitivity analysis for distinguishing the global importance of predictor variables in artificial neural networks. *Adv. Artif. Neural Syst.* **2016**, *2016*, 2303181. [CrossRef]

9. Weckman, G.; Millie, D.; Ganduri, C.; Rangwala, M.; Young, W.; Rinder, M.; Fahnenstiel, G. Knowledge extraction from the neural 'black box' in ecological monitoring. *Int. J. Ind. Syst. Eng.* **2009**, *3*, 38–55.

10. Millie, D.F.; Weckman, G.R.; Young, W.A., II; Ivey, J.E.; Carrick, H.J.; Fahnenstiel, G.L. Modeling microalgal abundance with artificial neural networks: Demonstration of a heuristic 'Grey-Box' to deconvolve and quantify environmental influences. *Environ. Model. Softw.* **2012**, *38*, 27–39. [CrossRef]

11. Yeung, D.S.; Cloete, I.; Shi, D.; Ng, W. *Sensitivity Analysis for Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2010.

12. Tsanas, A.; Xifara, A. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy Build.* **2012**, *49*, 560–567. [CrossRef]

13. Yezioro, A.; Dong, B.; Leite, F. An applied artificial intelligence approach towards assessing building performance simulation tools. *Energy Build.* **2008**, *40*, 612–620. [CrossRef]

14. Ul Haq, I.; Ullah, A.; Muhammad, K.; Lee, M.Y.; Baik, S.W. Personalized movie summarization using deep cnn-assisted facial expression recognition. *Complexity* **2019**, *2019*, 3581419. [CrossRef]

15. Hussain, T.; Muhammad, K.; Ullah, A.; Del Ser, J.; Baik, S.W. Multi-View Summarization and Activity Recognition Meet Edge Computing in IoT Environments. *IEEE Internet Things J.* **2020**. [CrossRef]

16. Ullah, A.; Haydarov, K.; Ul Haq, I.; Muhammad, K.; Rho, S.; Lee, M.; Baik, S.W. Deep Learning Assisted Buildings Energy Consumption Profiling Using Smart Meter Data. *Sensors* **2020**, *20*, 873. [CrossRef]

17. Khan, S.U.; Baik, R. MPPIF-Net: Identification of Plasmodium Falciparum Parasite Mitochondrial Proteins Using Deep Features with Multilayer Bi-directional LSTM. *Processes* **2020**, *8*, 725. [CrossRef]

18. Ullah, F.U.M.; Ullah, A.; Muhammad, K.; Haq, I.U.; Baik, S.W. Violence detection using spatiotemporal features with 3D convolutional neural network. *Sensors* **2019**, *19*, 2472. [CrossRef]

19. Khan, S.U.; Haq, I.U.; Rho, S.; Baik, S.W.; Lee, M.Y. Cover the violence: A novel Deep-Learning-Based approach towards violence-detection in movies. *Appl. Sci.* **2019**, *9*, 4963. [CrossRef]

20. Ullah, A.; Muhammad, K.; Hussain, T.; Baik, S.W. Conflux LSTMs Network: A Novel Approach for Multi-View Action Recognition. *Neurocomputing* **2020**, *414*, 90–100.

21. Bourdeau, M.; Zhai, X.-Q.; Nefzaoui, E.; Guo, X.; Chatellier, P. Modelling and forecasting building energy consumption: A review of data-driven techniques. *Sustain. Cities Soc.* **2019**, *48*, 101533. [CrossRef]

22. 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]

23. Cheng, M.-Y.; Cao, M.-T. Accurately predicting building energy performance using evolutionary multivariate adaptive regression splines. *Appl. Soft Comput.* **2014**, *22*, 178–188. [CrossRef]

24. Aqlan, F.; Ahmed, A.; Srihari, K.; Khasawneh, M.T. Integrating artificial neural networks and cluster analysis to assess energy efficiency of buildings. In Proceedings of the IIE Annual Conference, Montreal, QC, Canada, 31 May–4 June 2014; pp. 3936–3943.

25. Sonmez, Y.; Guvenc, U.; Kahraman, H.T.; Yilmaz, C. A comperative study on novel machine learning algorithms for estimation of energy performance of residential buildings. In Proceedings of the 2015 3rd International Istanbul Smart Grid Congress and Fair (ICSG), Istanbul, Turkey, 29–30 April 2015; pp. 1–7.

26. Alam, A.G.; Baek, C.I.; Han, H. Prediction and Analysis of Building Energy Efficiency Using Artificial Neural Network and Design of Experiments. In *Applied Mechanics and Materials*; Trans Tech Publications Ltd.: Stafa-Zurich, Switzerland, 2016; pp. 541–545.

27. Yang, F.; Gao, P.; Lu, Y. Evolving Resilient Back-Propagation Algorithm for Energy Efficiency Problem. In Proceedings of the 2016 3rd International Conference on Mechanics and Mechatronics Research (ICMMR 2016), Chongqing, China, 15–17 June 2016; EDP Sciences: Les Ulis, UK, 2016; p. 06016.

28. Duarte, G.; Capriles, P.; Goliatt, L.; Lemonge, A. Prediction of energy load of buildings using machine learning methods. In Proceedings of the 4th Conference of Computational Interdisciplinary Science (CCIS), Sao Jose dos Campus, Brazil, 7–10 November 2016.

29. Naji, S.; Shamshirband, S.; Basser, H.; Keivani, A.; Alengaram, U.J.; Jumaat, M.Z.; Petković, D. Application of adaptive neuro-fuzzy methodology for estimating building energy consumption. *Renew. Sustain. Energy Rev.* **2016**, *53*, 1520–1528. [CrossRef]

30. Naji, S.; Keivani, A.; Shamshirband, S.; Alengaram, U.J.; Jumaat, M.Z.; Mansor, Z.; Lee, M. Estimating building energy consumption using extreme learning machine method. *Energy* **2016**, *97*, 506–516. [CrossRef]

31. Nilashi, M.; Dalvi-Esfahani, M.; Ibrahim, O.; Bagherifard, K.; Mardani, A.; Zakuan, N. A soft computing method for the prediction of energy performance of residential buildings. *Measurement* **2017**, *109*, 268–280. [CrossRef]

32. Nwulu, N.I. An artificial neural network model for predicting building heating and cooling loads. In Proceedings of the 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, 16–17 September 2017; pp. 1–5.

33. Duarte, G.R.; da Fonseca, L.G.; Goliatt, P.V.Z.C.; de Castro Lemonge, A.C. Comparison of machine learning techniques for predicting energy loads in buildings. *Ambiente Construído* **2017**, *17*, 103–115. [CrossRef]

34. Kavaklioglu, K. Robust modeling of heating and cooling loads using partial least squares towards efficient residential building design. *J. Build. Eng.* **2018**, *18*, 467–475. [CrossRef]

35. Kumar, S.; Pal, S.K.; Singh, R.P. Intra ELM variants ensemble based model to predict energy performance in residential buildings. *Sustain. Energy Grids Netw.* **2018**, *16*, 177–187. [CrossRef]

36. Al-Rakhami, M.; Gumaei, A.; Alsanad, A.; Alamri, A.; Hassan, M.M. An Ensemble Learning Approach for Accurate Energy Load Prediction in Residential Buildings. *IEEE Access* **2019**, *7*, 48328–48338. [CrossRef]

37. Cariboni, J.; Gatelli, D.; Liska, R.; Saltelli, A. The role of sensitivity analysis in ecological modelling. *Ecol. Model.* **2007**, *203*, 167–182. [CrossRef]

38. Saltelli, A.; Ratto, M.; Andres, T.; Campolongo, F.; Cariboni, J.; Gatelli, D.; Saisana, M.; Tarantola, S. *Global Sensitivity Analysis: The Primer*; John Wiley & Sons: Hoboken, NJ, USA, 2008.

39. Khan, N.; Ullah, A.; Haq, I.U.; Menon, V.G.; Baik, S.W. SD-Net: Understanding overcrowded scenes in real-time via an efficient dilated convolutional neural network. *J. Real-Time Image Process.* **2020**, 1–15. [CrossRef]

40. Liu, D.; Sun, K. Random forest solar power forecast based on classification optimization. *Energy* **2019**, *187*, 115940. [CrossRef]

41. McCulloch, W.S.; Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [CrossRef]

42. Moayedi, H.; Rezaei, A. An artificial neural network approach for under-reamed piles subjected to uplift forces in dry sand. *Neural Comput. Appl.* **2019**, *31*, 327–336. [CrossRef]

43. Sajjad, M.; Khan, Z.A.; Ullah, A.; Hussain, T.; Ullah, W.; Lee, M.Y.; Baik, S.W. A novel CNN-GRU-based hybrid approach for short-term residential load forecasting. *IEEE Access* **2020**, *8*, 143759–143768. [CrossRef]

44. Sadeghi, A.; Younes Sinaki, R.; Young, W.A.; Weckman, G.R. An intelligent model to predict energy performances of residential buildings based on deep neural networks. *Energies* **2020**, *13*, 571. [CrossRef]