

Article

Short-Term Energy Forecasting to Improve the Estimation of Demand Response Baselines in Residential Neighborhoods: Deep Learning vs. Machine Learning

Abdo Abdullah Ahmed Gassar 

Laboratory of Engineering Sciences for the Environment (LaSIE, UMR CNRS 7356), La Rochelle University, 17000 La Rochelle, France; gassar@khu.ac.kr

Abstract: Promoting flexible energy demand through response programs in residential neighborhoods would play a vital role in addressing the issues associated with increasing the share of distributed solar systems and balancing supply and demand in energy networks. However, accurately identifying baseline-related energy measurements when activating energy demand response events remains challenging. In response, this study presents a deep learning-based, data-driven framework to improve short-term estimates of demand response baselines during the activation of response events. This framework includes bidirectional long-term memory (BiLSTM), long-term memory (LSTM), gated recurrent unit (GRU), convolutional neural networks (CNN), deep neural networks (DNN), and recurrent neural networks (RNN). Their performance is evaluated by considering different aggregation levels of the demand response baseline profile for 337 dwellings in the city of La Rochelle, France, over different time horizons, not exceeding 24 h. It is also compared with fifteen traditional statistical and machine learning methods in terms of forecasting accuracy. The results demonstrated that deep learning-based models, compared to others, significantly succeeded in minimizing the gap between the actual and forecasted values of demand response baselines at all different aggregation levels of dwelling units over the considered time-horizons. BiLSTM models, followed by GRU and LSTM, consistently demonstrated the lowest mean absolute percentage error (MAPE) in most comparison experiments, with values up to 9.08%, 8.71%, and 9.42%, respectively. Compared to traditional statistical and machine learning models, extreme gradient boosting (XGBoost) was among the best, with a value up to 11.56% of MAPE, but could not achieve the same level of forecasting accuracy in all comparison experiments. Such high performance reveals the potential of the proposed deep learning approach and highlights its importance for improving short-term estimates of future baselines when implementing demand response programs in residential neighborhood contexts.

Keywords: residential neighborhoods; demand response programs; baselines; short-term forecasts; deep learning; machine learning methods



Citation: Gassar, A.A.A. Short-Term Energy Forecasting to Improve the Estimation of Demand Response Baselines in Residential Neighborhoods: Deep Learning vs. Machine Learning. *Buildings* **2024**, *14*, 2242. <https://doi.org/10.3390/buildings14072242>

Academic Editor: Apple L.S. Chan

Received: 27 May 2024

Revised: 11 July 2024

Accepted: 18 July 2024

Published: 21 July 2024



Copyright: © 2024 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The European residential sector accounts for approximately 75% of European buildings and is solely responsible for over 25% of the final energy demand in the European Union (EU), making it the second-largest consumer after transport [1]. With this in mind, increasing the energy efficiency of residential and non-residential buildings is one of the main objectives of EU strategies to achieve the ambitious endeavor of decarbonizing European buildings [2]. Given that residential buildings are significant contributors to global carbon emissions, there is great interest in creating a low-carbon residential sector in Europe [3]. The widespread deployment of advanced demand-side management strategies through demand response programs, enabling flexible energy demand in European residential buildings, is seen as a promising direction to maximize energy efficiency while meeting comfort requirements. Demand response programs focus significantly on the increased integration of low-carbon energy generation systems, such as distributed solar photovoltaics

systems [4], and the modification of natural energy usages for residential buildings in response to fluctuations in supply and demand when the reliability and security of energy network systems are compromised [5]. At the same time, end-use customers become participants in demand response programs by modulating their natural consumption patterns according to electricity prices or through corresponding payment incentives in response to control signals issued by energy network operators or aggregators [6,7].

Thus, various types of demand response scenarios/programs, including load-shifting, valley-filling, peak-clipping, and the shaping of flexible loads, have been introduced [6,8] to accurately operate and manage the available local energy resources considering outdoor weather conditions, consumption patterns, and network security. In parallel, demand response strategies that reduce a building's energy demands during stressful times of the energy network are seen as feasible approaches to harness flexible energy demand without the need for substantial investments [9]. Despite the transition towards greater energy efficiency in residential buildings, driven by the widespread adoption of demand response programs, incomplete and significant improvements are still ongoing in this direction. However, there is a strong need to understand the baseline demand trends (i.e., so-called demand response baselines) for energy in residential buildings on the side of end-use customers/occupants and the interaction with the energy network. This requires providing an accurate estimation of demand response baselines, which would be consumed by end-use customers in the absence of demand response programs [10]. Demand response baselines serve as a fundamental reference point for measuring, optimizing, and assessing the potential reduction in energy demand during response events [11]. In contrast, demand response baselines in buildings are distinguished by highly fluctuating with non-linear characteristics due to the nature of consumption, which depends on occupancy, the culture of a particular building, the working schedule of each building, and outdoor weather conditions [12]. Thus, developing a data-driven learning framework to characterize baseline demand patterns and provide accurate estimates of demand response baselines, enabling the calculation of energy reductions in the context of residential buildings, is crucial [13,14].

In response, energy demand forecasts are a pivotal component of demand response programs to investigate the effectiveness of demand response scenarios and maximize their benefits. Specifically, accurate forecasts of energy demand in the short term and very short term can be employed to address different types of challenges at both the building level and the energy network level [15]. Common building-level challenges that can be addressed include tracking progress in energy efficiency improvements and defining abnormal behaviors and deviations in expected energy usage patterns, which enable the detection of potential energy losses, breakdowns, and inefficiency within the building's systems [16,17]. Energy network-level challenges are short-term optimal scheduling and identification of the optimal energy flow to meet expected demand [18], facilitating increased integration of low-carbon energy sources into the energy network [19], and demand response flexibility optimization by tracking the improvement progress in energy reductions [20]. In demand response contexts, accurate short-term forecasts of demand response baselines would be utilized by aggregators (intermediaries between end-use consumers and energy utility suppliers) to support the fair compensation of participating households in demand response programs [6]. Accurate demand response baselines can also serve as essential information for resource planners and energy system operators interested in implementing demand response programs with high effectiveness [14]. However, how to develop an accurate and reliable data-driven framework that can fulfill the above-gap remains a difficult task [21]. Therefore, this study aims to address this research need by developing a short-term forecasting framework for demand response baselines based on time-series data. This is essential for any optimal demand response strategy, particularly in large-scale residential neighborhoods that would interact with intermittent renewables (i.e., low-carbon energy sources, such as distributed solar photovoltaic systems).

2. Literature Review

Over the past few years, several research studies have been devoted to accurately estimating demand response baselines at both the individual customer-level and the aggregate-level, as summarized in Table 1. Various methods have been employed, which can be categorized into statistical and traditional machine learning methods. Concerning statistical methods, Ghasemi et al. [22] and Wijaya et al. [23] introduced the averaging-based method (XofY method), which is based on historical datasets to investigate their effectiveness in providing accurate estimates of the demand response baselines for 32 industrial and 782 residential customers in Iran and Switzerland, respectively. Despite the importance of this work, the main limitation of these methods is that they tend to provide a simplified and less accurate representation of historical energy demand data and may not accurately capture the nuances and variations in demand patterns, which can lead to sub-optimal estimates for demand response baselines. Similarly, Zhang et al. [24] and Wang et al. [25] proposed using residential consumption of non-participants in demand response programs to estimate baselines of demand response participants. The problem with such a practical approach is that there must be reference buildings with similar characteristics that do not participate in demand response actions. Furthermore, this approach becomes problematic under frequent demand response actions. In the same context, the authors in [10,26–30] presented statistical regression with external inputs, such as weather variables as predictive factors, to perform predictions of demand response baselines. The results revealed that statistical regression models have significant potential to provide accurate estimates of demand response baselines. However, a drawback of statistical regression-based models is inadequately quantified uncertainty in predicting energy demand baselines due to their inability to measure non-linear relationships between energy demand and relevant influencing factors such as consumer behaviors and ambient weather conditions [31].

Table 1. Distribution of the reviewed studies according to method, study scale, data type, and time horizon for estimation.

Method	Scale	Temporal Granularity	Data Size	Time-Horizon for Estimation	Reference
Statistical regression	138 residential customers	Sub-hourly	6 months	2-day	Ziras et al. [10]
SVR	Four office buildings	Hourly	12 days	9 am–5 pm (9 h)	Chen et al. [14]
Averaging-based	32 industrial customers	Hourly	50 days	--	Ghasemi et al. [22]
Averaging-based	782 residential customers	Hourly	12 months	--	Wijaya et al. [23]
Cohort-based	6427 residential customers	Hourly	122 days	0 am–23 pm (24 h)	Zhang et al. [24]
Cohort-based	736 residential customers	Hourly	--	3-day, 3-week, 3-month	Wang et al. [25]
Statistical regression-based	33 commercial buildings	Hourly	16 months	3-day, 5-day, 10-day	Coughlin et al. [26]
Statistical regression-based	29 commercial offices	Sub-hourly	16 months	4, 7, 10 months	Granderson and Price [27]
Statistical regression-based	17 office buildings	Hourly	12–27 months	~month	Walter et al. [28]
Statistical regression-based	Individual consumers	Sub-hourly	Winter months	~5-day	Hatton et al. [29]

Table 1. Cont.

Method	Scale	Temporal Granularity	Data Size	Time-Horizon for Estimation	Reference
Statistical regression-based	City	Hourly	10 days	~5 day	Sharifi et al. [30]
SVR	300 households	Hourly	6 months	10-h	Li et al. [32]
GMR	Individual retail store	Sub-hourly	12 months	--	Srivastav et al. [33]
GMR	441 customers	--	12 months	--	Zhang et al. [34]
MNN, SVR, RF, XGBoost	Individual house	Hourly	12 months	1 h, 1-day	Bampoulas et al. [35]
MLR, SVR, RF, CatBoost, LightGBM, ANN	20 commercial buildings	Hourly, Daily	24 months	--	Sha et al. [36]
GCN	3561 household customers	Sub-hourly	12 months	24 h	Tao et al. [37]
Self-organizing map, K-means clustering	3 multi-story buildings	Hourly	27 months	~5–10 days	Park et al. [38]
Averaging-based, ANN, MLR, PolyR	66 household customers	Hourly	4 months	24 h	Jazaeri et al. [39]
Averaging-based, exponential smoothing, regression	200 customers	Hourly, Daily	12 months	12-day	Schwarz et al. [40]

As a potential procedure to overcome the problems associated with statistical methods, several researchers have proposed a diverse combination of traditional machine learning methods to construct accurate demand response baselines. In this context, Chen et al. [14] proposed a support vector regression (SVR) method to estimate demand response baselines for office buildings, using factors such as weather and building working schedules as inputs to the SVR models. Similarly, Li et al. [32] proposed an SVR method to estimate customer demand response baselines in the presence of integrated distributed photovoltaic systems. Srivastav et al. [33] and Zhang et al. [34] proposed the development of predictive models based on the Gaussian Mixture Regression (GMR) method to characterize demand response baselines of building clusters. However, the GMR method has difficulties in processing time series data and requires the use of complex algorithmic models to calculate demand response baselines. Bampoulas et al. [35] compared the performance of RF (random forests), MNN (multilayer neural networks), SVR, and XGBoost (extreme gradient boosting) methods in providing accurate estimates of residential energy demand response baselines. Similarly, Sha et al. [36] developed six types of predictive models based on multiple linear regression (MLR), SVR, RF, CatBoost, LightGBM (light gradient boosting machine), and ANN (artificial neural network) to improve the calculation of demand response baselines for commercial buildings over the next 24 h. Tao et al. [37] proposed a graph convolutional network (GCN) method to improve the estimation of aggregated demand response baselines, as its performance was compared with that of their counterparts from SVR, MLR, and averaging methods. This is in addition to the other methods proposed in [38–40] to improve the estimation of demand response baselines in buildings.

Notwithstanding the effectiveness of some data-driven machine learning methods in estimating demand response baselines, as mentioned above, these methods require substantial improvement by considering more external factors, such as occupancy and indoor

environmental conditions, which can be difficult to acquire in the context of large-scale neighborhoods and district buildings. In addition, implementation strategies for demand response in buildings necessitate high-resolution forecasts (from hourly to daily), leading to the need for developing accurate forecasting models [12,41]. As demand patterns fluctuate randomly, inaccurate estimates of demand response baselines can lead to significant errors when aggregated to determine the total energy reductions caused by the activation of response events [37]. In the face of such challenges, deep learning methods have brought the issue of reliable and accurate estimates in short-term energy forecasting studies in the building sector back into the spotlight and have received considerable attention in recent years. Researchers have pointed out the great potential of these methods in providing accurate results for building energy demand forecasting [42,43]. Accordingly, this study introduces the deep learning approach as a potential candidate to improve the accuracy of residential demand response baseline estimates over a short-term forecast horizon. The aim is to develop a robust and reliable deep learning-based data-driven framework and evaluate its performance, considering different residential energy demand profiles in neighborhood buildings, in order to provide accurate estimates for aggregated demand response baselines over multiple time forecast horizons, not exceeding 24 h. To the author's knowledge, no previous studies have applied bidirectional long-short-term memory (BiLSTM) and gated recurrent unit neural network (GRU) to estimate aggregated demand response baselines in a neighborhood context.

Contribution of the Study

In light of the study's objective and considering the strengths and weaknesses identified in the literature, the main contribution of this work is as follows:

- A data-driven framework is proposed to identify the most effective deep learning methods in providing accurate estimates of residential demand response baselines over various time-horizons, not exceeding 24 h. This provides a novel insight for a deeper understanding of the forecasting characteristics exhibited by different data-driven models.
- Verify the change in model performance during the evaluation phase by considering the demand response baseline profile according to different aggregation levels of residential units and other input features. This investigation is essential for understanding the different behaviors of forecasting models and the importance level of input features.
- The performance of the deep learning models is compared with that of the traditional statistical and machine learning models developed in this work, considering both the type of forecasting model and the expected margin of error. This comparison helps to identify the strengths and limitations of each model and method in the context of short-term demand forecasts for residential demand response baselines.

The rest of this paper is organized into four main sections as follows: Section 3 presents the deep learning methods proposed in this work. Section 4 describes the methodology of this work developed from the previously mentioned methods. Next, the findings obtained from this work are presented and discussed in Section 5, and finally, the main conclusions and potential future developments of this work are drawn in Section 6.

3. Proposed Forecasting Methods

Basic concepts of the deep learning methods proposed in this work are outlined as follows.

3.1. Deep Neural Networks

Deep neural networks (DNNs) are one of the most used and popular neural network architectures in energy demand forecasting, commonly known as multilayer perceptron neural network models, due to the inclusion of more hidden layers. They are widely regarded as a robust and effective tool for solving complex problems, including clas-

sifications and forecasting tasks, due to their capacity to learn and represent intricate non-linear relationships between input and output data [44]. To achieve this, the basic structure of DNNs consists of three types of successive layers: input layers, hidden layers, and output layers, as shown in Figure 1a. The back-propagation algorithm is used to train DNNs [45]. As shown in Figure 1a, the input layers receive input signals $\mu(t - \tau_1)$, $\mu(t - \tau_2)$, $\mu(t - \tau_3)$, \dots , $\mu(t - \tau_n)$, where $\tau_1, \tau_2, \tau_3, \dots, \tau_n$ are constants. The summation of control signals and the system's outputs at time t are represented by $u(t)$. The weights that connect the input layers to the hidden layers are represented by $w_{11}^1, w_{12}^1, \dots, w_{1n}^1$ for the first neuron, $w_{21}^1, w_{22}^1, \dots, w_{2n}^1$ for the second neuron and $w_{31}^1, w_{32}^1, \dots, w_{3n}^1$ for the third neuron. The weights associated with hidden layer neurons with neuron q are denoted by $w_{h1}^1, w_{h2}^1, \dots, w_{qn}^1$, where q denotes the number of neurons. The weights that connect hidden layers to an output layer are represented by $w_{21}, w_{22}, \dots, w_{2q}$ [46,47]. These weights and connections between different layers enable the DNN to learn and make reliable and accurate predictions based on the input data. Thus, DNNs have gained significant attention in time-series forecasting of building energy demand, with residential buildings receiving a substantial part of this attention [48,49].

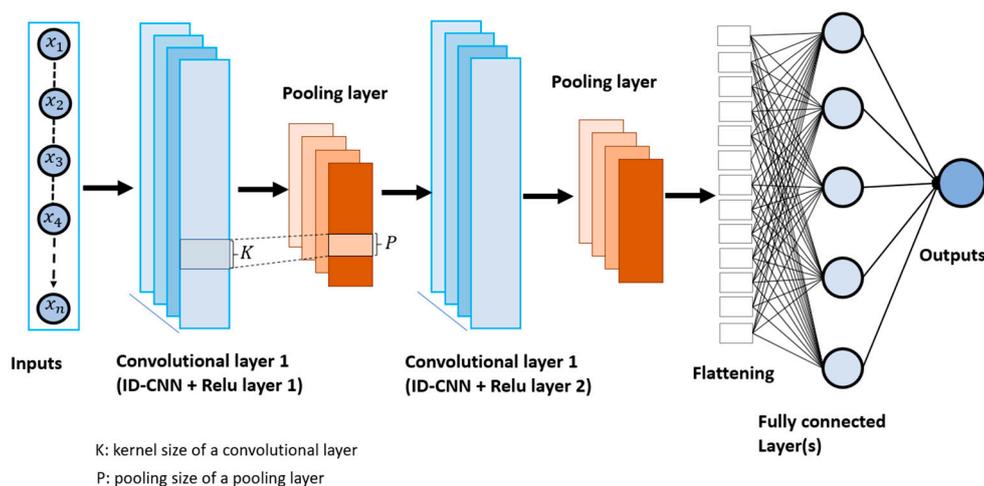
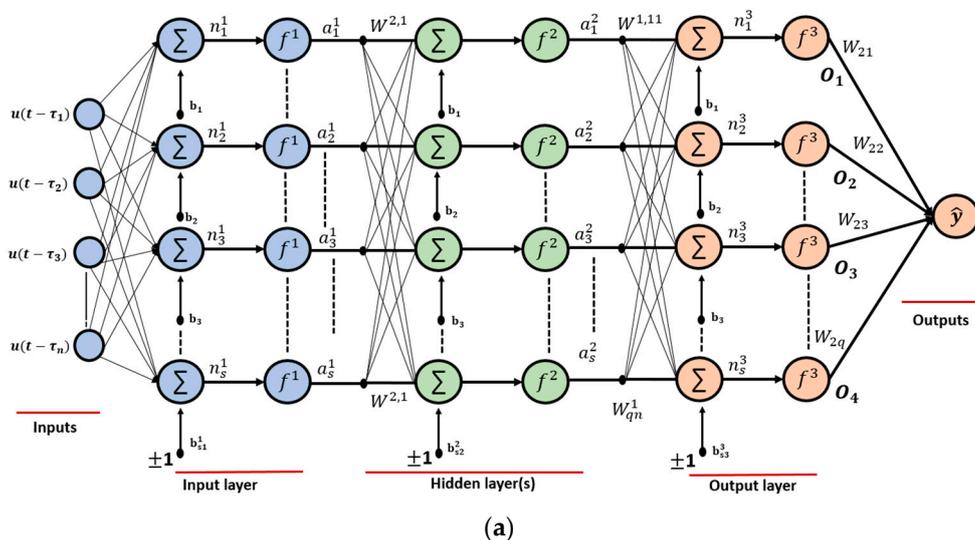


Figure 1. Architecture of (a) deep neural network (DNN) and (b) convolutional neural network (CNN) methods.

3.2. Convolutional Neural Networks

Convolutional neural networks (CNNs) are a unique class of advanced neural network methods that hierarchically perform convolutional operations on input time-series data. In recent years, CNNs have received increasing attention in building energy demand forecasting [48,50] due to their ability to capture time-series dependencies. The typical structure of CNNs consists of five layers, namely input, convolutional, pooling, fully connected, and output layers [51]. They are characterized by their capability to process and transform time-series datasets utilizing three building blocks. These include (1) convolutional layer, which implements two types of operations. (a) The first of these operations are convolutional operations, which require two components, called the kernel and the time-series data. The kernel implements convolution on the time series by moving from the beginning to the end of a series (i.e., in one direction), and the dot product between the kernel and corresponding parts of the series is computed. (b) The other operations involve the non-linear activation being applied to the final output of convolutional operations. The other building blocks are (2) a pooling layer to maintain the stability and prevent overfitting of the model and (3) a fully connected layer to perform the same duties in the conventional neural networks [52].

In this work, a 1-D convolutional neural network (Conv1D) was utilized to extract the features from the time series data for demand response baselines in residential buildings. This network applies sliding convolutional operations along the sequence of one-dimensional time series data [53]. The proposed Conv1D consists of five foundational layers: the convolutional layer, the pooling layer, the fully connected layer, the dropout layer, and the ReLU (relu) correction layer. The CNN network performance depends on the parameters of these layers. They consist of several components, including the number of filters in the layers, filter size, padding, stride, and batch size. Figure 1b displays the CNN architecture used in this work.

3.3. Recurrent Neural Networks

Recurrent neural networks (RNNs) are an advanced class of neural networks designed to overcome the disadvantages of a traditional neural network in accounting for temporal correlations and dependencies in data sequences [54]. They are also distinguished from other deep learning neural network architectures by their recurrent connections, which enable them to memorize the information from previous outputs and incorporate it into the computation of the current result [55]. Thus, the impact of recurrent neural network models has been remarkable in many disciplines, including energy demand forecasting, where the temporal order in the dataset is a fundamental feature in model design. RNNs are typically networks composed of standard recurrent connection cells, hidden states, and input and output layers. The input nodes in RNN models have no incoming connections, and the output nodes have no outgoing connections, while the hidden state nodes have both incoming and outgoing connections [52]. Each time, the RNN model updates the information in its memory according to the following Equation (1).

$$h_t = fc((W_h h_{t-1}), x_t) \quad (1)$$

where h_t is the current hidden state at time t ; fc is the activation function, typically the hyperbolic tangent (tanh) or the rectified linear unit (relu); W_h is the weight matrix for the recurrent connections; h_{t-1} is the previous hidden state at time $t - 1$; and x_t is the input at time t .

The architecture of the RNN is exhibited in Figure 2a. Each node represents a neuron for a single timestep. $W1$ represents the connection weight for inputs, $W2$ signifies the self-connection weight for each neuron, and $W3$ denotes the connection weight for outputs. The input data sequence is processed sequentially within the network based on time steps, and the weight coefficients are reused in a recycling fashion. The training process of an RNN model includes a forward pass and a backward pass. The forward pass of an RNN model mirrors that of a single-hidden-layer multilayer perceptron, except that the hidden

layer in an RNN model receives activations from both the current external input and the hidden layer activations from the previous timestep [56]. The process of computing weight derivatives for an RNN during the backward pass is referred to as “backpropagation” through time. The advantage of RNN models in time-series forecasting is that they can predict not only the next time step but also multiple future time steps, making them versatile for different forecasting horizons.

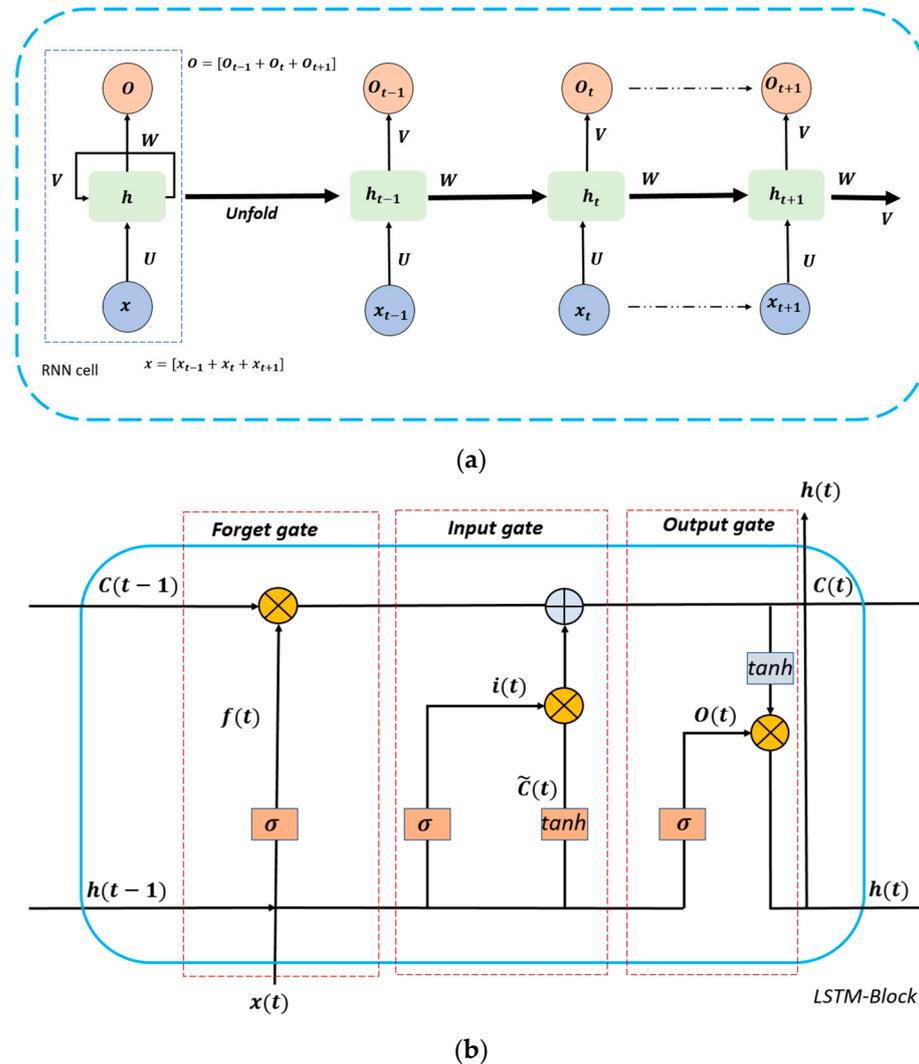


Figure 2. Architecture of (a) recurrent neural network (RNN) and (b) long short-term memory (LSTM) methods.

3.4. Long Short-Term Memory

Long short-term memory (LSTM) is an upgraded variant for RNN architectures designed to overcome the vanishing gradient and gradient explosion problems in the long-sequence training process. For this reason, LSTMs can be trained utilizing the time-series dataset to make predictions for the future energy demand of buildings, each time utilizing the historical dataset processed by the LSTM cells. Typically, the architecture of LSTM is composed of frequently interconnected subnetworks called memory blocks, as shown in Figure 2b. Each block consists of a forget gate, an input gate, an output gate, and one or more self-connected memory cells [57]. During the training of LSTM models, the LSTM gates facilitate the long-term storage and retrieval of information within the memory cells, effectively addressing the problem of vanishing gradients [58]. For instance, if the input gate remains closed (i.e., with an activation close to 0), the cell’s activation persists and is

not overwritten by new inputs entering the network. Consequently, this information can be retained and made accessible to the network at a later point in the sequence by simply opening the output gate.

Significantly, the values of the forget gate and input gate are influenced by both the previous hidden state and the current input. The operation of these units in LSTM is parameterized as follows.

$$I_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \quad (2)$$

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \quad (3)$$

$$G_t = \tanh(W_{gx}x_t + W_{gh}h_{t-1} + b_g) \quad (4)$$

$$C_t = (f_t \times C_{t-1}) + (I_t \times G_t) \quad (5)$$

$$O_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \quad (6)$$

$$H_t = O_t \times \tanh(C_t) \quad (7)$$

where I_t , f_t , G_t , and O_t are the input, forget, update, and output gate activations at time t . The notations σ and \tanh represent non-linear activation functions that take values in the range the $[0, 1]$ and $[-1, 1]$, respectively. W^* and b^* are weight matrices and bias vectors specific to each gate, while C_t is the cell state at time t .

3.5. Bidirectional Long Short-Term Memory

Bidirectional long short-term memory (BiLSTM) is an extension of the LSTM network to capture dependencies in both the past and future contexts of a given data point. With respect to prediction based on time series data, the multiple sequences of energy loads are highly time-dependent, and loads at any point in time are significantly correlated with loads at its previous and subsequent points in time, requiring a deeper temporal-feature-extractor [59]. Compared to the unidirectional-state-transmission in LSTMs, BiLSTM consists of two LSTM layers, namely the forward LSTM and the backward LSTM, as shown in Figure 3a, and the output is jointly identified by the states of these two LSTMs [12,60]. This configuration achieves bidirectional time series feature extraction, which can fully exploit the temporal correlations of energy load sequences. By fusing two bidirectional LSTM layers, the output outcome is computed as follows.

$$\vec{h}_t = LSTM(x_t, \vec{h}_{t-1}) \quad (8)$$

$$\overleftarrow{h}_t = LSTM(x_t, \overleftarrow{h}_{t+1}) \quad (9)$$

$$O_t = \sigma(\vec{W}_{oh} \vec{h}_t + \overleftarrow{W}_{oh} \overleftarrow{h}_t + b_o) \quad (10)$$

Here, $LSTM$ represents the LSTM function, while \vec{W} and \overleftarrow{W} are the weight matrices of the forward and backward LSTMs for the computed output outcome.

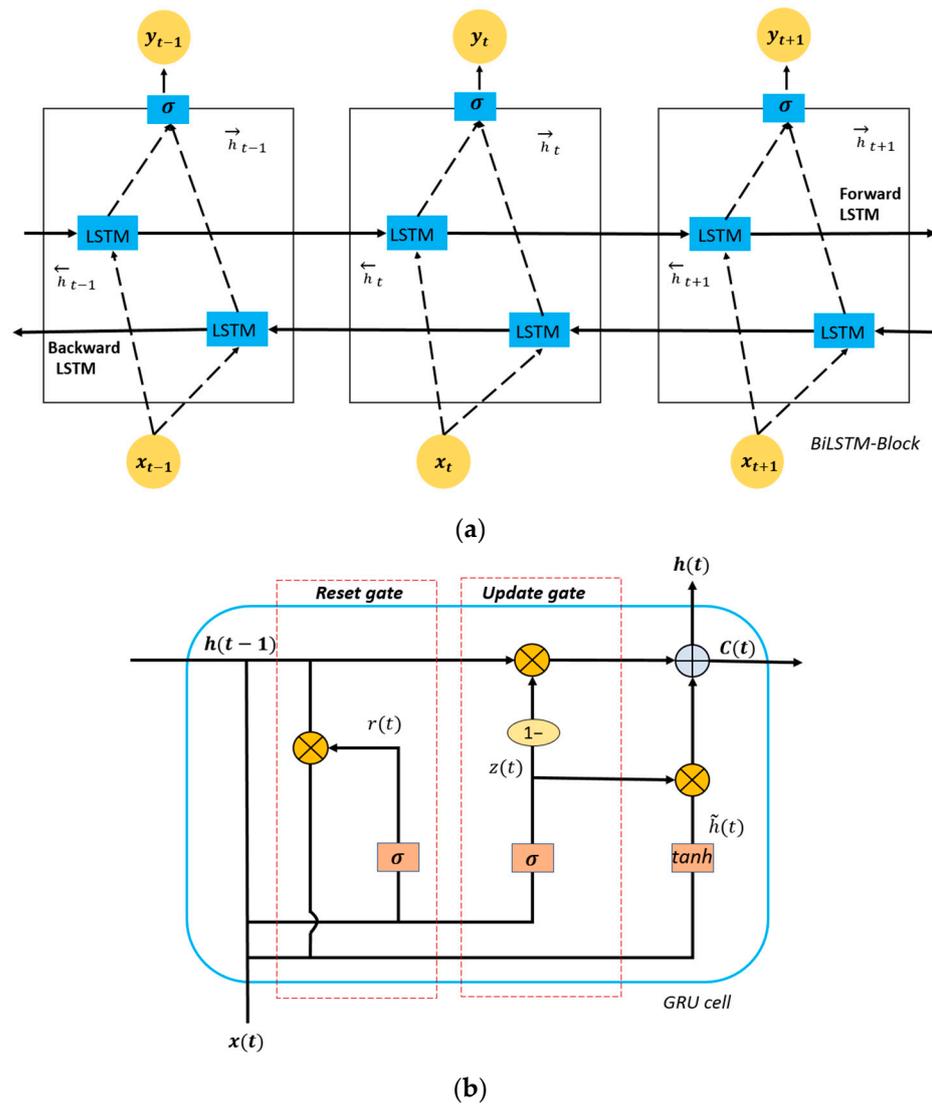


Figure 3. Architecture of (a) bidirectional long short-term memory (BiLSTM) and (b) gated recurrent unit (GRU) methods.

3.6. Gated Recurrent Units

Gated recurrent units (GRUs) are a simplified alternative to LSTMs, designed to alleviate the computational burden associated with the large number of parameters in the LSTM network [61]. Compared with LSTMs, the GRU architecture has only two gates, namely the update gate and the reset gate (see Figure 3b) [62]. Typically, the update gate dictates what to discard from the memory of the previous unit, and the reset gate guides the fusion of the new input with the last memory [63]. Based on both the previous output h_{t-1} and the current input x_t , the functioning principle of the GRU cell is listed as follows.

$$Z_t = \sigma(W_{zh}h_{t-1} + W_{zx}x_t + b_z) \quad (11)$$

$$r_t = \sigma(W_{rh}h_{t-1} + W_{rx}x_t + b_r) \quad (12)$$

$$\bar{h}_t = \tanh(W_{hh}[r_t \odot h_{t-1}] + W_{hx}x_t + b_h) \quad (13)$$

$$\tilde{h}_t = (1 - Z_t) \odot h_{t-1} + Z_t \odot \bar{h}_t \quad (14)$$

As exhibited in Equations (11)–(14), Z_t , r_t , \bar{h}_t , and \tilde{h}_t are the update gate, reset gate, output candidate vector, and the shared memory at time t , respectively, while \odot is an

element-wise product. As demonstrated in Equations (11)–(14), the shared memory \tilde{h}_t passes through various time steps to encode new information while discarding memories that are no longer relevant in terms of timing. Hence, the shared memory stores significant information over an extended period. r_t defines how much information h_{t-1} should be retained. A smaller value for r_t (close to 1) denotes that more information h_{t-1} is retained in \tilde{h}_t . Then, the update gate Z_t defines how much information h_{t-1} should be discarded. A bigger Z_t indicates that more information h_{t-1} was not discarded, whereas a smaller Z_t means that a significant amount of information h_{t-1} was ignored.

4. Methodology

As mentioned above, the primary objective of this study is to investigate the potential of deep learning methods in providing an accurate estimation of residential demand response baselines. This includes data preprocessing, input feature selection, hyperparameters, development of baseline models, forecasting, and evaluations. Therefore, the methodology of this study includes several steps consisting of the following:

- Obtaining a representative profile of the baseline residential energy demand (i.e., demand response baselines) in building clusters, which vary in terms of household space size and occupant behavior in the absence of response events.
- Pre-processing to understand how the demand response baselines of residential buildings correlate with potential input features, obtaining an input feature selection based on the Pearson Correlation Coefficient (PCC) technique [64,65].
- Training the models on a dataset composed of all the input features processed during the feature engineering stage and those determined as significant inputs by PCC in the input feature selection stage, with the error measured by performance indicators in the validation and evaluation stages.
- Finally, the trained models and the specified input features are used to predict the future demand response baselines of residential units over various time horizons, not exceeding 24 h, and the results are utilized to assess the performance of each forecasting model.

The methodology can also be divided into the steps described as data acquisition (i.e., residential building and energy demand data inventory) and data preprocessing and input feature selection, as shown in Figure 4. The subsequent sections provide further details on the model training process, which involves the utilization of a rolling window, the hyperparameter selection, and performance evaluations.

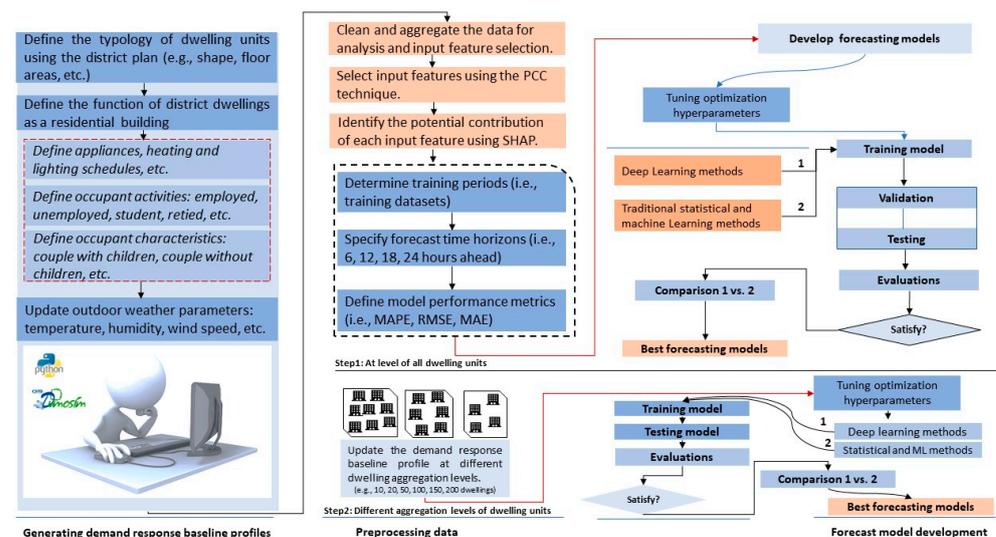


Figure 4. Research methodology used in this work, including simulations and model development.

4.1. Residential Building and Energy Demand Data Inventory

To investigate the performance of the proposed forecasting methods, 337 dwellings in the Atlantech of La Rochelle City, France, were selected as a typical case study. In order to secure training data for developing forecasting models, the baseline energy demand profile of 337 dwelling units was generated by simulating the real dwellings in DIMOSIM (District MOdeller and SIMulator), a Python-based urban building energy simulation platform (more details in [66,67]), the numeric datasets were saved in comma-separated values (CSV) files. The reason for this is that the measured data for the residential energy demand of all dwelling units, which must represent real demand response baselines in the absence of demand response events, are not currently available. Therefore, the DIMOSIM models were used to generate meticulously prepared datasets that accurately represent the behavior of existing dwellings exposed to the typical climatic conditions of Atlantech in the city of La Rochelle.

These datasets cover all aspects of dwelling units, with a particular focus on space heating, lighting, and electric appliances to represent the smart-meter readings of dwellings. They consist of 674 columns representing the most important information on the thermal behaviors of the dwelling and the different energy demand patterns during the heating season. In terms of size, the dataset contains a considerable amount of information due to both the five-month simulation duration (from November to the end of March) and the 10 min sampling interval. In this study, energy demand simulations for dwelling units during the heating season were carried out because of the significance of demand response programs in reducing non-essential electricity consumption during peak heating hours while at the same time providing economic benefits and promoting energy efficiency. The results of the DIMOSIM models are also compared with external references (for more details, see Ref. [68]), with DIMOSIM showing good agreement with the other tools in terms of energy production. As depicted in Figure 5, the black solid lines represent the average demand response baseline of all dwelling units, while both the red and blue solid ones are the metered heating loads and lighting and appliances during the heating months.

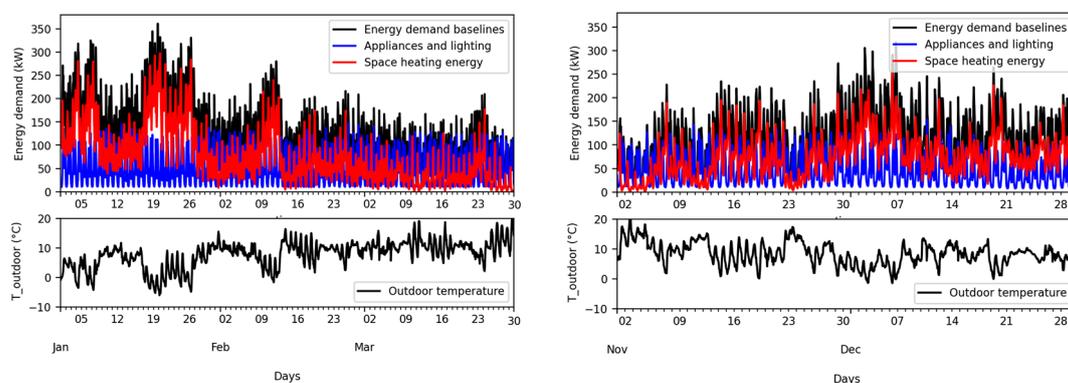


Figure 5. Hourly baseline energy demand behaviors for dwelling units over the days of the five heating months, as the energy demand baselines are the average for space heating, appliances, and lighting.

In terms of dwelling characteristics, the floor area of the dwelling units ranges from 38 m² up to 225 m², with an average size of 71 m². The annual energy demand of each dwelling unit is approximately 105.85 kWh/m². For space heating, the heating system consists of air-to-water heat pumps installed in each dwelling unit. In addition, each heat pump is equipped with its dedicated thermostat, enabling control over individual zones. The heat pump equipment is variable speed, and its coefficient of performance (COP) depends on the polynomial regression of the nominal performance coefficient to estimate the thermal power output as a function of both the radiator temperature (i.e., the sink) and the ambient temperature (i.e., the source). For the population and occupancy characteristics, each dwelling unit is occupied by either a couple with or without children or a single adult

living alone. The dwelling occupancy characteristics include people who are employed, unemployed, students, or retired, as shown in Table 2.

Table 2. Occupancy characteristics of the targeted dwelling units in Atlantech in the city of La Rochelle.

		Min	Max	Mean	Standard Deviation
Household composition (person/m ²)	Couple with children	0.004	0.024	0.014	0.004
	Couple without children	0.008	0.026	0.014	0.003
	Single	0.008	0.026	0.016	0.004
Occupant status (person/m ²)	Employed	0.004	0.074	0.025	0.014
	Unemployed	0.011	0.073	0.031	0.018
	Stay-at-home	0.004	0.018	0.012	0.003
	Student	0.009	0.063	0.029	0.011
	Retired	0.008	0.059	0.021	0.009
Average lighting and electric equipment energy density (kW/m ²)		0.019	1.241	0.290	0.203
Heating system operation	Comfort setpoint (°C)	17.3	22.40	19.83	1.26
	Use of setback (%)	-	-	80	-
	COP of heat pumps (-)	1.71	4.95	3.33	0.81

4.2. Data Preprocessing

Since the baseline energy demand of the dwelling units involves separate profiles for space heating, lighting, and electric appliances in each dwelling, the initial step of data preprocessing was to appropriately integrate the dataset of demand response baselines for all dwelling units to create a unified dataset for analysis and model training. In the first stage, the data for space heating, lighting, and electric appliances are aggregated together for each dwelling. The next step is to determine the total and hourly energy demand for all dwelling units, as shown in Figure 6. On the other hand, the measured outdoor weather data collected at 1 h intervals in this study is not aligned with the generated dataset of demand response baselines, which had a 10 min sampling interval. Therefore, the data are reorganized using the technique of resampling and indexing time series to form unified readings. The reorganized dataset includes the data of demand response baselines, a total of 52,555 samples. This step is of great importance in order to facilitate the estimation of the correlation between demand response baselines for dwelling units and other related input features, as explained in Section 4.3. In this context, other new features (e.g., hour of the day or day of the week) are derived using a time-based feature engineering/temporal feature engineering technique. Feature engineering is used in this work due to its importance in the development of forecasting models based on time series data, as it directly affects the performance and accuracy of these models.

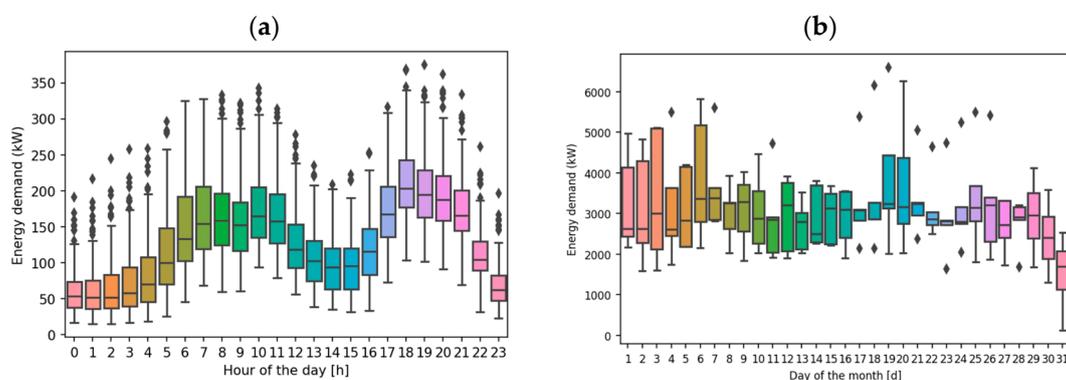


Figure 6. Energy demand distributions (demand response baseline profile) of dwelling units during the five heating months: (a) hourly energy demand and (b) total daily energy demand.

The final datasets, which included the hourly dataset for demand response baselines, weather factors, and other input features, were obtained. These datasets have not been subjected to the normalization process [0, 1] to ensure that all models work with the original data. They go through the input selection process to select the final inputs to the forecasting models and to understand how the future baseline energy demand of residential buildings correlates with the possible input features, as explained in Section 4.3.

4.3. Selection of the Input Features

Using the feature engineering technique results in datasets that have multiple attributes associated with the considered demand response baseline profile. Since there are multiple associated characteristics, it is essential to use the appropriate method to determine the importance of each of them to the demand response baselines of the dwelling units. Along with the measured outdoor weather factors (e.g., the outdoor temperature and direct solar radiation), the impact of sixteen (16) independent input features is also considered. Nine input features (9) capture previous energy demand patterns, encompassing the past 1 h, the past 2 hours, the past 3 hours, the past 24 h, the past 48 h, the past 72 h, and so on. The other seven input features (7) relate to the working schedule factors of the dwelling, such as the hour of the day, the day of the week, the day of the month, the day of the year, the week of the month, and the month of the year. In this context, both the Pearson Correlation Coefficient (PCC) and the Shapley Additive Explanation (SHAP) techniques are used to minimize redundant or useless input features and to identify the most important variables for the forecasting models.

PCC was used to estimate the correlation coefficients (R) between the average energy demand of all dwelling units in the district and each input feature, as described in Equation (15).

$$R = \frac{\sum_{i=1}^n (x_i - \bar{x}) + (y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (15)$$

where x_i and y_i represent the actual values, \bar{x} and \bar{y} donate the average values, and n is the number of observations. The correlation coefficient of PCC takes a value between (+1) and (−1), as the highest absolute value of PCC translates to a parameter closely related to the future energy demand of dwelling units. Figure 7a displays the PCC values between the baseline residential energy demand (i.e., the demand response baselines) and the most meaningful input features in descending order when considering the whole evaluation. As shown in Figure 7a, the factors associated with the previous energy demand patterns show strong positive correlations with the demand response baselines of the dwelling units. Weather-related outdoor temperatures also show strong negative correlations, while the other factors have relatively different correlation values. However, these factors with low PCC values were selected as input features for the forecasting models because deep learning-based data-driven models have a high ability to detect non-linear and linear relationships between the energy demand baselines of the dwelling units and other relevant influencing factors. Based on this, the use of small correlation values of the PCC ensures that the low-degree correlation is not neglected in model training.

As mentioned above, SHAP is also used to identify the potential contribution of each input feature to the forecasting model. Figure 7b shows the contributions of each input feature in descending order. It is important to note the importance of factors related to past energy demand patterns, outdoor temperature, and the hour of the day in forecasting baseline values of residential demand response over very short-term periods, not exceeding 24 h. Together with PCCs, these findings demonstrate the importance of including previous input features, including those related to previous energy demand pattern-related ones, for accurate forecasting of demand response baselines. The selection of the three (3) historical energy demand-related input features is based on a comprehensive analysis of factors that significantly affect energy demand forecasting. These features have been chosen to capture the diverse aspects of historical energy consumption patterns, ensuring a robust

and accurate forecasting model. Accordingly, these nine factors out of all input features were selected as the final input features for all forecasting models, as described in Table 3.

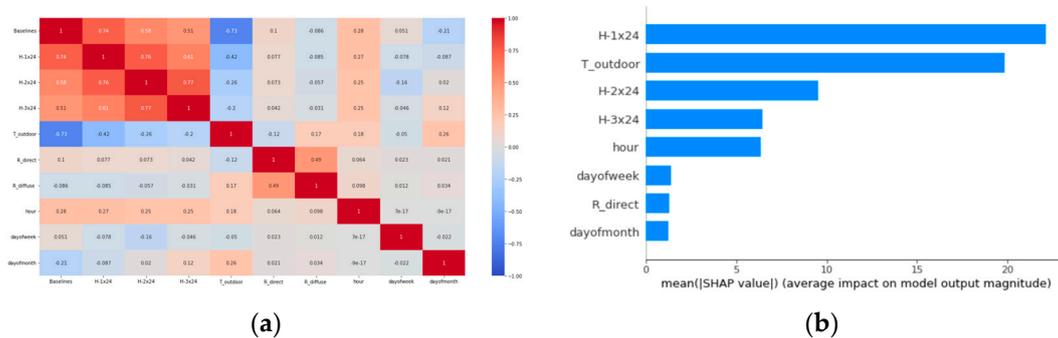


Figure 7. Example of the input feature importance estimated by the PCC technique and determined by SHAP tests using values of demand response baselines for all dwelling units.

Table 3. Input features selected as predictors for forecasting models.

Input Feature (Factor)	Abbreviation	Unit
Previous energy demand patterns	24 h	H-1×24
	48 h	H-2×24
	72 h	H-3×24
Weather	Outdoor temperature	T_outdoor
	Director solar radiation	R_direct
	Diffuse solar radiation	R_diffuse
Dwelling working schedule (occupancy)	Hour of the day	hour
	Day of the week	dayofweek
	Day of the month	dayofmonth

4.4. Forecast Model Development

The development of deep learning-based forecasting models explained in Section 3, involves two fundamental steps: hyperparameter tuning and post-training. It is important to note that the performance of these models is compared with that of classic and ensemble models based on traditional statistical and machine learning methods used in previous literature [69–72]. The classic models included support vector regression (SVR), Autoregressive integrated moving average (ARIMA), multiple linear regression (MLR), Lasso regression (Lasso), ridge regression (Ridge), polynomial regression (PolyR), Bayesian regression (Bayesian), kernel ridge regression (KernelR), and stochastic gradient descent regression (SGDReg) algorithms. The tree-based ensemble machine learning models included extreme gradient boosting (XGBoost), light gradient-boosting machine (LightGBM), gradient boosting (GB), random forest (RF), Adaptive Boosting (AdaBoost), Bagging (Bagging), and categorical gradient boosting (CatBoost) algorithms. Both classic and ensemble algorithms are among the most popular traditional statistical and machine learning methods in building energy analysis applications (see Refs. [69–71]), which have become specifically data-driven methods for predicting, benchmarking, and mapping baseline energy demand in buildings [72].

4.4.1. Hyperparameter Tuning

Tuning the appropriate hyperparameters is a critical step in the development of accurate data-driven forecasting models and can have a considerable impact on convergence speeds and generalizability. In this work, the controlled-variable method, relying on empirical expertise, is utilized in the experiments to optimize the choice of hyperparameters. To determine the optimal architecture of the proposed models, this work referred to references [73–75] and identified the practicable scopes and the optimization scope for

hyperparameters, as shown in Table 4. In terms of deep learning models, forecasting based on BiLSTM and CNN is presented here as a typical example. First, with default BiLSTM parameters (activation function = relu, optimizer = Adam, loss function = RMSE, batch size = 32, verbose = 2, learning rate = 0.0001, epochs = 30) and the default parameters of CNN (activation function = relu, optimizer = Adam, loss function = RMSE, filters = 64, kernel size = 2, pool size = 2, learning rate = 0.0001, verbose = 2, epochs = 30), the number of hidden units in each hidden layer was fixed. Second, the learning rate, epochs, batch size, filters, and kernel size were adjusted. For example, the RMSE value of BiLSTM models decreases when the learning rate is set to 0.01, and the number of epochs is set to 200 or 300. For CNN models, the RMSE value decreases when the learning rate is set to 0.001, and the number of epochs is set to 300 or 500. Third, in cases where there was a tendency for overfitting to occur, dropout was implemented based on empirical expertise. Dropout is a useful regularization technique to control the proportion of randomly selected neurons during each training iteration. This helps to prevent co-adaptation of neurons, making the network more robust and reducing overfitting. Finally, to minimize redundancy in the process of defining hyperparameters for optimization, parameters of the same model type in the same case were kept consistent, and adjustments were only made to those parameters that prevented overfitting.

Table 4. Hyperparameter tuning information for forecasting models based on deep learning and classic and ensemble methods.

Model	Hyperparameters	Search Scopes	Optimal Values		
			Case I	Case II	
Classic	Ridge	Regularization_coefficient for a norm of the weight_vector (l_2)	{3, 6, 10, 13, 20, 50, 100, 200}	100	50
	Lasso	Regularization_coefficient for a norm of the weight_vector (l_1)	{0.001, 0.01, 0.1, 3, 5, 10, 20, 30}	10	3
	PolyR	Regularization_coefficient (alphas)	{ 1×10^{-6} , $1 \times 10^{+6}$, 1×10^{-3} , $1 \times 10^{+3}$, 1×10^{-2} , $1 \times 10^{+2}$ }	{ 1×10^{-6} , $1 \times 10^{+6}$ }	{ 1×10^{-6} , $1 \times 10^{+6}$ }
		Maximum number of iterations	{300, 1000, 3000}	3000	3000
	Bayesian	Regularization_coefficients (parameters: alphas, lambdas)	{ 1×10^{-6} , $1 \times 10^{+6}$, 1×10^{-3} , $1 \times 10^{+3}$, 1×10^{-2} , $1 \times 10^{+2}$, 1×10^{-1} , $1 \times 10^{+1}$ }	{ 1×10^{-1} , $1 \times 10^{+1}$ }	{ 1×10^{-1} , $1 \times 10^{+1}$ }
		Maximum number of iterations	{100, 300, 1000, 3000}	3000	1000
	SGDReg	Regularization_coefficient (alpha)	{0.001, 0.01, 1, 10}	0.1	0.1
		Maximum number of passes (epochs)	{300, 600, 1000, 3000, 6000}	3000	2000
	KernelR	Types of Kernel used in the algorithm	{str, callable, linear}	Linear	Linear
		Degree of Kernel	{2, 3, 6}	3	3
Regularization_coefficient (alphas)		{0.1, 1, 10, 30, 100}	100	30	
SVR	Types of Kernel used in the algorithm	{linear, sigmoid, rbf, poly}	rbf	rbf	
	Regularization_parameter "regressor_gamma"	{0.001, 0.01, 0.1, 1, 10}	{0.001, 0.01, 0.1}	{0.001, 0.01, 0.1}	
	Regularization_parameter "regressor_C"	{50, 60, 70, 100, 300, 1000}	{50, 60, 70}	{50, 60, 70}	

Table 4. Cont.

Model	Hyperparameters	Search Scopes	Optimal Values		
			Case I	Case II	
Ensemble	XGBoost	N_estimators	{100, 250, 500, 900}	600	500
		Maximum tree depth	{3, 6, 7, 8, 9, 10}	7	3
		Learning rate	{0.001, 0.01, 0.02, 0.03}	0.01	0.01
		Minimum loss reduction "gamma"	{0.02, 0.05, 0.1, 3}	3	3
		Subsample	{0.3, 0.6, 0.8, 0.9, 1}	0.8	0.6
	LightGBM	N_estimators	{200, 300, 800, 1000}	800	600
		Maximum tree depth	{3, 4, 6, 8, 10, 15}	10	5
		Learning rate	{0.05, 0.01, 0.03, 1.0}	0.1	0.1
		Number of leaves	{10, 60, 100, 300}	300	300
		Subsample	{0.2, 0.5, 0.6, 0.75, 1}	0.75	0.75
	GB	N_estimators (trees)	{20, 60, 200, 300}	200	200
		Maximum tree depth	{2, 4, 6, 8, 10}	2	2
		Learning rate	{0.01, 0.1, 0.2, 0.3}	0.3	0.3
	RF	N_estimators (trees)	{30, 100, 150, 200}	200	200
Maximum tree depth		{4, 5, 6, 10, 15}	15	15	
Maximum features		{sqrt, auto}	sqrt	sqrt	
Minimum number of samples		{2, 3, 4, 5, 6}	200	200	
AdaBoost	Learning rate	{0.001, 0.1, 0.2, 0.3}	0.1	0.1	
	N_estimators (trees)	{100, 200, 250, 300}	300	300	
Bagging	Minimum number of features	{0.1, 0.3, 0.6, 1, 4}	1	1	
	Maximum number of samples	{1, 3, 7, 10, 15, 30}	1	1	
	N_estimators (trees)	{200, 500, 700, 1000}	1000	1000	
CatBoost	Learning rate	{0.01, 0.03, 0.1, 1}	1	1	
	Maximum tree depth	{3, 10, 20, 30}	3	3	
Deep Learning	ANN	Number of hidden layers	{1}	1	1
		Number of neurons per hidden layer	{32, 64, 80, 100, 156}	156	100
		Number of epochs	{30, 100, 300, 600, 800}	600	600
		Learning rate	{0.001, 0.01, 0.02, 0.03}	0.001	0.001
		Dropout rate	{0, 0.1, 0.2, 0.3, 0.4}	0.2	0.2
	DNN	Number of hidden layers	{2, 3, 4, 5, 6}	2	3
		Number of neurons per hidden layer	{20, 30, 64, 80}	64	64
		Number of epochs	{30, 200, 300, 400, 600}	300	300
		Learning rate	{0.001, 0.01, 0.03, 0.04}	0.001	0.001
		Dropout rate	{0, 0.1, 0.2, 0.3, 0.4}	0.2	0.2
	CNN	Number of hidden layers	{2, 3, 4, 5, 6}	3	2, 3
		Number of neurons per hidden layer	{20, 32, 50, 64}	64	32, 64
		Number of epochs	{30, 100, 200, 300, 500}	500	300
		Learning rate	{0.001, 0.01, 0.03, 0.05}	0.001	0.001
		Dropout rate	{0, 0.1, 0.2, 0.3, 0.4}	0.2	0.2
	RNN	Number of hidden layers	{2, 3, 4, 5, 6}	2, 3	2
		Number of neurons per hidden layer	{16, 32, 64, 76}	64	32, 64
		Number of epochs	{30, 100, 300, 400, 500}	500	400
Learning rate		{0.001, 0.01, 0.02, 0.03}	0.001	0.01	
Dropout rate		{0, 0.1, 0.2, 0.3, 0.4}	0.2	0.1	
LSTM	Number of hidden layers	{2, 3, 4, 5, 6}	2, 3	2	
	Number of neurons per hidden layer	{16, 32, 64, 70}	32, 64	50	
	Number of epochs	{30, 200, 300, 600}	600	300	
	Learning rate	{0.001, 0.01, 0.02, 0.03}	0.01	0.01	
	Dropout rate	{0, 0.1, 0.2, 0.3, 0.4}	0.2	0.1	

Table 4. Cont.

Model	Hyperparameters	Search Scopes	Optimal Values		
			Case I	Case II	
Deep Learning	GRU	Number of hidden layers	{2, 3, 4, 5, 6}	2, 3	2
		Number of neurons per hidden layer	{20, 40, 50, 64}	64	64, 40
		Number of epochs	{30, 100, 300, 400, 500}	500	300
		Learning rate	{0.001, 0.01, 0.02, 0.03}	0.01	0.01
		Dropout rate	{0, 0.1, 0.2, 0.3, 0.4}	0.2	0.1
	BiLSTM	Number of hidden layers	{2, 3, 4, 5}	2, 3	2
		Number of neurons per hidden layer	{20, 32, 50, 64}	32, 64	32, 64
		Number of epochs	{30, 100, 200, 300, 400}	300	200
		Learning rate	{0.001, 0.01, 0.02, 0.04}	0.01	0.01
		Dropout rate	{0, 0.1, 0.2, 0.3, 0.4}	0.2	0.1

Notes: Case I and Case II refer to forecasting models with demand response baseline profiles with all dwelling units and different aggregation levels of dwelling units. Adam and Relu are also optimization techniques and activation functions for deep learning models.

For both classic and ensemble models, additional hyperparameters were employed to optimize performance and prevent overfitting, as shown in Table 4. One of these hyperparameters is `max_depth`, which regulates the maximum depth of each tree in the ensemble, thus limiting the complexity of individual trees. The second is `n_estimators`, which allows controlling the size of the ensemble in terms of the number of base learners while training the model. The third parameter is `subsample`, which determines the proportion of randomly sampled training data utilized to grow each tree within the ensemble. Another technique is the kernel coefficient of SVR, which specifically controls the influence of individual training samples on the decision boundary. All models developed (as described in Section 4.4.2) in this work were implemented using the Python programming language and executed within the Scikit-learn [76] and TensorFlow [77] frameworks. The experimental hardware configuration included an Intel(R) Xeon(R) Bronze 3106 CPU, a 64-bit operating system, and 64 GB of RAM on an Intel processor.

4.4.2. Training and Validation

Once the final input features have been specified, as explained in Section 4.3, it is crucial to develop robust and accurate models for predicting the future energy demand (demand response baselines) of dwellings over multiple time horizons, not exceeding 24 h. To achieve this goal, the datasets were divided into training and validation sets, comprising 80% and 20%, respectively. Considering the training dataset, each model is trained to learn patterns and trends of energy demand, as well as relationships between data series for all dwelling units and different dwelling unit aggregation levels—not a specific pattern of a particular dwelling unit. The goal is to know the performance of each model in the context of all dwelling units and different aggregation levels of dwelling units when traditional machine learning and deep learning models are considered. Each model is trained to predict the demand response baselines of dwelling units, starting at 0:00 am and ending at 11:00 pm (23:00 pm) of the day, as this lasts for the entire period under consideration (five months). Secondly, the trained model, together with its learned information, will later be used to make predictions about the future energy demand of these dwelling units.

With optimal hyperparameters, each model of traditional classic, ensemble, and deep learning models is trained, and its performance is evaluated using three different performance indicators described in Section 4.5. More specifically, using the training dataset, each model is trained with all input features (nine factors) previously defined in Section 3.3. Then, each model is saved at the optimal time for each trained dataset, and its performance is evaluated and compared with its other counterparts. This performance evaluation step is always based on the optimal hyperparameters for each trained model. The purpose of this step is to obtain accurate forecasting models. Subsequently, these forecasting models (i.e.,

pre-trained models) are used to periodically predict the future demand response baseline of dwelling units over various time horizons: 6 h (00:00 am to 06:00 am), 12 h (00:00 am to 12:00 pm), 18 h (00:00 am to 06:00 pm), and 24 h (00:00 am to 11:00 pm), as shown in Figure 8. In the forecasting step, each model only includes the input features (9) defined previously in Section 4.3., while the demand response baseline profile of the dwelling units was excluded. The aim is to determine the ability of each model to predict the future energy demand (demand response basis) of dwellings by considering only those factors related to previous energy demand patterns, dwelling working schedules, and outdoor weather conditions. Ultimately, the performance of those forecasting models is evaluated and validated using the validation dataset. In addition, the discrepancy between the values of the actual and forecasted demand response baselines was used to express the overall performance of the deep learning and traditional machine learning-based forecasting models by considering performance indicators, as described in Section 4.5.

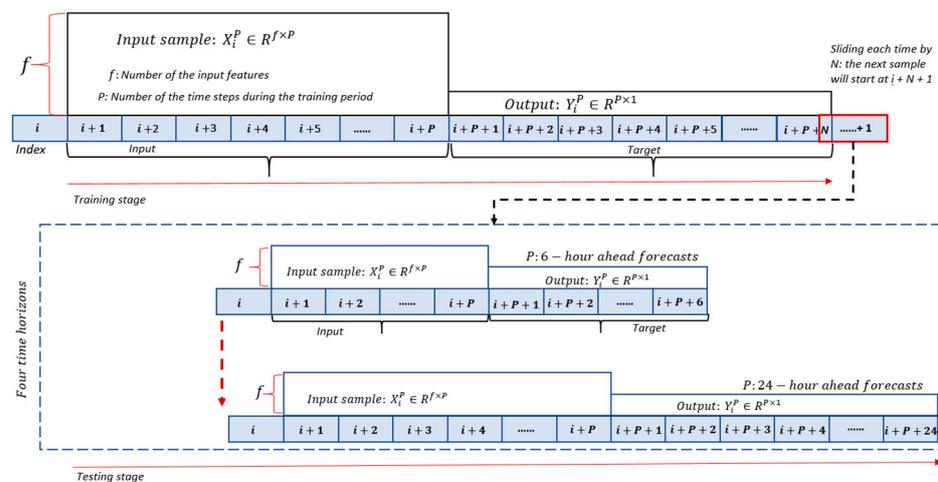


Figure 8. Process of training and testing deep learning and traditional machine learning models using the demand response baseline values of dwelling units.

To further investigate the forecasting behavior of the proposed deep learning and machine learning models, this work is extended to incorporate various aggregation levels of the demand response baseline profile (i.e., 200, 150, 100, 50, 30, 20, and 10 dwelling units). Each time, these models are trained and validated and then evaluated and compared in their performance at each level of aggregation for the demand response baseline profile of dwelling units. The motivation behind this is to better understand the forecasting behaviors of these models in the case of fluctuations or changes occurring in the profile of the demand response baselines of dwelling units. This can be performed by forecasting the energy demand profile of dwelling units [48,78] over a short-term horizon. All forecasting models, whether based on deep learning or classic and ensemble machine learning methods, had the same input features as defined in Section 4.3 and were compared in terms of forecasting accuracy, as discussed in Section 5.

4.5. Performance Assessment

The primary goal of adopting deep learning models, as well as classic and ensemble, is to minimize the gap between the actual aggregated demand response baselines and their forecasted counterparts for the next day. Therefore, the forecasting accuracy of aggregated demand response baselines over various time horizons (6 h, 12 h, 18 h, and 24 h ahead) was measured utilizing three performance metrics. The most commonly used statistical metric to determine forecast accuracy in the literature is the mean absolute percentage error (MAPE) [78,79]. However, MAPE has a limitation concerning the actual values of demand response baselines being zero, which can occur in forecasting demand response baselines. MAPE might take extreme values when the actual values are close to zero in some cases,

making it less reliable. To avoid this limitation, this study also uses mean absolute error (MAE) and root mean squared error (RMSE), which do not have the above-mentioned limitations. The three metrics are formulated as follows.

$$\text{MAPE}(\%) = \frac{1}{N} \sum_{i=0}^N \left| \frac{y_{\text{actual},i} - y_{\text{forecast},i}}{y_{\text{actual},i}} \right| \times 100 \quad (16)$$

$$\text{MAE}(\text{kW}) = \frac{1}{N} \sum_{i=0}^N |y_{\text{actual},i} - y_{\text{forecast},i}| \quad (17)$$

$$\text{RMSE}(\text{kW}) = \sqrt{\frac{\sum_{i=0}^N (y_{\text{actual},i} - y_{\text{forecast},i})^2}{N}} \quad (18)$$

where $y_{\text{actual},i}$ is the actual aggregated demand response baselines per hour i , $y_{\text{forecast},i}$ is the forecasted aggregated demand response baseline per hour i , and N is the number of hours in the datasets. The MAE reflects the magnitude of deviation between forecasted and actual values by utilizing the absolute error, while the RMSE refers to the standard deviation of the residuals between forecasted and actual values of demand response baselines for dwelling units. In contrast, the MAPE measures the forecasting accuracy between forecasted and actual values of demand response baselines for dwelling units, expressed as a relative percentage of forecasting errors. Both RMSE and MAE are scale-dependent metrics and describe the forecasting errors at their original scale. MAPE is a scale-independent metric because the denominator of its equation includes actual values, making it suitable for comparing performance with other studies. The lower values of these metrics mean that the dispersion is more similar between the actual and the forecast demand response baselines. In this study, the MAPE was used as the primary performance measure, with the MAE and RMSE used only as linkage breakers when the MAPE did not show a significant difference between the forecasting models.

5. Results and Discussion

In this section, the performance of the forecasting models developed based on deep learning algorithms is analyzed (Section 5.1.) and compared with those based on classic and ensemble machine learning algorithms (Sections 5.2 and 5.3) over various time horizons, as explained in the previous sections. Section 5.4 shows the forecasting behaviors of all models developed over different aggregation levels of demand response baseline profile for dwelling units. Note that the performance results represent multiple testing outcomes for the developed predictive model, reflecting different test periods. Section 5.5 provides an example to assess the ability of deep learning and classic and ensemble machine learning methods in forecasting energy reductions resulting from the activation of demand response events for 3 h on peak days of the heating season.

5.1. Performance of Deep Learning Models

Table 5 lists the MAPE, RMSE, and MAE values for the forecasting performance of the proposed deep learning methods over various time horizons, compared to their classic and ensemble counterparts, as a function of the input features and each forecasting method. Given the performance results, the authors of this study evaluated the effectiveness of each model over multiple test periods, surpassing five consecutive periods. As shown in Table 5, BiLSTM-based forecasting models could considerably reduce the gap between the measured demand response baselines of dwelling units and their forecasted counterparties, considering different forecast horizons, leading to better performance. As a result, BiLSTM models outperformed their deep learning counterparts, including traditional neural networks (ANNs), with MAPE values of 9.08% (6 h ahead), 11.14% (12 h ahead), 11.11% (18 h ahead), and 11.59% (24 h ahead), respectively. The error represented by RMSE and MAE was also lower than ANN, DNN, CNN, and RNN in all cases of forecasting demand response baselines of dwelling units. The best conditions yielded an RMSE of 7.07 kW and

an MAE of 5.41 kW when forecasting 6 h demand response baselines. This performance is attributed to the bidirectional processing (in both forward and backward directions) of BiLSTM, which allows the neural network to efficiently learn and capture information from both past and future states. Since the primary objective of BiLSTM is to acquire further knowledge concerning a given context by capturing it from more than one perspective and then concatenating the two outputs into a single contextual representation.

Table 5. Average performance of deep learning and classic and ensemble models in forecasting aggregated demand response baselines for dwelling units over various time horizons.

Model	6 h Ahead			12 h Ahead			18 h Ahead			24 h Ahead			
	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	
Classic	MLR	18.63	19.98	34.58	21.64	24.49	25.65	21.76	25.07	24.56	21.89	25.61	22.73
	Ridge	18.52	19.88	34.06	21.57	24.44	25.37	21.58	24.91	24.24	21.74	25.46	22.45
	Lasso	17.39	18.72	29.75	21.06	24.09	23.42	20.84	24.37	22.47	21.02	24.85	20.92
	PolyR	18.62	19.97	34.52	21.64	24.49	25.62	21.73	25.05	24.52	21.88	25.59	22.70
	Bayesian	18.53	19.89	33.65	21.64	24.52	25.19	21.61	24.94	24.06	21.76	25.49	22.31
	KernelR	23.24	24.89	36.88	28.09	32.18	29.64	27.37	31.91	27.48	28.27	33.21	26.59
	SGDReg	23.44	24.77	41.29	25.83	28.72	30.32	24.48	27.84	28.19	24.51	28.35	25.44
	ARIMA	17.31	22.58	27.60	21.32	26.97	23.24	24.15	31.01	24.72	27.42	35.47	25.97
SVR	14.19	16.16	22.54	18.13	21.63	18.77	18.39	22.09	18.71	19.46	23.44	18.18	
Ensemble	XGBoost	8.14	10.28	11.56	13.95	17.61	14.01	14.39	18.07	14.43	15.78	19.84	14.65
	LightGBM	12.21	15.27	18.32	15.86	19.88	16.04	16.31	20.47	16.51	16.88	21.15	15.81
	GB	10.61	13.75	15.68	14.91	19.04	14.69	15.58	19.81	15.47	16.09	20.34	14.86
	RF	15.16	18.01	24.01	18.90	22.97	19.93	18.63	22.71	19.53	19.07	23.47	18.41
	Bagging	14.91	17.33	24.54	18.15	21.68	19.83	17.95	21.66	19.44	18.69	22.83	18.45
	AdaBoost	20.34	22.45	39.70	21.42	24.76	27.74	20.58	24.21	25.31	20.85	24.88	23.27
	CatBoost	20.84	24.76	33.79	24.62	29.88	27.17	24.75	30.29	26.36	25.73	31.61	25.09
Deep Learning	ANN	6.49	7.30	10.18	11.96	15.12	11.81	14.09	17.49	13.98	14.45	18.27	14.49
	DNN	5.89	7.17	8.86	13.07	17.35	12.08	14.97	17.24	12.87	13.93	16.70	13.67
	CNN	6.92	7.35	9.15	13.17	17.11	13.27	14.15	17.33	12.96	14.09	17.66	13.97
	RNN	7.85	8.03	9.92	11.63	15.10	11.18	12.62	16.36	12.74	13.24	16.62	13.58
	LSTM	5.47	8.93	9.23	11.92	15.94	11.73	11.63	15.35	12.27	12.85	16.29	12.03
	GRU	5.81	7.78	8.92	11.51	15.55	11.31	10.48	13.92	11.22	13.04	16.32	12.52
	BiLSTM	5.41	7.07	9.08	11.15	14.86	11.14	10.31	13.13	11.11	12.41	15.12	11.59

Compared to other deep learning models, GRU showed considerable improvements in forecast accuracy over all time horizons considered, better than LSTM, RNN, DNN, and CNN. The performance values of the GRU forecasts were better than those methods by 8.92% and 11.22% in MAPE for the 6 h and 18 h ahead forecasts. The LSTM models performed reasonably well in terms of forecasting accuracy, with MAPE values of up to 9.23% (6 h) and 12.03% (24 h). However, these improvements in the forecasting accuracy were slightly lower than those achieved by BiLSTM. For all four time horizons, BiLSTM performed better than GRU and LSTM, indicating that BiLSTM has a higher learning potential over the next 24 h due to its bidirectional learning capability and produces fewer errors than GRU and LSTM. On the contrary, this potential was found to be lowest with a traditional ANN-based forecasting model, which, due to its shallow structure, was not able to effectively learn time series data and produce accurate predictions for demand response baselines of dwelling units over the four time horizons.

In Figures 9 and 10, four time horizons have been illustrated to provide an intuitive representation of the forecast, each encompassing a single test period. Figure 9 illustrates a comparison of the efficiency of the deep learning methods in representing the closeness between the curves of forecasted and actual values for the demand response baselines of dwelling units over the four time horizons. As shown in Figure 9, these methods were able to produce a similar profile of demand response baselines for dwelling units, with observable differences in the efficiency of each. Figure 10 shows a comparison of the proposed deep learning methods in terms of the magnitude of the error at each hour, in kW, and at different time horizons. As depicted in Figure 10, it illustrates how the best model

reduces the magnitude of an error when producing estimates of the demand response baselines over 6 h, 12 h, 18 h, and 24 h ahead forecasts. The magnitude of the forecast error decreased to some extent as the length of the input interval decreased. For each model, the 6 h input produced the lowest forecast error. However, in terms of conciseness, the BiLSTM, followed by the GRU and LSTM models, outperformed the others in all cases. As accuracy is very important in demand response program applications, BiLSTM, GRU, and LSTM models would be preferred over ANN and DNN models for short-term baseline energy forecasting in residential buildings. Therefore, BiLSTM, GRU, and LSTM, along with the RNN and CNN as alternative methods, should be one of the deep learning methods to consider when developing baseline demand response forecasting models for dwelling units.

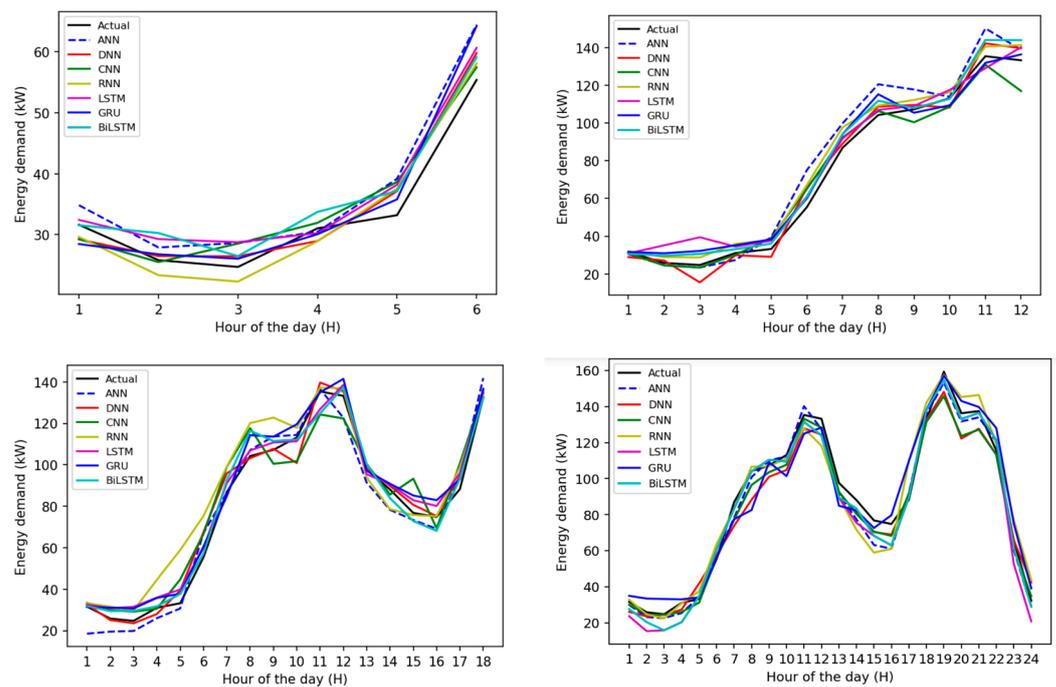


Figure 9. Comparison of forecasted and actual demand response baselines for dwelling units based on the performance of deep learning models over various time horizons, considering a single test period.

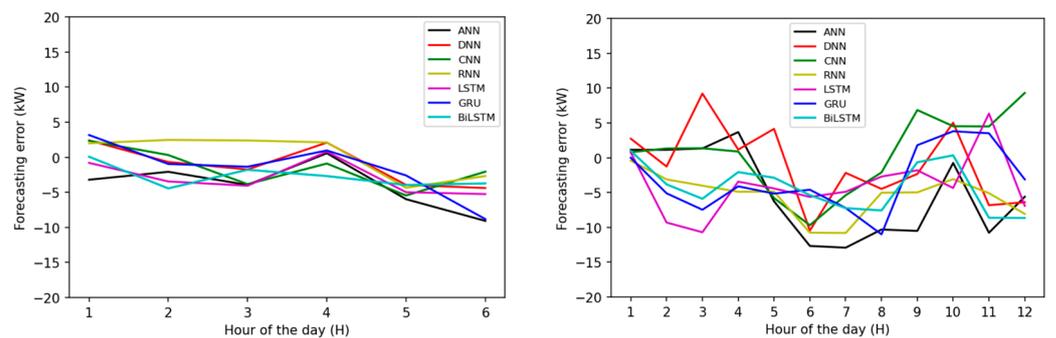


Figure 10. Cont.

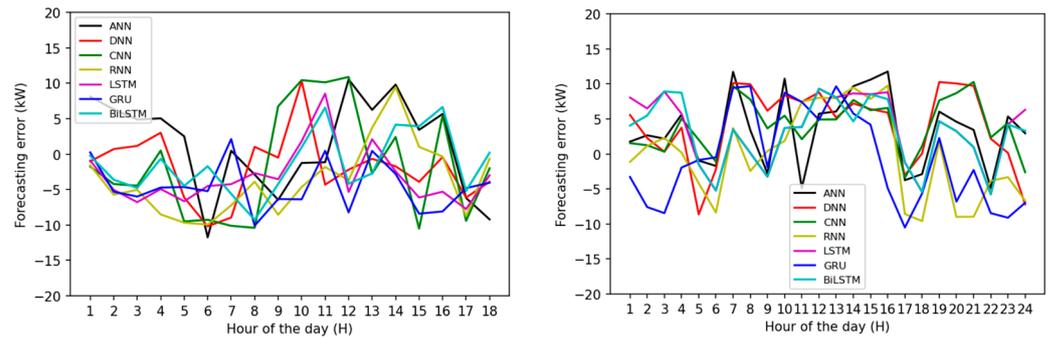


Figure 10. Comparison of forecast errors for deep learning models at each hour and over various time horizons, considering a single test period.

5.2. Comparison of Deep Learning and Ensemble Models

As shown in Table 5, the performance of seven ensemble methods, based on Bagging and boosting techniques, was assessed over various time horizons. Among these seven methods, XGBoost showed the best performance, with the lowest values of MAPE, RMSE, and MAE, when forecasting the demand response baselines of dwelling units over the four time horizons. As a result, the magnitude of error was lower than its counterparts, with values (8.14 kW, 13.95 kW, 14.39 kW, and 15.78 kW) and (10.28 kW, 17.61 kW, 18.07 kW, 19.84 kW) for MAE and RMSE, respectively. In contrast, the performance of AdaBoost and CatBoost was the worst, with (39.70%, 27.74%, 25.31%, and 23.27%) and (33.79%, 27.17%, 26.36%, and 25.09%) of MAPE, (22.45 kW, 24.76 kW, 24.21 kW, and 24.88 kW) and (24.76 kW, 29.88 kW, 30.29 kW, and 31.61 kW) of RMSE, and (20.34 kW, 21.42 kW, 20.58 kW, and 20.85 kW) and (20.84 kW, 24.62 kW, 24.79 kW, and 25.73 kW) of MAE over 6 h, 12 h, 18 h and 24 h ahead forecasts, respectively. Otherwise, other ensemble methods, such as GB, LightGBM, and RF, showed reasonable performance with slightly lower error variability. This is attributed to two main reasons: (1) the strong non-linear mapping generalization and parallelization potential of XGBoost, which is derived from its boosted decision tree-based architecture [80], and (2) the limited ability of other models to effectively learn from a given time series dataset and generalize outcomes has negatively affected the forecast accuracy, despite the implementation of hyperparameter adjustments.

Compared to deep learning methods, all ensemble models could not achieve the same performance observed in deep learning-based forecasting models. This difference is primarily due to the essential structural characteristics of these approaches. As ensemble algorithms without inherent memory, such ensemble methods are unable to capture and preserve past information. As a result, they exhibit suboptimal performance in scenarios where the input time-series information is intricate and where a shorter output interval is required. As shown in Figure 11, ensemble-based forecasting models can produce the general pattern of demand response baselines for dwelling units. However, the magnitude of forecast errors remains considerable, as depicted in Figure 12. For example, the total magnitude error of CatBoost models was 29.88 kW, 30.29 kW, and 31.61 kW of RMSE, whereas that was 15.12 kW, 17.79 kW, and 18.27 kW of RMSE for traditional artificial neural network (ANN) of deep learning models for 12 h, 18 h, and 24 h ahead forecasts, respectively.

To be more concise, in terms of forecasting accuracy, deep learning-based baseline models have exhibited superiority in reducing the magnitude error between forecasted and measured values of demand response baselines, thus demonstrating their advantage in short-term forecasts of demand response baselines for dwelling units. However, the XGBoost method should be one of the alternative forecasting methods to be considered, along with deep learning methods, when estimating demand response baselines in residential neighborhood contexts over various time horizons, not exceeding 24 h.

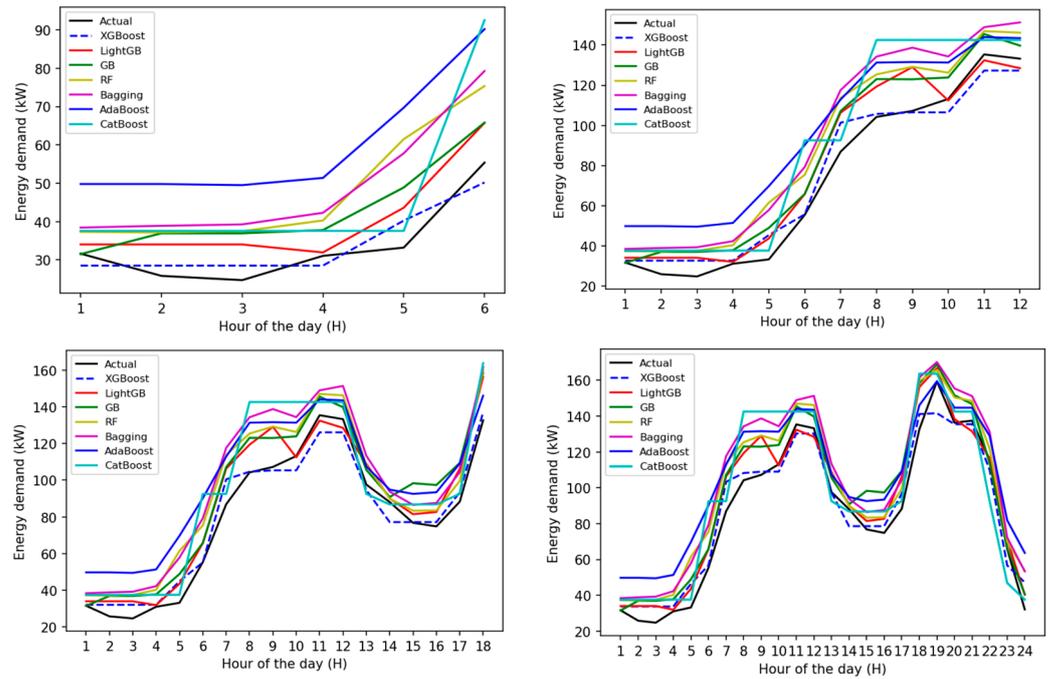


Figure 11. Comparison of forecasted and actual demand response baselines for dwelling units based on the performance of tree-based ensemble models over various time horizons, considering a single test period.

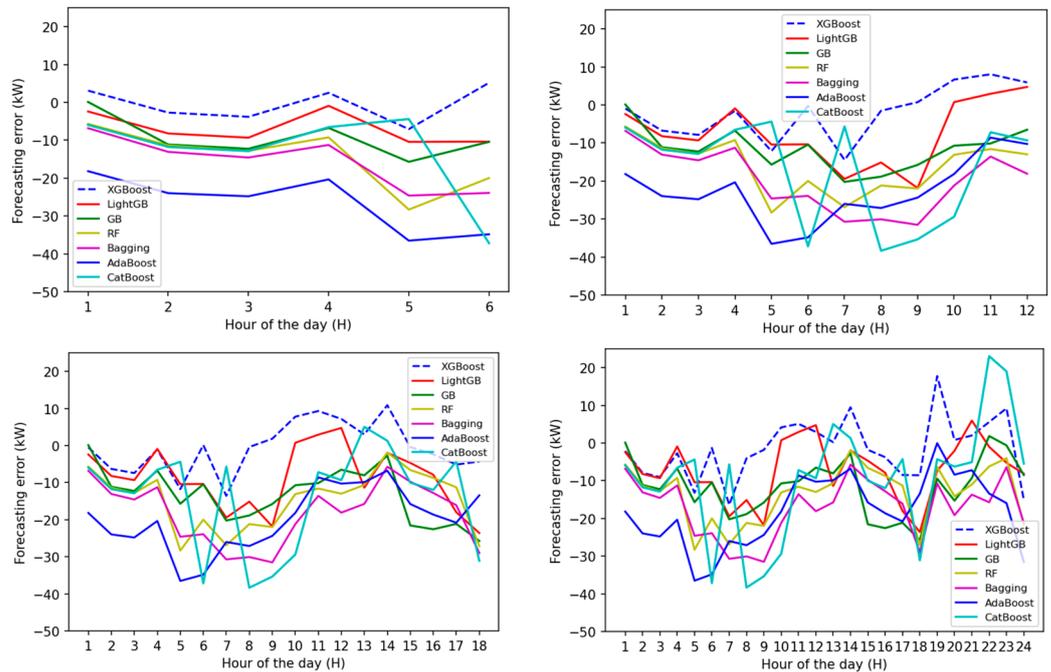


Figure 12. Comparison of forecast errors for ensemble models at each hour and over various time horizons, considering a single test period.

5.3. Comparison of Deep Learning and Classic Models

Compared to deep learning methods, the nine (9) classic methods proposed in this study showed no observable improvements in forecast accuracy over the four time horizons considered. Although methods such as SVR exhibited reasonable performance, they were far from achieving the performance obtained by deep learning-based forecasting models. As shown in Table 4, the best performance conditions for the SVR resulted in an RMSE of

16.16 kW and an MAE of 14.19 kW for the 6 h ahead forecasts. In addition, the RMSE and MAE values for ARIMA and Lasso were (26.97 kW and 24.85 kW) and (21.32 kW and 21.02 kW) at 12 h and 24 h ahead forecasts, respectively. Among the classic models, KernelR and SGDReg performed the worst compared with the other classic models, including SVR, ARIMA, and Lasso, with errors of magnitude of up to 33.21 kW and 28.27 kW (RMSE and MSE) and 28.35 kW and 24.51 kW (RMSE and MSE), respectively. The KernelR and SGDReg models often suffered from instability when using relatively diverse and sparse data samples (because they have different very short-term and short-term forecast horizons), resulting in considerable differences in forecast accuracy.

Figure 13 shows the demand response baseline profiles produced by classical approach-based forecasting models. It can be seen that using the classic models would produce the general pattern of demand response baselines for dwelling units. However, when the classic models are used, more deviations from the measured values of the demand response baselines are likely to be seen. This is due to the presence of more non-linear values in the input features, leading to error magnitudes of up to 36.88% (KernelR) and 41.29% (SGDReg) of MAPE. One of the drawbacks of the classic methods' linear nature is that they prevent high-quality forecasts from being obtained with the original input features and, therefore, fail to achieve performance levels similar to those of deep learning models at the same time horizons. This is due to their inability to capture complex non-linear relationships in the demand response baselines' datasets. In situations where the underlying models are highly non-linear or involve interactions between input features, linear models may not perform as well as more sophisticated non-linear models.

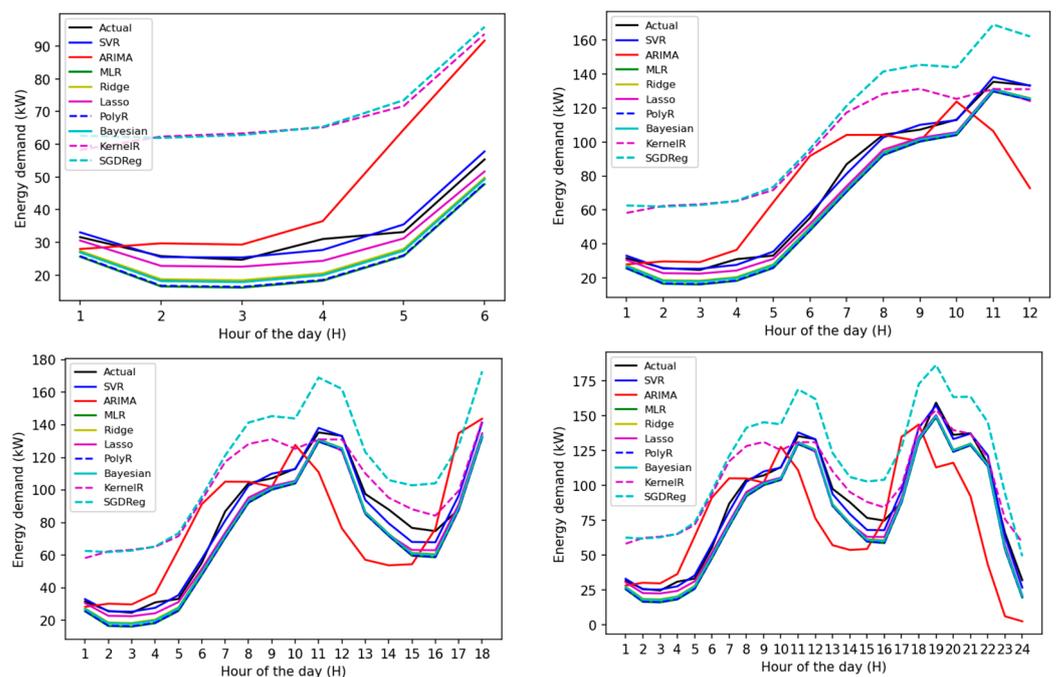


Figure 13. Comparison of forecasted and actual demand response baselines for dwelling units based on the performance of classic models over various time horizons, considering a single test period.

Figure 14 shows a comparison of the forecasted and measured value curves for one test period, where the maximum error amplitudes of the classic models were between (−40.52 and 12.73 kW), (−40.52 and 16.22 kW), (−40.52 and 36.77 kW), and (−40.52 and 34.26 kW) for forecasts at 6 h, 12 h, 18 h, and 24 h, respectively. At the same time, the maximum error magnitude of the deep learning models was between (−9.07 and 3.16 kW), (−12.91 and 9.31 kW), (−11.73 and 10.88 kW), and (−9.59 and 11.76 kW) at 6 h, 12 h, 18 h, and 24 h ahead forecasts, respectively. Two observations in this context can be made, as follows: (1) Deep learning methods demonstrated the best performance, highlighting their advantages in

short-term residential energy forecasting. (2) Classic methods with regularization terms exhibited larger error fluctuations when forecasting demand response baseline values compared to their deep learning counterparts. Therefore, it can be concluded that the classic method failed to significantly reduce the forecasting error when producing accurate estimates of demand response baselines for dwelling units.

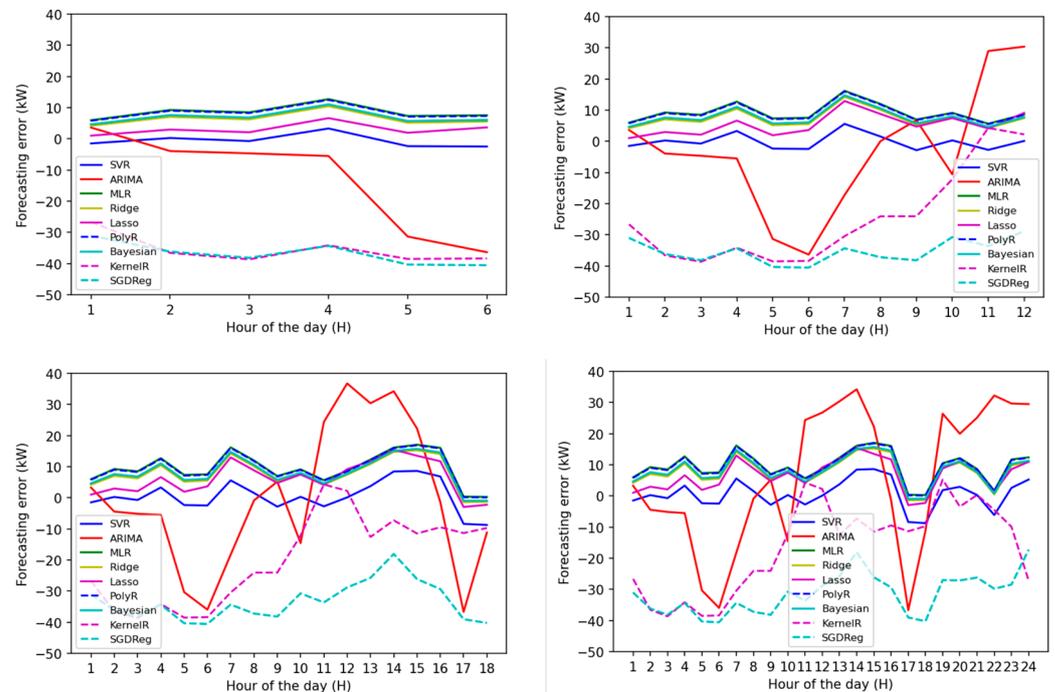


Figure 14. Comparison of forecast errors for classic models at each hour and over various time horizons, considering a single test period.

5.4. Performance at Different Aggregation Levels

To better understand the forecasting behavior of each method, different demand response baseline profiles were used at different levels of dwelling unit aggregation, namely, 200, 150, 100, 50, 20, and 10, as mentioned above. Each method has the same time horizons and input features, allowing analysis of the difference in forecast performance between different models with different levels of forecastability. Figures 15–17 show the overall performance in terms of MAPE values for each of the deep learning, classic, and ensemble models over different time horizons. As depicted in Figure 15, the variations in the demand response baseline profiles for dwelling units indicate that deep learning models typically avoid overestimating future demand response baselines for different dwelling units. It was observed that deep learning-based forecasting models could achieve higher accuracy compared to other models, with reduced errors of up to 8.71%, 10.59%, 11.53%, and 13.05% of MAPE for 6 h, 12 h, 18 h, and 24 h ahead forecasts, respectively. In this respect, BiLSTM-based forecasting models demonstrated superior performance, followed by GRU, LSTM, and RNN, over different time horizons (see Appendices A and B). This behavior in forecasting performance is due to the robust learning capabilities of these methods in mapping energy data of dwelling units, resulting in lower errors than their counterparts from other deep learning methods.

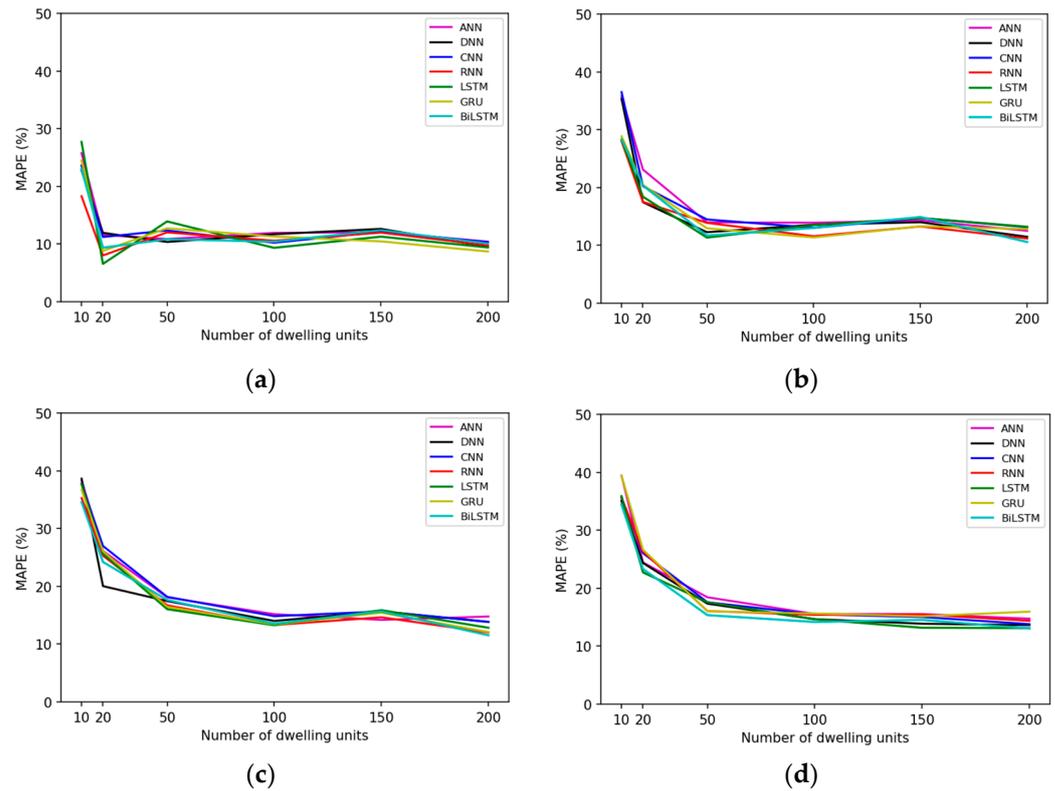


Figure 15. Comparison of MAPE values for deep learning models considering different aggregation levels of the demand response baseline profile of the dwelling units over four time horizons: (a) 6 h, (b) 12 h, (c) 18 h, and (d) 24 h ahead forecasts.

Considering both classic and ensemble methods, the evaluation results demonstrated that the forecasting models based on these methods often tend to overestimate future demand response baselines when using different profiles of demand response baselines, leading to inaccurate forecasts. As shown in Figures 16 and 17, SGDFreg, ARIMA, KernelR, AdaBoost, and RF have considerable error variability, with MAPE values of (39.96%, 31.67%, 30.66%, 29.02%), (25.91%, 23.58%, 28.44%, 29.85%), (25.12%, 20.98%, 20.96%, 20.21%), and (37.91%, 27.63%, 26.11%, 24.63%); these are (49.94%, 75.10%, 111.65%, 96.08%), (70.38%, 63.03%, 70.62%, 67.06%), (94.24%, 72.88%, 74.16%, 65.96%), and (97.03%, 77.48%, 80.41%, 71.25%) when using demand response baseline profiles for 200 and 10 dwelling units over 6 h, 12 h, 18 h, and 24 h ahead forecasts, respectively. These considerable variations in error for each method are due to changes in the dataset of demand response baseline profiles. Consequently, the margin of forecasting behavior between these models and deep learning models is more apparent in such sparse datasets of demand response baselines because, for example, several classic and ensemble models, including Bagging and RF models, suffer more from instability in sparse datasets [81].

Intuitively, the forecasting behavior of the XGBoost, GB, LightGBM, and SVR models was better in terms of forecasting accuracy compared to their counterparts of other classic and ensemble models, as these models enabled better learning from different datasets of demand response baseline profiles (see Tables A1–A6 in Appendix B). However, these models could not outperform the deep learning models in terms of improving forecasting accuracy. This result was expected due to (1) the stability of deep learning models when dealing with sparse datasets and (2) the ability of deep learning models to better learn and efficiently deal with the complexity of sequential datasets (time series data) based on training data. Concerning the values of the accuracy metrics, using the demand response baseline profile for different aggregated dwelling units, for example, 200 dwelling units, results in (10.10%, 10.59%, 11.53%, 13.05%) of MAPE with BiLSTM, in contrast to (19.73%,

18.53%, 20.03%, 19.91%) and (11.05%, 14.12%, 15.45%, 16.02%) with the SVR and XGBoost models over 6 h, 12 h, 18 h, and 24 h ahead forecasts, respectively.

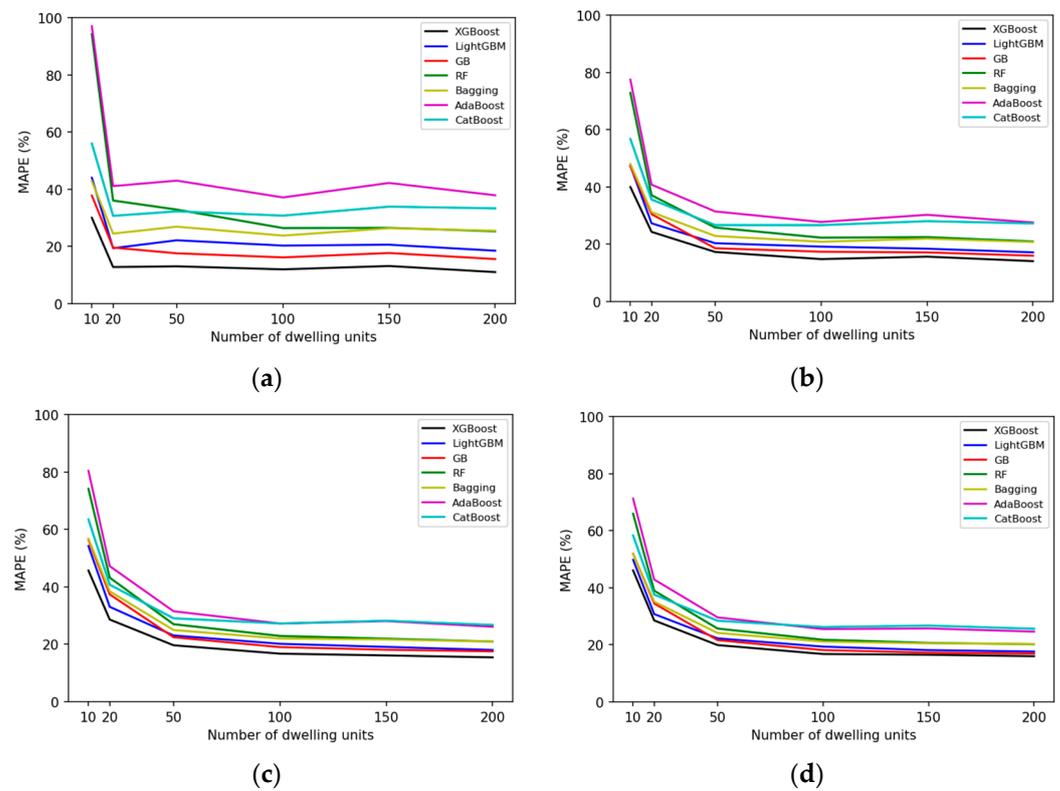


Figure 16. Comparison of MAPE values for ensemble models considering different aggregation levels of the demand response baseline profile of the dwelling units over four time horizons: (a) 6 h, (b) 12 h, (c) 18 h, and (d) 24 h ahead forecasts.

Furthermore, the RMSE and MAE for the deep learning models were lower compared to the classic and ensemble models, demonstrating better forecasting performance than others, as shown in Figures A1–A4 from Appendix A and Tables A1–A6 from Appendix B. Therefore, it can be said that the proposed deep learning methods consistently outperform the comparative classic and ensemble methods for forecasting demand response baselines at different aggregation levels of dwelling units, indicating accurate forecasting behaviors. The significant improvement in MAE and RMSE values primarily demonstrates the ability of the deep learning methods to effectively learn the complexity of time series datasets, which enables the correct capture of the data points of demand response baselines for different aggregations of dwelling units. It can be seen that the proposed deep learning models outperform the other models, as shown in Tables A1–A6. The deep learning models can adequately capture the dynamic stochastic nature of the aggregated demand response baselines, caused by the outdoor weather conditions and the energy demand behavior of the occupants, represented by the working schedules (calendar factor) of the dwelling. As a result, the gap between measured and forecasted aggregated demand response baselines was minimized to some extent, and good forecasting accuracy levels were achieved.

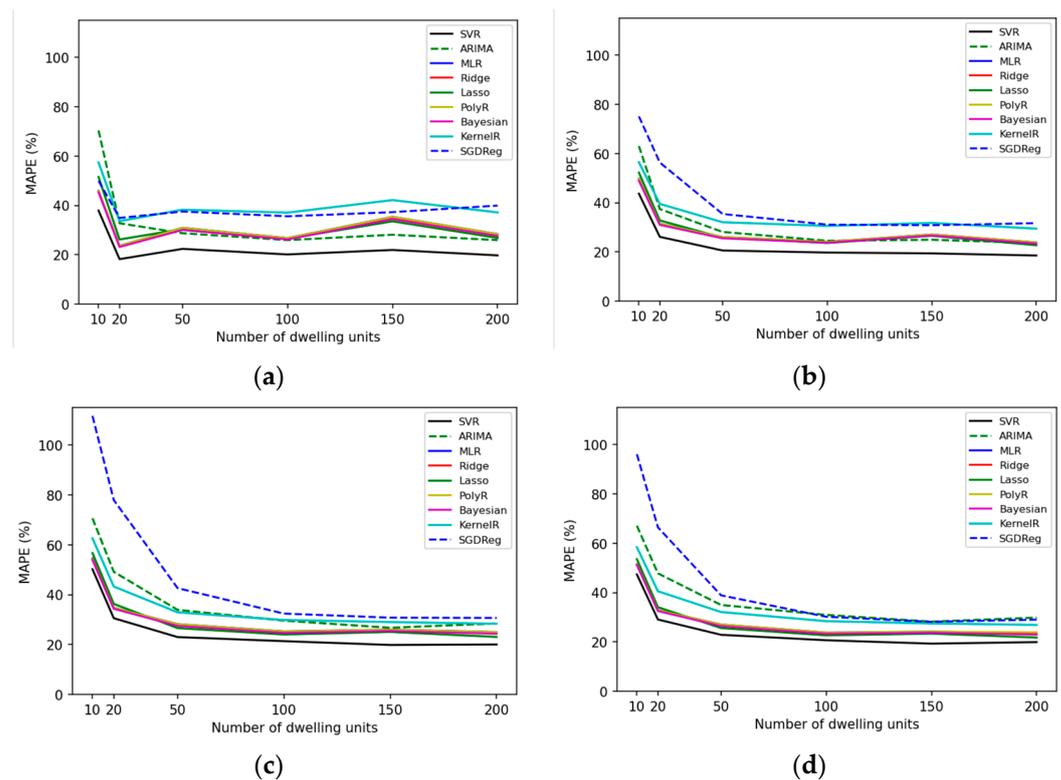


Figure 17. Comparison of MAPE values for classic models considering different aggregation levels of the demand response baseline profile of the dwelling units over four time horizons: (a) 6 h, (b) 12 h, (c) 18 h, and (d) 24 h ahead forecasts.

5.5. Example for Demand Response Forecasts

In order to integrate the forecasting models developed so far for demand response baselines into the residential energy management systems, it is necessary to evaluate their performance not only for the baseline demand patterns but also during the activation of demand response events (i.e., assess their ability in capturing energy reductions when response events are triggered). Energy reductions are calculated based on the difference in energy values between the baselines and the energy profile while the response events are triggered. To do so, datasets of the energy demand profile of all dwelling units when demand response events are activated for 3 h per day (i.e., from 6:00 pm to 9:00 pm) during the coldest days are utilized (see Figure A5 in Appendix A). The proposed deep learning models are also used to provide an accurate estimate of the energy reductions resulting from demand response events during these three hours, and their performance is then evaluated against the classic and ensemble models by considering performance metrics. Table 6 presents the performance results of the deep learning, ensemble, and classic models as a function of the given time horizon, input features, and for each proposed forecasting method. As expected, the deep learning-based forecasting models demonstrated better performance in capturing the energy reductions of the dwelling units due to demand response events over a 3 h time horizon. The LSTM models, followed by ANN, GRU, and BiLSTM, showed a significant ability to improve the forecasting accuracies by minimizing the gap between forecasted and measured values, up to 8.02%, 9.56%, 9.59, and 9.89% of MAPE, respectively. Table 6 also shows that the accuracy of other forecasting models based on deep learning is acceptable. The highest error magnitude of CNN is 7.55 kW of RMSE and 7.15 kW of MAE, making it the least accurate of the deep learning models. However, the forecast outcomes are all close to the actual reductions in energy demand. In contrast, all classic and ensemble models, including SGDReg, KernelR, ARIMA, and LightGBM models, failed to improve forecasting accuracy, with errors up to 46.32%, 34.24%, 36.94%, and 23.15% of MAPE, respectively. Both RMSE and MAE values were also significant,

around (29.33 kW, 25.96 kW, 26.12 kW, 16.35 kW) and (26.68 kW, 23.72 kW, 22.18 kW, 15.65 kW), respectively. Otherwise, XGBoost performed better than its other classic and ensemble counterparts, with an error of 13.12% of MAPE, 9.58 kW of RMSE, and 8.01 kW of MAE, which is the best. However, it was not able to achieve the same level of accuracy when forecasting the energy reduction of dwelling units during the next three hours (3 h ahead forecast).

Table 6. Performance of different forecasting models to evaluate energy reductions due to demand response events for all dwelling units during peak heating days over 3 h ahead.

Model		MAE (kW)	RMSE (kW)	MAPE (%)
Classic	MLR	13.32	15.19	22.82
	Ridge	12.98	14.78	20.68
	Lasso	11.90	13.63	19.99
	PolyR	13.98	15.63	21.01
	Bayesian	14.06	15.72	21.09
	KernelR	23.72	25.96	34.24
	SGDReg	26.68	29.33	46.32
	ARIMA	22.18	26.12	36.94
	SVR	8.44	10.26	15.75
Ensemble	CatBoost	11.73	12.41	18.50
	AdaBoost	10.16	12.52	17.03
	Bagging	10.20	11.75	16.02
	RF	9.09	10.69	14.01
	GB	10.95	11.98	14.93
	LightGBM	15.65	16.35	23.15
	XGBoost	8.01	9.58	13.12
Deep learning	ANN	5.22	6.03	9.56
	DNN	5.42	6.23	10.35
	CNN	7.12	7.55	12.64
	RNN	4.92	5.40	10.16
	LSTM	4.19	4.62	8.02
	GRU	4.86	5.27	9.59
	BiLSTM	4.97	5.87	9.89

For conciseness in this section, a notable observation was the satisfactory performance of the deep learning models compared to the classic and ensemble models in providing accurate estimates of energy reductions over a 3 h ahead. At the same time, the comparison of the actual profile of energy demand reductions with the forecasted profiles of each forecasting method showed that deep learning-based forecasting models can provide more accurate profiles of energy demand reductions resulting from demand response events over short-term and very short-term horizons. This result is due to the intrinsic characteristics of deep learning structures. XGBoost can also be used as an alternative method, along with deep learning methods to support demand response programs in providing accurate estimates of the energy reductions of buildings during the activation of residential neighborhood-level response events.

6. Conclusions

The paper presents the development of a deep learning-based, data-driven learning framework to provide accurate and reliable estimates of demand response baselines in a residential neighborhood context over short-term and very short-term time horizons. Several predictive models based on a deep learning approach, including ANN, DNN, CNN, RNN, LSTM, GRU, and BiLSTM, were developed to predict the future demand response baselines of 337 dwelling units and explore the influence of using different levels of aggregation (200, 150, 100, 50, 20, 10 dwelling units) for the demand response baseline profiles on the forecast accuracy. At the same time, all these methods are compared with

fifteen different classic and ensemble methods to verify their potential to provide accurate and reliable estimates of demand response baselines over a time horizon not exceeding 24 h. The classic methods included MLR, Lasso, Ridge, PolyR, Bayesian, KernelR, SGDR, and ARIMA, while the ensemble methods included XGBoost, LightGBM, GR, RF, Bagging, CatBoost, and AdaBoost.

In all these methods, firstly, the PCC technique is used to select the most significant variables (input features) that influence the energy demand baselines of dwelling units. Secondly, SHAP is used to identify the potential contribution of each input feature to the predictive model. This not only effectively reduces the dimensionality of the input parameters but also enhances the model's running speed while ensuring the incorporation of scientifically and rationally chosen input features. Finally, the controlled-variable method, relying on empirical expertise, is utilized in the experiments to determine the best combination of hyperparameters for building a robust demand response baseline model. Several demand response baseline models were developed and then their performance was analyzed based on MAE, RMSE, and MAPE measured over energy demand baseline datasets from different dwelling aggregation levels to identify the most accurate models over multiple forecast horizons.

The results showed that deep learning-based forecasting models, in comparison with others, could significantly minimize the gap between the actual and forecasted values of demand response baselines at all the different dwelling unit aggregation levels over the time horizons considered. The ANN, DNN, CNN, RNN, LSTM, GRU, and BiLSTM models consistently showed the smallest MAE, RMSE, and MAPE in all comparison experiments, with values up to (6.49 kW, 5.89 kW, 6.92 kW, 7.85 kW, 5.47 kW, 5.81 kW, 5.41 kW), (7.30 kW, 7.17 kW, 7.35 kW, 8.03 kW, 8.93 kW, 7.78 kW, 7.07 kW), and (10.18%, 8.86%, 9.15%, 9.92%, 9.23%, 8.92%, 9.08%), respectively. The BiLSTM models, followed by the GRU and LSTM, had the highest forecasting accuracies, as demonstrated by their superiority in most demand response baseline forecasting experiments. Compared to the performance of classic and ensemble models, XGBoost-based models were among the best for demand response baseline forecasts at different dwelling aggregation levels over the four time horizons considered. Meanwhile, KernelR, SGDR, ARIMA, CatBoost, and AdaBoost were among the worst models when forecasting demand response baselines of dwellings. The classic and ensemble models could not achieve the same level of forecast accuracy in all comparative experiments over the time horizons considered. The optimal combination of hyperparameters, such as hidden layers and hidden units, was sufficient to characterize the different underlying patterns of demand response baselines for dwelling units in datasets. In some cases, the ANN, DNN, and CNN models suffered from instability but were able to self-regulate and achieve high performance in reliably and accurately forecasting residential demand response baselines. This is due to the training techniques associated with the deep learning approach, including the use of ReLU as the activation function and the dropout method for model regularization, which help to improve the forecasting performance of the neural network.

This work, in general, contributes to the body of knowledge on two levels. First, this research not only presents a performance comparison of each proposed method but also highlights the importance of employing advanced neural network models to improve the short-term and very short-term estimation of demand response baselines, as well as the energy reductions resulting from the implementation of demand response programs in residential neighborhood contexts. The MAE, RMSE, and MAPE values clearly show that method structure and other features can significantly influence the production of accurate and reliable demand response baselines. With this comparison, the predictions of residential demand response baselines can be expanded and associated with other issues to promote the efficiency of implementing demand response programs. Second, this work provides important insights into the domain of advanced deep learning-based energy reduction estimation. The results demonstrated that the neural network model with optimal hyperparameters can serve as a useful tool for enhancing demand response

programs by providing accurate and reliable estimates of baseline values in residential neighborhoods. In future work, the author will focus on advanced hybrid neural network techniques and address the associated limitations of real datasets representing the real-world complexity of occupant behaviors. In addition, other features related to energy prices, building typology, such as thermal insulations of dwellings, and occupant behavior, can be incorporated along with concurrent prediction intervals to investigate the effect of uncertainties in the forecasting processes and improve forecast accuracy in the context of residential neighborhoods.

Funding: This research no received external funding.

Data Availability Statement: The datasets and analyses used in this paper will be made available on reasonable request.

Acknowledgments: The author would like to gratefully acknowledge the Engineering Sciences for the Environment Laboratory (LaSIE UMR—7356 CNRS) at La Rochelle University for facilitating this research through the excellent research network, thus enabling fruitful studies like the present one. The author also extends sincere appreciation to Jérôme LE DREAU for his invaluable notes, which improved this study.

Conflicts of Interest: The author declares no conflicts of interest.

Appendix A

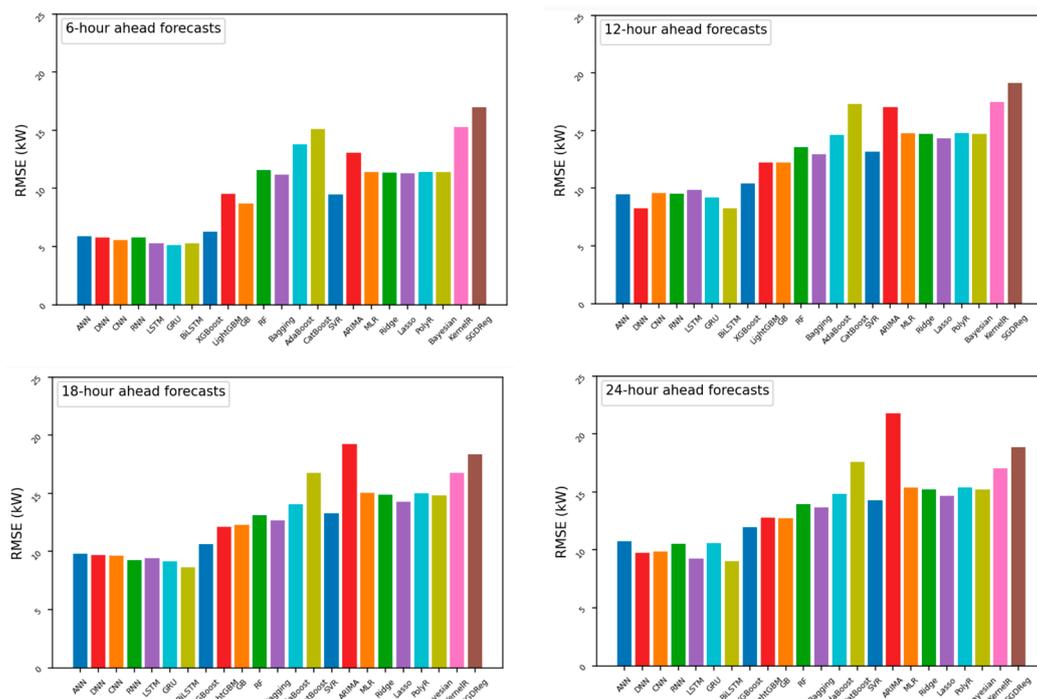


Figure A1. RMSE values for forecasting models at the aggregation level of 200 dwelling units for the demand response baseline profile over four time horizons.

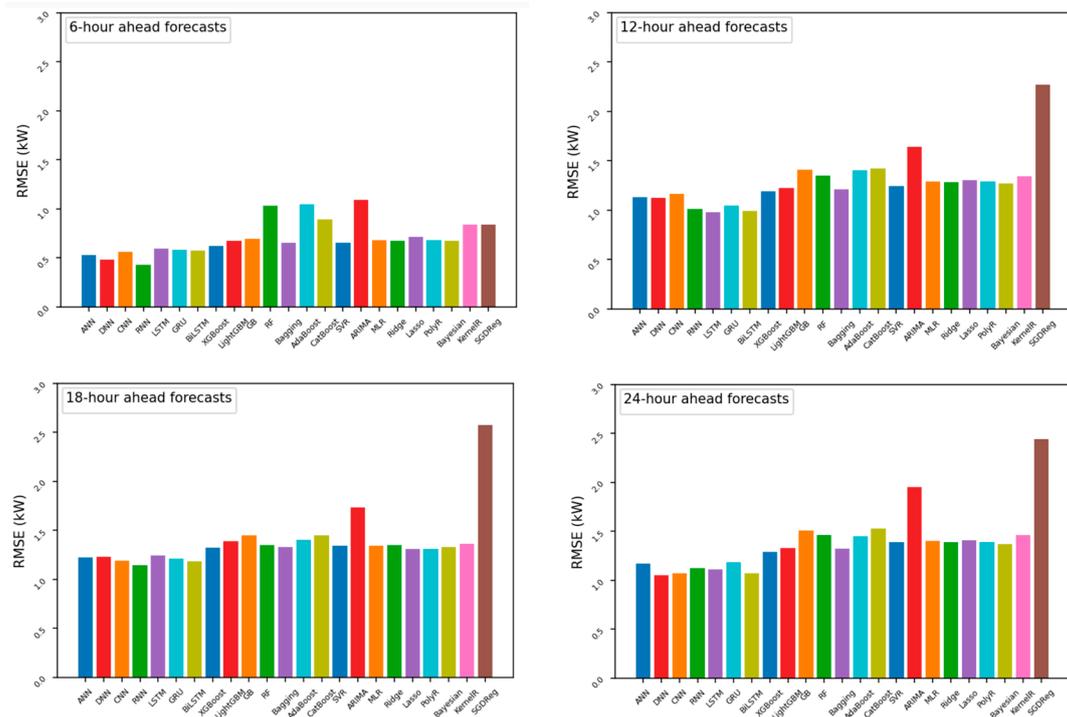


Figure A2. RMSE values for forecasting models at the aggregation level of 10 dwelling units for the demand response baseline profile over four time horizons.

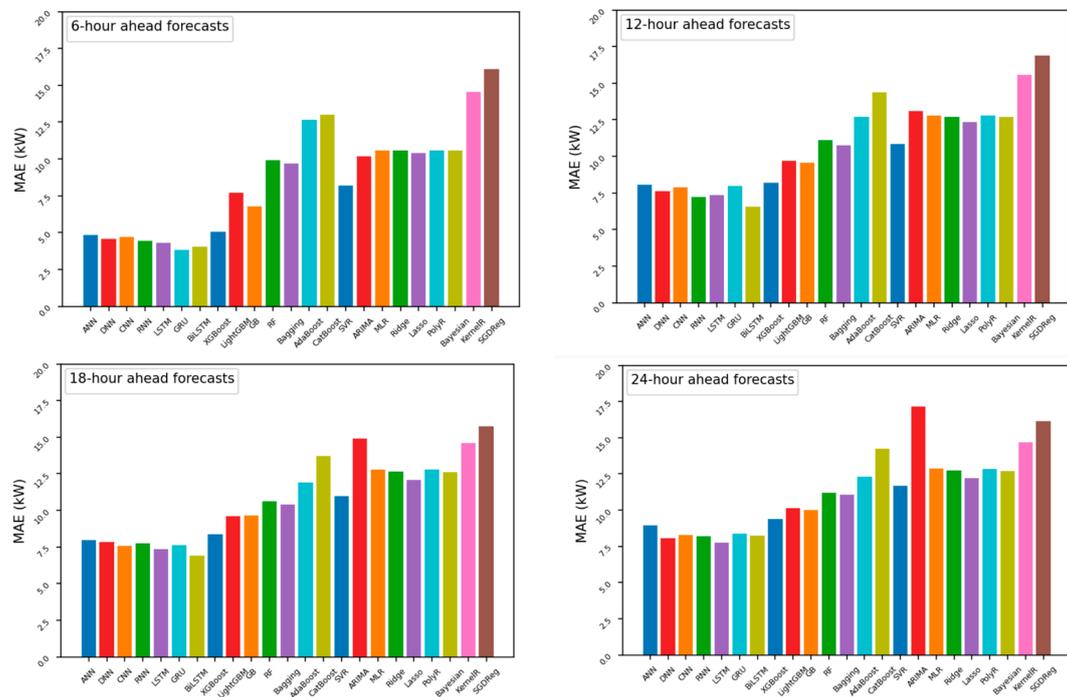


Figure A3. MAE values for forecasting models at the aggregation level of 200 dwelling units for the demand response baseline profile over four time horizons.

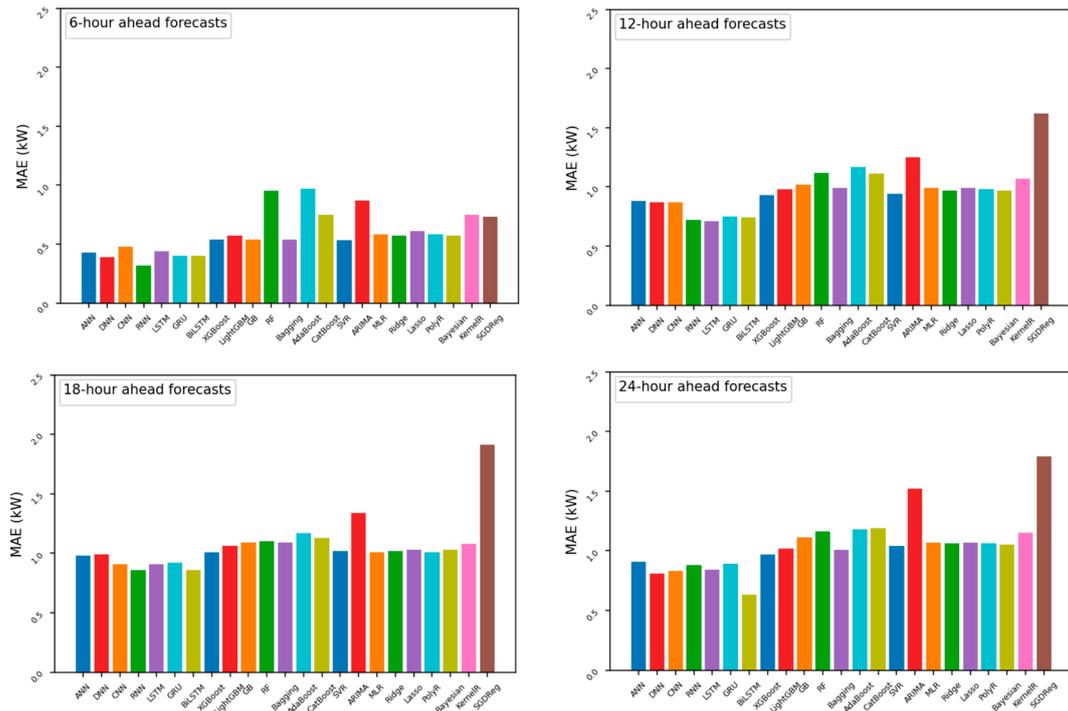


Figure A4. MAE values for forecasting models at the aggregation level of 10 dwelling units for the demand response baseline profile over four time horizons.

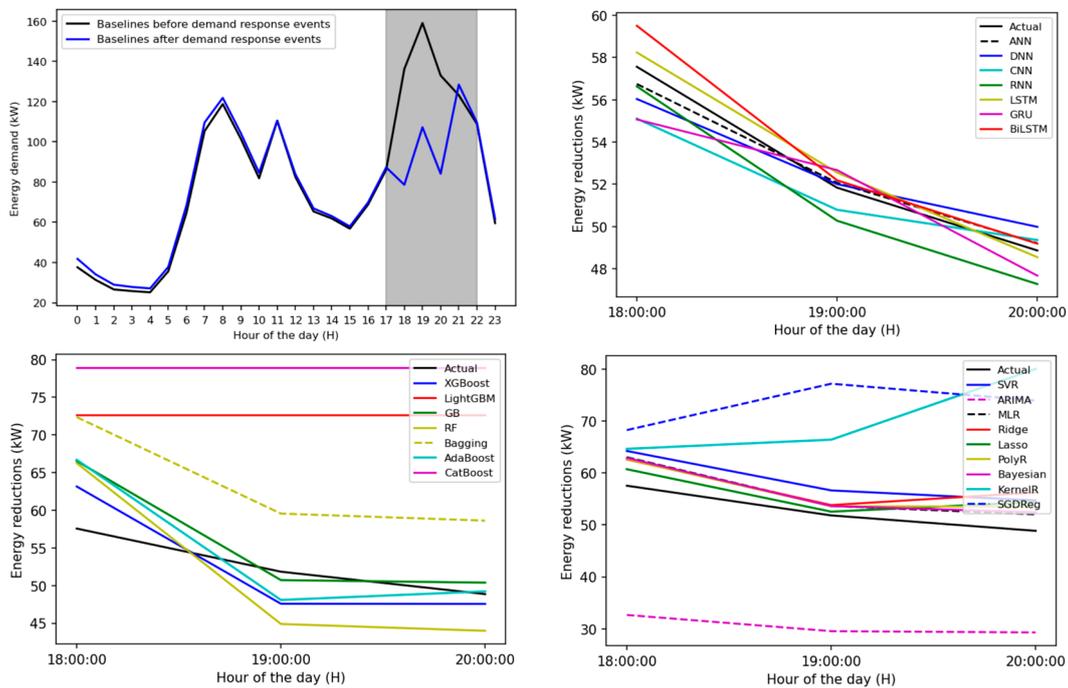


Figure A5. Examples of residential energy demand profiles considering energy reductions caused by the activation of demand response events and predictions for proposed models.

Appendix B

Table A1. Average performance of forecasting models at the aggregation level of 200 dwelling units for the profile of demand response baselines over four time horizons.

Model	6 h Ahead			12 h Ahead			18 h Ahead			24 h Ahead			
	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	
Classic	MLR	10.59	11.39	28.36	12.78	14.77	23.85	12.79	15.02	25.01	12.85	15.36	23.52
	Ridge	10.56	11.36	28.05	12.71	14.69	23.63	12.65	14.87	24.63	12.73	15.23	23.22
	Lasso	10.38	11.29	26.80	12.34	14.33	22.72	12.07	14.27	23.04	12.22	14.67	21.77
	PolyR	10.58	11.38	28.33	12.77	14.76	23.83	12.77	15.01	24.96	12.84	15.39	23.89
	Bayesian	10.59	11.38	27.60	12.71	14.69	23.36	12.60	14.83	24.31	12.69	15.19	22.94
	KernelR	14.56	15.25	37.13	15.57	17.47	29.44	14.58	16.76	28.31	14.67	17.03	26.88
	SGDReg	16.08	16.95	39.90	16.88	19.13	31.67	15.76	18.34	30.66	16.12	18.84	29.02
	ARIMA	10.18	13.06	25.91	13.09	17.02	23.58	14.92	19.26	28.44	17.16	21.79	29.85
SVR	8.18	9.47	19.73	10.85	13.15	18.53	10.96	13.30	20.03	11.67	14.26	19.91	
Ensemble	XGBoost	5.06	6.26	11.05	8.19	10.39	14.12	8.36	10.64	15.45	9.39	11.96	16.02
	LightGBM	7.72	9.54	18.53	9.69	12.24	17.19	9.61	12.13	18.04	10.15	12.79	17.65
	GB	6.79	8.71	15.61	9.56	12.21	16.02	9.66	12.30	17.53	9.99	12.74	16.94
	RF	9.92	11.56	25.32	11.11	13.53	20.98	10.64	13.10	20.96	11.18	13.92	20.21
	Bagging	9.71	11.19	25.52	10.77	12.95	20.79	10.41	12.69	20.93	11.06	13.66	20.29
	AdaBoost	12.66	13.76	37.91	12.71	14.62	27.63	11.88	14.07	26.11	12.29	14.84	24.63
	CatBoost	13.00	15.08	33.34	14.37	17.28	27.30	13.69	16.74	26.72	14.22	17.59	25.63
Deep Learning	ANN	4.84	5.90	10.43	8.07	9.46	12.52	7.97	9.79	14.78	8.94	10.74	14.75
	DNN	4.58	5.79	9.52	7.61	8.27	11.49	7.82	9.66	13.86	8.04	9.77	13.63
	CNN	4.69	5.55	10.34	7.90	9.55	13.16	7.57	9.62	13.83	8.28	9.88	13.81
	RNN	4.46	5.79	9.77	7.22	9.51	11.20	7.75	9.25	12.00	8.17	10.53	14.38
	LSTM	4.31	5.28	9.42	7.35	9.83	13.22	7.34	9.39	12.83	7.75	9.23	13.12
	GRU	3.81	5.12	8.71	7.99	9.21	12.82	7.61	9.13	12.10	8.36	10.55	15.96
	BiLSTM	4.02	5.29	10.10	6.56	8.23	10.59	6.90	8.62	11.53	8.22	9.03	13.05

Table A2. Average performance of forecasting models at the aggregation level of 150 dwelling units for the profile of demand response baselines over four time horizons.

Model	6 h Ahead			12 h Ahead			18 h Ahead			24 h Ahead			
	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	
Classic	MLR	8.65	9.39	35.35	10.45	12.07	27.10	10.52	12.38	25.97	10.62	12.61	24.16
	Ridge	8.64	9.37	34.98	10.43	12.05	26.89	10.46	12.31	25.68	10.58	12.56	23.99
	Lasso	8.42	9.21	33.53	10.35	12.06	26.53	10.26	12.17	25.06	10.38	12.38	23.35
	PolyR	8.65	9.38	35.31	10.45	12.07	27.07	10.51	12.37	25.93	10.61	12.59	24.13
	Bayesian	8.64	9.37	34.39	10.45	12.07	26.62	10.45	12.29	25.40	10.55	12.52	23.68
	KernelR	10.83	11.53	42.13	12.24	13.79	31.77	11.73	13.47	29.01	11.95	13.85	27.47
	SGDReg	10.61	11.46	37.24	12.96	15.13	30.83	12.89	15.21	30.77	12.77	15.18	28.12
	ARIMA	7.86	10.06	28.09	10.89	14.01	24.94	11.83	15.08	26.72	13.83	17.62	28.21
	SVR	6.44	7.54	21.92	8.73	10.65	19.39	8.96	10.98	19.83	9.46	11.58	19.29
Ensemble	XGBoost	4.29	5.41	13.18	7.84	9.93	15.69	7.93	9.89	16.12	7.98	10.08	16.54
	LightGBM	6.09	7.55	20.63	8.12	10.24	18.46	8.33	10.47	19.09	8.59	10.79	18.16
	GB	5.41	7.03	17.73	7.96	10.23	17.18	8.20	10.52	18.10	8.47	10.88	17.26
	RF	7.44	8.83	26.51	9.41	11.45	22.50	9.21	11.28	21.95	9.43	11.58	20.68
	Bagging	7.22	8.40	26.38	8.98	10.82	21.98	8.88	10.79	21.68	9.23	11.27	20.53
	AdaBoost	9.73	10.74	42.21	10.48	12.15	30.28	10.17	11.95	28.07	10.27	12.20	25.76
	CatBoost	9.44	11.42	33.93	11.45	14.14	28.07	11.84	14.58	28.20	12.28	15.19	26.75
Deep Learning	ANN	4.09	5.12	12.06	7.11	9.04	14.33	6.47	8.17	14.20	6.98	9.88	15.54
	DNN	3.73	4.50	12.64	6.79	9.01	14.01	7.60	9.64	15.68	7.45	9.44	13.88
	CNN	3.64	4.43	12.13	7.09	9.34	14.74	7.25	9.36	15.66	7.03	8.92	15.03
	RNN	3.81	5.02	11.99	5.58	7.47	13.27	6.79	8.87	14.65	7.52	9.46	15.46
	LSTM	3.36	4.11	11.29	6.21	8.23	14.73	7.33	9.52	15.88	6.50	8.35	13.18
	GRU	3.28	4.07	10.47	5.92	7.94	13.30	7.37	9.36	15.44	6.96	9.04	15.16
	BiLSTM	3.67	4.38	12.41	6.28	8.44	14.92	7.17	9.11	15.69	6.38	8.08	14.54

Table A3. Average performance of forecasting models at the aggregation level of 100 dwelling units for the profile of demand response baselines over four time horizons.

Model	6 h Ahead			12 h Ahead			18 h Ahead			24 h Ahead			
	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	
Classic	MLR	5.27	5.79	26.74	6.71	7.97	24.09	6.77	8.09	25.28	6.78	8.16	23.79
	Ridge	5.26	5.80	26.50	6.68	7.95	23.92	6.71	8.04	24.98	6.74	8.12	23.63
	Lasso	5.13	5.66	26.49	6.46	7.74	23.60	6.36	7.71	24.02	6.39	7.78	22.66
	PolyR	5.27	5.80	26.72	6.71	7.96	24.09	6.76	8.08	25.24	6.77	8.15	23.76
	Bayesian	5.29	5.83	26.17	6.70	7.97	23.73	6.68	8.02	24.67	6.71	8.09	23.27
	KernelR	7.44	7.91	37.05	8.26	9.44	30.56	7.87	9.16	29.83	7.92	9.27	28.39
	SGDReg	7.71	8.26	35.56	8.75	10.21	31.10	8.54	10.15	32.40	8.51	10.19	30.19
	ARIMA	5.35	6.83	25.92	7.12	9.22	24.53	8.03	10.31	29.57	9.17	11.56	30.94
	SVR	4.36	5.12	20.10	5.91	7.30	19.73	6.02	7.44	21.35	6.25	7.73	20.66
Ensemble	XGBoost	3.90	4.67	12.01	4.52	5.92	14.83	4.72	6.09	16.74	5.04	6.46	16.79
	LightGBM	4.49	5.54	20.32	5.53	7.00	19.19	5.44	6.95	20.17	5.61	7.11	19.38
	GB	3.85	5.07	16.19	5.42	7.12	17.41	5.42	7.09	19.02	5.52	7.12	18.17
	RF	5.47	6.36	26.45	6.09	7.53	22.32	5.93	7.40	22.91	6.09	7.63	21.79
	Bagging	4.89	5.74	23.81	5.74	7.15	20.88	5.69	7.11	22.03	5.93	7.39	21.15
	AdaBoost	6.49	7.19	37.15	6.63	7.85	27.79	6.37	7.64	27.23	6.51	7.91	25.51
	CatBoost	6.27	7.35	30.79	7.27	9.05	26.69	7.15	8.92	27.24	7.41	9.25	26.23
Deep Learning	ANN	2.81	3.49	11.95	3.59	4.82	13.93	3.98	5.55	15.19	4.91	5.91	15.56
	DNN	2.76	3.77	11.71	3.94	4.44	13.58	3.26	5.73	14.01	4.22	5.32	14.66
	CNN	2.53	3.45	10.23	3.84	4.97	13.05	3.79	5.13	14.84	4.25	5.34	15.49
	RNN	2.65	3.56	10.59	3.11	4.35	11.59	3.69	5.23	13.31	4.42	5.69	15.41
	LSTM	2.37	3.28	9.35	3.94	4.70	13.52	3.78	5.21	13.27	4.19	5.27	14.65
	GRU	2.58	3.12	11.32	3.87	4.32	11.38	3.50	5.98	13.40	4.47	5.87	15.65
	BiLSTM	2.13	3.08	10.42	3.18	4.39	13.03	3.40	5.79	13.53	4.74	5.01	14.18

Table A4. Average performance of forecasting models at the aggregation level of 50 dwelling units for the profile of demand response baselines over four time horizons.

Model	6 h Ahead			12 h Ahead			18 h Ahead			24 h Ahead			
	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	
Classic	MLR	3.19	3.52	30.91	3.95	4.73	26.04	3.98	4.91	28.19	4.06	4.94	27.04
	Ridge	3.16	3.50	30.51	3.92	4.71	25.78	3.94	4.77	27.83	4.03	4.91	26.84
	Lasso	3.03	3.39	30.39	3.79	4.62	25.77	3.74	4.62	26.59	3.86	4.77	25.61
	PolyR	3.19	3.52	30.87	3.94	4.72	26.01	3.97	4.81	28.15	4.05	4.93	27.01
	Bayesian	3.15	3.49	30.01	3.91	4.71	25.51	3.91	4.75	27.46	4.01	4.89	26.40
	KernelR	4.07	4.38	38.25	4.84	5.60	32.06	4.66	5.48	32.92	4.79	5.66	32.10
	SGDReg	4.16	4.50	37.49	5.46	6.56	35.41	5.57	6.72	42.62	5.47	6.64	38.87
	ARIMA	3.06	3.86	28.71	4.61	6.04	28.11	4.95	6.37	33.91	5.51	6.93	34.96
	SVR	2.56	2.99	22.36	3.39	4.23	20.56	3.43	4.31	22.99	3.65	4.57	22.87
Ensemble	XGBoost	1.68	2.09	13.07	2.79	3.61	17.33	2.93	3.76	19.66	3.15	4.06	19.93
	LightGBM	2.59	3.13	22.18	3.22	4.09	20.38	3.32	4.19	23.08	3.44	4.38	22.28
	GB	2.10	2.68	17.62	3.10	4.05	18.62	3.33	4.33	22.45	3.42	4.45	21.66
	RF	3.39	3.84	32.90	3.63	4.40	25.85	3.59	4.39	26.99	3.71	4.58	25.71
	Bagging	2.87	3.35	26.93	3.39	4.23	22.91	3.41	4.25	24.97	3.58	4.49	24.18
	AdaBoost	3.87	4.23	43.03	3.96	4.66	31.47	3.83	4.59	31.50	3.91	4.71	29.63
	CatBoost	3.69	4.37	32.28	4.09	5.09	26.70	4.16	5.19	29.05	4.32	5.42	28.45
Deep Learning	ANN	1.48	1.91	10.88	3.12	3.99	13.99	2.81	3.87	18.10	2.99	3.65	18.44
	DNN	1.41	1.92	10.37	2.75	3.67	12.28	2.38	3.47	17.45	2.48	3.54	17.34
	CNN	1.65	1.99	12.36	3.33	4.32	14.49	2.79	3.23	18.18	2.95	3.82	17.58
	RNN	1.44	1.93	12.03	2.83	3.74	13.90	2.10	3.99	16.71	2.69	3.59	16.07
	LSTM	1.83	2.36	13.93	2.58	3.46	11.35	2.03	3.82	16.05	2.80	3.65	17.58
	GRU	1.51	1.93	12.74	2.69	3.50	12.96	2.16	3.91	16.45	2.98	3.87	16.03
	BiLSTM	1.34	1.88	10.85	2.54	3.36	11.69	2.20	3.19	17.63	2.43	3.18	15.36

Table A5. Average performance of forecasting models at the aggregation level of 20 dwelling units for the profile of demand response baselines over four time horizons.

Model	6 h Ahead			12 h Ahead			18 h Ahead			24 h Ahead			
	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	
Classic	MLR	0.98	1.11	23.64	1.47	1.88	31.48	1.49	1.91	34.77	1.56	2.05	32.87
	Ridge	0.97	1.10	23.34	1.46	1.86	31.18	1.47	1.89	34.48	1.55	2.04	32.71
	Lasso	1.04	1.17	26.16	1.50	1.89	32.78	1.51	1.92	36.28	1.58	2.07	34.06
	PolyR	0.97	1.11	23.61	1.47	1.87	31.45	1.48	1.91	34.73	1.56	2.05	32.84
	Bayesian	0.96	1.09	23.12	1.45	1.85	30.97	1.46	1.88	34.32	1.53	2.01	32.49
	KernelR	1.39	1.52	33.60	1.78	2.16	39.52	1.76	2.15	43.26	1.83	2.28	40.52
	SGDReg	1.46	1.60	34.89	2.44	3.26	56.33	2.75	3.58	77.94	2.56	3.42	66.43
	ARIMA	1.38	1.72	32.74	1.81	2.31	37.42	2.06	2.62	49.16	2.26	2.88	47.77
	SVR	0.80	0.96	18.20	1.29	1.71	26.11	1.45	1.79	30.56	1.44	1.95	29.07
Ensemble	XGBoost	0.76	0.92	12.82	1.45	1.86	24.34	1.43	1.92	28.62	1.40	1.82	28.55
	LightGBM	0.84	1.01	19.40	1.30	1.92	27.33	1.54	1.97	33.05	1.42	1.89	30.78
	GB	0.85	1.09	19.62	1.47	1.97	30.51	1.55	2.07	37.38	1.62	2.17	34.50
	RF	1.31	1.45	36.07	1.57	1.93	37.11	1.60	1.97	43.20	1.64	2.09	38.98
	Bagging	0.94	1.11	24.51	1.87	1.97	31.22	1.55	1.85	38.45	1.49	1.95	35.11
	AdaBoost	1.42	1.54	41.11	1.65	1.98	40.82	1.69	2.04	47.21	1.72	2.12	42.89
	CatBoost	1.28	1.48	30.72	1.67	2.08	35.63	1.67	2.11	40.75	1.73	2.23	37.62
Deep Learning	ANN	0.62	0.75	11.59	1.35	1.79	23.12	1.34	1.68	26.13	1.18	1.60	24.45
	DNN	0.67	0.82	11.96	0.96	1.27	17.47	1.08	1.37	20.04	1.17	1.54	24.36
	CNN	0.56	0.64	11.23	1.21	1.73	20.24	1.33	1.69	26.99	1.29	1.62	26.06
	RNN	0.42	0.57	8.03	1.04	1.43	17.53	1.32	1.58	25.31	1.34	1.73	26.38
	LSTM	0.36	0.45	6.57	1.14	1.48	18.43	1.37	1.59	25.65	1.16	1.59	22.74
	GRU	0.46	0.57	8.82	1.26	1.75	20.45	1.37	1.76	26.05	1.32	1.84	26.66
	BiLSTM	0.52	0.64	9.38	1.29	1.80	20.36	1.26	1.49	24.22	1.02	1.09	23.33

Table A6. Average performance of forecasting models at the aggregation level of 10 dwelling units for the profile of demand response baselines over four time horizons.

Model	6 h Ahead			12 h Ahead			18 h Ahead			24 h Ahead			
	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAE (kW)	RMSE (kW)	MAPE (%)	MAEkW	RMSE (kW)	MAPE (%)	
Classic	MLR	0.58	0.68	45.96	0.99	1.29	49.76	1.01	1.34	54.52	1.07	1.40	51.49
	Ridge	0.57	0.67	45.49	0.97	1.28	49.12	1.02	1.35	54.06	1.06	1.39	51.32
	Lasso	0.61	0.71	51.54	0.99	1.30	52.26	1.03	1.31	56.68	1.07	1.41	53.54
	PolyR	0.58	0.68	45.89	0.98	1.29	49.68	1.01	1.31	54.26	1.06	1.39	51.46
	Bayesian	0.57	0.67	45.27	0.97	1.27	48.91	1.03	1.33	53.91	1.05	1.37	51.17
	KernelR	0.75	0.84	57.38	1.07	1.34	56.46	1.08	1.36	62.60	1.15	1.46	58.37
	SGDReg	0.73	0.84	49.94	1.62	2.27	75.10	1.91	2.57	111.65	1.79	2.44	96.08
	ARIMA	0.87	1.09	70.38	1.25	1.64	63.03	1.34	1.73	70.62	1.52	1.95	67.06
	SVR	0.53	0.65	37.91	0.94	1.24	43.68	1.02	1.34	50.21	1.04	1.39	47.35
Ensemble	XGBoost	0.54	0.62	30.04	0.93	1.19	39.99	1.01	1.32	45.68	0.97	1.29	46.09
	LightGBM	0.57	0.67	44.04	0.98	1.22	47.41	1.06	1.39	54.22	1.02	1.33	49.81
	GB	0.54	0.69	37.81	1.02	1.41	47.25	1.09	1.45	56.34	1.11	1.51	51.93
	RF	0.95	1.03	94.24	1.12	1.35	72.88	1.10	1.35	74.16	1.16	1.46	65.96
	Bagging	0.54	0.65	42.84	0.99	1.21	48.06	1.09	1.33	56.75	1.01	1.32	51.85
	AdaBoost	0.97	1.04	97.03	1.17	1.40	77.48	1.17	1.40	80.41	1.18	1.45	71.25
	CatBoost	0.75	0.89	55.99	1.11	1.42	56.76	1.13	1.45	63.47	1.19	1.53	58.31
Deep Learning	ANN	0.43	0.53	25.77	0.88	1.13	35.81	0.98	1.22	38.33	0.91	1.19	39.43
	DNN	0.39	0.48	22.81	0.87	1.12	35.35	0.99	1.23	38.64	0.81	1.05	35.09
	CNN	0.48	0.56	23.63	0.87	1.16	36.51	0.91	1.19	37.75	0.83	1.07	35.55
	RNN	0.32	0.43	18.31	0.72	1.01	28.02	0.86	1.14	35.28	0.88	1.12	35.66
	LSTM	0.44	0.59	27.71	0.71	0.98	28.28	0.91	1.24	37.68	0.84	1.11	35.89
	GRU	0.40	0.58	24.51	0.75	1.04	28.87	0.92	1.21	36.72	0.89	1.18	39.41
	BiLSTM	0.40	0.57	23.03	0.74	0.99	28.33	0.86	1.18	34.58	0.63	1.07	34.47

References

1. Pardalis, G.; Mahapatra, K.; Mainali, B. Comparing public-and private-driven one-stop-shops for energy renovations of residential buildings in Europe. *J. Clean. Prod.* **2022**, *365*, 132683. [[CrossRef](#)]
2. Habib, M.; Timoudas, T.O.; Ding, Y.; Nord, N.; Chen, S.; Wang, Q. A hybrid machine learning approach for the load prediction in the sustainable transition of district heating networks. *Sustain. Cities Soc.* **2023**, *99*, 104892. [[CrossRef](#)]
3. Chreim, B.; Esseghir, M.; Merghem-Boulahia, L. Energy management in residential communities with shared storage based on multi-agent systems: Application to smart grids. *Eng. Appl. Artif. Intell.* **2023**, *126*, 106886. [[CrossRef](#)]
4. Junker, R.G.; Azar, A.G.; Lopes, R.A.; Lindberg, K.B.; Reynders, G.; Relan, R.; Madsen, H. Characterizing the energy flexibility of buildings and districts. *Appl. Energy* **2018**, *225*, 175–182. [[CrossRef](#)]
5. Hu, M.; Xiao, F.; Wang, S. Neighborhood-level coordination and negotiation techniques for managing demand-side flexibility in residential microgrids. *Renew. Sustain. Energy Rev.* **2021**, *135*, 110248. [[CrossRef](#)]
6. Jadhav, P.; More, D.; Salkuti, S.R. Smart residential distribution energy management system with integration of demand response and aggregator. *Clean. Responsible Consum.* **2023**, *9*, 100115. [[CrossRef](#)]
7. Xu, F.Y.; Zhang, T.; Lai, L.L.; Zhou, H. Shifting boundary for price-based residential demand response and applications. *Appl. Energy* **2015**, *146*, 353–370. [[CrossRef](#)]
8. Balakumar, P.; Vinopraba, T.; Chandrasekarn, K. Real time implementation of demand side management scheme for IoT enabled PV integrated smart residential building. *J. Build. Eng.* **2022**, *52*, 1044485. [[CrossRef](#)]
9. Chen, Y.; Xu, P.; Gu, J.; Schmidt, F.; Li, W. Measures to improve energy demand flexibility in buildings for demand response (DR): A review. *Energy Build.* **2018**, *177*, 125–139. [[CrossRef](#)]
10. Ziras, C.; Heinrich, C.; Pertl, M.; Bindner, H.W. Experimental flexibility identification of aggregated residential thermal loads using behind-the-meter data. *Appl. Energy* **2019**, *242*, 1407–1421. [[CrossRef](#)]
11. Valles, M.; Bello, A.; Reneses, J.; Frias, P. Probabilistic characterization of electricity consumer responsiveness to economic incentives. *Appl. Energy* **2018**, *216*, 296–310. [[CrossRef](#)]
12. Sekhar, C.; Dahiya, R. Robust framework based on hybrid learning approach for short-term load forecasting of building electricity demand. *Energy* **2023**, *265*, 126660. [[CrossRef](#)]
13. Li, R.; Satchwell, A.J.; Finn, D.; Christensen, T.H.; Kummert, M.; Le Dreau, J.; Lopes, R.A.; Madsen, H.; Salom, J.; Henze, G.; et al. Ten questions concerning energy flexibility in buildings. *Build. Environ.* **2022**, *223*, 109461. [[CrossRef](#)]
14. Chen, Y.; Xu, P.; Chu, Y.; Li, W.; Wu, Y.; Ni, L.; Bao, Y.; Wang, K. Short-term electrical load forecasting using the support vector regression (SVR) model to calculate the demand response baseline for office buildings. *Appl. Energy* **2017**, *195*, 659–670. [[CrossRef](#)]
15. Zhang, L.; Chen, Y.; Yan, Z. Predicting the short-term electricity demand based on the weather variables using a hybrid CatBoost-PPSO model. *J. Build. Eng.* **2023**, *7*, 106432. [[CrossRef](#)]
16. Lin, J.; Fernandez, J.A.; Rayhana, R.; Zaji, A.; Zhang, R.; Herrera, O.E.; Liu, Z.; Merida, W. Predictive analytics for building power demand: Day-ahead forecasting and anomaly prediction. *Energy Build.* **2022**, *255*, 111670. [[CrossRef](#)]
17. Kazmi, H.; Fu, C.; Miller, C. Ten questions concerning data-driven modelling and forecasting operational energy demand at building and urban scale. *Build. Environ.* **2023**, *239*, 110407. [[CrossRef](#)]
18. Zhou, M.; Yu, J.; Sun, F.; Wang, M. Forecasting of short term electric power consumption for different types buildings using improved transfer learning: A case study of primary school in China. *J. Build. Eng.* **2023**, *78*, 107618. [[CrossRef](#)]
19. Ghenai, C.; Al-Mufti, O.A.A.; Al-Isawi, O.A.M.; Amirah, L.H.L.; Merabet, A. Short-term building electrical load forecasting using adaptive neuro-fuzzy inference system (ANFIS). *J. Build. Eng.* **2022**, *52*, 104323. [[CrossRef](#)]
20. Correa-Florez, C.A.; Michiorri, A.; Kariniotakis, G. Robust optimization for day-ahead market participation for smart-home aggregators. *Appl. Energy* **2018**, *229*, 433–445. [[CrossRef](#)]
21. Ebrahimi, J.; Abedini, M. A two-stage framework for demand-side management and energy savings of various buildings in multi smart grid using robust optimization algorithms. *J. Build. Eng.* **2022**, *53*, 104486. [[CrossRef](#)]
22. Ghasemi, A.; Hojjat, M.; Saebi, J.; Neisaz, H.R.; Hosseinzade, M.R. An investigation of the customer baseline load (CBL) calculation for industrial demand response participants—A regional case study from Iran. *Sustain. Oper. Comput.* **2023**, *4*, 88–95. [[CrossRef](#)]
23. Wijaya, T.K.; Vasirani, M.; Aberer, K. When bias matters: An economic assessment of demand response baselines for residential customers. *IEEE Trans. Smart Grid* **2014**, *5*, 1755–1763. [[CrossRef](#)]
24. Zhang, Y.; Chen, W.; Xu, R.; Black, J. A cluster-based on method for calculating baselines for residential loads. *IEEE Trans. Smart Grid* **2016**, *7*, 2368–2377. [[CrossRef](#)]
25. Wang, F.; Li, K.; Liu, C.; Mi, Z.; Shafie-khah, M.; Catalao, J.P.S. Synchronous pattern matching principle based residential demand response baseline estimation: Mechanism analysis and approach description. *IEEE Trans. Smart Grid* **2018**, *9*, 6972–6985. [[CrossRef](#)]
26. Coughlin, K.; Piette, M.A.; Goldman, C.; Kiliccote, S. Statistical analysis of baseline load models for non-residential buildings. *Energy Build.* **2009**, *41*, 374–381. [[CrossRef](#)]
27. Granderson, J.; Price, P.N. Development and application of a statistical methodology to evaluate the predictive accuracy of building energy baseline models. *Energy* **2014**, *66*, 981–990. [[CrossRef](#)]
28. Walter, T.; Price, P.N.; Sohn, M.D. Uncertainty estimation improves energy measurement and verification procedures. *Appl. Energy* **2014**, *130*, 230–236. [[CrossRef](#)]

29. Hatton, L.; Charpentier, P.; Matzner-Lober, E. Statistical estimation of the residential baseline. *IEEE Trans. Power Syst.* **2016**, *31*, 1752–1759. [[CrossRef](#)]
30. Sharifi, R.; Fathi, S.H.; Vahidinasab, V. Customer baseline load models for residential sector in a smart-grid environment. *Energy Rep.* **2016**, *2*, 74–81. [[CrossRef](#)]
31. Gassar, A.A.A.; Yun, G.Y.; Kim, S. Data-driven approach to prediction of residential energy consumption at urban scales in London. *Energy* **2019**, *187*, 115973. [[CrossRef](#)]
32. Li, K.; Wang, F.; Mi, Z.; Fotuhi-Firuzabad, M.; Duic, N.; Wang, T. Capacity and output power estimation approach of individual behind-the-meter distributed photovoltaic system for demand response baseline estimation. *Appl. Energy* **2019**, *253*, 113595. [[CrossRef](#)]
33. Strivastav, A.; Tewari, A.; Dong, B. Baseline building energy modeling and localized uncertainty quantification using Gaussian mixture models. *Energy Build.* **2013**, *65*, 438–447. [[CrossRef](#)]
34. Zhang, Y.; Ai, Q.; Li, Z. Improving aggregated baseline load estimation by Gaussian mixture model. *Energy Rep.* **2020**, *6*, 1221–1225. [[CrossRef](#)]
35. Bampoulas, A.; Pallonetto, F.; Mangina, E.; Finn, D.P. An ensemble learning-based framework for assessing the energy flexibility of residential buildings with multicomponent energy systems. *Appl. Energy* **2022**, *315*, 118947. [[CrossRef](#)]
36. Sha, H.; Xu, P.; Lin, M.; Peng, C.; Dou, Q. Development of a multi-granularity energy forecasting toolkit for demand response baseline calculation. *Appl. Energy* **2021**, *289*, 116652. [[CrossRef](#)]
37. Tao, P.; Xu, F.; Dong, Z.; Zhang, C.; Peng, X.; Zhao, J.; Li, K.; Wang, F. Graph convolutional network-based aggregated demand response baseline load estimation. *Energy* **2022**, *251*, 123847. [[CrossRef](#)]
38. Park, S.; Ryu, S.; Choi, Y.; Kim, J.; Kim, H. Data-driven baseline estimation of residential buildings for demand response. *Energies* **2015**, *8*, 10239–10259. [[CrossRef](#)]
39. Jazaeri, J.; Alpean, T.; Gordon, R.; Brandao, M.; Hoban, T.; Seeling, C. Baseline methodologies for small scale residential demand response. In *In Proceedings of the IEEE Innovative Smart Grid Technologies—Asia (ISGT-Asia)*, Melbourne, Australia, 28 November–1 December 2016; pp. 747–752. [[CrossRef](#)]
40. Schwarz, P.; Mohajeryami, S.; Cecchi, V. Building a better baseline for residential demand response programs: Mitigating the effects of customer heterogeneity and random variations. *Electronics* **2020**, *9*, 570. [[CrossRef](#)]
41. Rahim, S.; Ahmad, H. Data-driven multi-layered intelligent energy management system for domestic decentralized power distributed systems. *J. Build. Eng.* **2023**, *68*, 106113. [[CrossRef](#)]
42. Ramos, P.V.B.; Villela, S.M.; Silva, W.N.; Dias, B.H. Residential energy consumption forecasting using deep learning models. *Appl. Energy* **2023**, *350*, 121705. [[CrossRef](#)]
43. Santos, M.L.; Garcia, S.D.; Garcia-Santiago, X.; Ogando-Martinez, A.; Camarero, F.E.; Gil, G.B.; Ortega, P.C. Deep learning and transfer learning techniques applied to short-term load forecasting of data-poor buildings in local energy communities. *Energy Build.* **2023**, *292*, 113164. [[CrossRef](#)]
44. Fouladfar, M.H.; Soppelsa, A.; Nagpal, H.; Fedrizzi, R.; Franchini, G. Adaptive thermal load prediction in residential buildings using artificial neural networks. *J. Build. Eng.* **2023**, *77*, 107464. [[CrossRef](#)]
45. Xu, Y.; Li, F.; Asgari, A. Prediction and optimization of heating and cooling loads in a residential building based on multi-layer perceptron neural network and different optimization algorithms. *Energy* **2022**, *240*, 122692. [[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. Afzal, S.; Ziapour, B.M.; Shokri, A.; Shakibi, H.; Sobhani, B. Building energy consumption prediction using multilayer perceptron neural network-assisted models: Comparison of different optimization algorithms. *Energy* **2023**, *22*, 128446. [[CrossRef](#)]
48. Shaqour, A.; Ono, T.; Hagishima, A.; Farzaneh, H. Electrical demand aggregation effects on the performance deep learning-based short-term load forecasting of a residential building. *Energy AI* **2022**, *8*, 100141. [[CrossRef](#)]
49. Olu-Ajayi, R.; Alaka, H.; Suaimon, 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](#)]
50. Wang, L.; Xie, D.; Zhou, L.; Zhang, Z. Application of the hybrid neural network model for energy consumption prediction of office buildings. *J. Build. Eng.* **2023**, *72*, 106503. [[CrossRef](#)]
51. Shaikh, A.K.; Nazir, A.; Khaliq, N.; Shah, A.-S.; Adhikari, N. A new approach to seasonal energy consumption forecasting using temporal convolutional networks. *Results Eng.* **2023**, *19*, 101296. [[CrossRef](#)]
52. Khalil, M.; McGough, A.S.; Pourmirza, Z.; Pazhoohesh, M.; Walker, S. Machine learning, deep learning and statistical analysis for forecasting building energy consumption—A systematic review. *Eng. Appl. Artif. Intell.* **2022**, *115*, 105287. [[CrossRef](#)]
53. Walser, T.; Sauer, A. Typical load profile-supported convolutional neural network for short-term load forecasting in the industrial sector. *Energy AI* **2021**, *5*, 100104. [[CrossRef](#)]
54. Hewamalage, H.; Bergmeir, C.; Bandara, K. Recurrent neural networks for time series forecasting: Current status and future directions. *Int. J. Forecast.* **2021**, *37*, 388–427. [[CrossRef](#)]
55. Amalou, I.; Mouhni, N.; Abdali, A. Multivariate time series prediction by RNN architectures for energy consumption forecasting, TRMEES22-Fr, EURACA Conference 2022. *Energy Rep.* **2022**, *8*, 1084–1091. [[CrossRef](#)]
56. Wang, F.; Xuan, Z.; Zhen, Z.; Li, K.; Wang, T.; Shi, M. A day-ahead PV power forecasting method based on LSTM-RNN model and time correlation under partial daily pattern prediction. *Energy Convers. Manag.* **2020**, *212*, 112766. [[CrossRef](#)]

57. AL-Alimi, D.; AlRassas, A.M.; Al-qaness, M.A.A.; Cai, Z.; Aseeri, A.O.; Elaziz, M.A.; Ewees, A.A. TLIA: Time-series forecasting model using short-term memory integrated with artificial neural networks for volatile energy markets. *Appl. Energy* **2023**, *343*, 121230. [CrossRef]
58. Peng, C.; Tao, Y.; Chen, Z.; Zhang, Y.; Sun, X. Multi-source transfer learning guided ensemble LSTM for building multi-load forecasting. *Expert Syst. Appl.* **2022**, *202*, 117194. [CrossRef]
59. Li, K.; Mu, Y.; Yang, F.; Wang, H.; Yan, Y.; Zhang, C. A novel short-term multi-energy load forecasting method for integrated energy system based on feature separation-fusion technology and improved CNN. *Appl. Energy* **2023**, *351*, 121823. [CrossRef]
60. Song, Y.; Xie, H.; Zhu, Z.; Ji, R. Predicting energy consumption of chiller plant using WOA-BiLSTM hybrid prediction model: A case study for a hospital building. *Energy Build.* **2023**, *300*, 113642. [CrossRef]
61. Xu, L.; Hu, M.; Fan, C. Probabilistic electrical load forecasting for buildings using Bayesian deep neural networks. *J. Build. Eng.* **2022**, *46*, 103853. [CrossRef]
62. Zhang, X.; Zhong, C.; Zhang, J.; Wang, T.; Ng, W.W.Y. Robust recurrent neural networks for time series forecasting. *Neurocomputing* **2023**, *526*, 143–157. [CrossRef]
63. Li, D.; Sun, G.; Miao, S.; Gu, Y.; Zhang, Y.; He, S. A short-term electric load forecast method based on improved sequence-to-sequence GRU with adaptive temporal dependence. *Int. J. Electr. Power Energy Syst.* **2022**, *137*, 107627. [CrossRef]
64. Meng, Y.; Yun, S.; Zhao, Z.; Guo, J.; Li, X.; Ye, D.; Jia, L.; Yang, L. Short-term electricity load forecasting based on a novel data preprocessing system and data reconstruction strategy. *J. Build. Eng.* **2023**, *77*, 107432. [CrossRef]
65. Maltais, L.-G.; Gosselin, L. Forecasting of short-term lighting and plus load electricity consumption in single residential units: Development and assessment of data-driven models for different horizons. *Appl. Energy* **2022**, *307*, 118229. [CrossRef]
66. DIMOSIM (DIstrict Modeller and SIMulator), CSTB (Centre Scientifique et Technique du Bâtiment). Available online: <https://dimosim.cstb.fr/index.html> (accessed on 19 November 2023).
67. Garreau, E.; Abdelouadoud, Y.; Herrera, E.; Keilholz, W.; Kyriakodis, G.-E.; Partenay, V.; Riederer, P. District MOdeller and SIMulator (DIMOSIM)—A dynamic simulation platform based on a bottom-up approach for district and territory energetic assessment. *Energy Build.* **2021**, *251*, 111354. [CrossRef]
68. Martinez, S.; Vellei, M.; Le Dreau, J. Demand-side flexibility in residential district: What are the main sources of uncertainty? *Energy Build.* **2022**, *255*, 111595. [CrossRef]
69. Fathi, S.; Srinivasan, R.; Fenner, A.; Fathi, S. Machine learning applications in urban building energy performance forecasting: A systematic review. *Renew. Sustain. Energy Rev.* **2020**, *133*, 110287. [CrossRef]
70. Sun, Y.; Haghghat, F.; Fung, B.C.M. A review of the-state-of-art in data-driven approaches for building energy prediction. *Energy Build.* **2020**, *221*, 110022. [CrossRef]
71. Gassar, A.A.A.; Cha, S.H. Energy prediction techniques for large-scale buildings towards a sustainable built environment: A review. *Energy Build.* **2020**, *224*, 110238. [CrossRef]
72. Wei, Y.; Zhang, X.; Shi, Y.; Xia, L.; Pan, S.; Wu, J.; Han, M.; Zhao, X. A review of data-driven approaches for prediction and classification of building energy consumption. *Renew. Sustain. Energy Rev.* **2018**, *82*, 1027–1047. [CrossRef]
73. Yu, H.F.; Zhong, F.; Du, Y.; Xie, X.; Wang, Y.; Zhang, X.; Huang, S. Short-term cooling and heating loads forecasting of building district energy system based on data-driven models. *Energy Build.* **2023**, *298*, 113513. [CrossRef]
74. Wang, Z.; Hong, T.; Piette, M.A. Building thermal load prediction through shallow machine learning and deep learning. *Appl. Energy* **2020**, *263*, 114683. [CrossRef]
75. Sehovac, L.; Grolinger, K. Deep learning for load forecasting: Sequence to sequence recurrent neural networks with attention. *IEEE Access* **2020**, *8*, 36411–364226. [CrossRef]
76. Fan, T.J.; Gramfort, A.; Grisel, O.; Halchenko, Y.; Hug, N.; Jalali, A.; Jerphanion, J.; Leraître, G.; Mueller, A.; Metzén, J.-H.; et al. Scikit-Learn (Free and Open-Source Machine Learning Library), Version 1.1.3. 2007. Available online: <https://scikit-learn.org/stable/> (accessed on 30 November 2023).
77. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. Large-Scale Machine Learning on Heterogeneous Systems, Version 2.7.0; TensorFlow Organization (Free and Open-Source Software Library for Machine Learning and Artificial Intelligence): 2015. Available online: <https://www.tensorflow.org/> (accessed on 30 November 2023).
78. Dab, K.; Henaoui, N.; Nagarcheth, S.; Dube, Y.; Sansret, S.; Agbossou, K. Consensus-based time-series clustering approach to short-term load forecasting for residential electricity demand. *Energy Build.* **2023**, *299*, 113550. [CrossRef]
79. Tofallis, C. A better measure of relative prediction accuracy for model selection and model estimation. *J. Oper. Res. Soc.* **2015**, *66*, 1352–1362. [CrossRef]
80. Kumari, P.; Toshniwal, D. Extreme gradient boosting and neural network based ensemble learning approach to forecast hourly solar irradiance. *J. Clean. Prod.* **2021**, *279*, 123285. [CrossRef]
81. Wang, Z.; Wang, Y.; Srinivasan, R.S. A novel ensemble learning approach to support building energy use prediction. *Energy Build.* **2018**, *159*, 109–122. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.