

Article

A Machine Learning Model Ensemble for Mixed Power Load Forecasting across Multiple Time Horizons

Nikolaos Giamarelos ¹, Myron Papadimitrakīs ¹, Marios Stogiannos ¹, Elias N. Zois ¹,
Nikolaos-Antonios I. Livanos ^{1,2} and Alex Alexandridis ^{1,*}

¹ Department of Electrical and Electronic Engineering, University of West Attica, Thivon 250, 122 41 Aigaleo, Greece

² EMTECH SPACE P.C., Korinthou 32 & S. Davaki, Metamorfosi, 144 51 Athens, Greece

* Correspondence: alexx@uniwa.gr; Tel.: +30-210-538-1571

Abstract: The increasing penetration of renewable energy sources tends to redirect the power systems community's interest from the traditional power grid model towards the smart grid framework. During this transition, load forecasting for various time horizons constitutes an essential electric utility task in network planning, operation, and management. This paper presents a novel mixed power-load forecasting scheme for multiple prediction horizons ranging from 15 min to 24 h ahead. The proposed approach makes use of a pool of models trained by several machine-learning methods with different characteristics, namely neural networks, linear regression, support vector regression, random forests, and sparse regression. The final prediction values are calculated using an online decision mechanism based on weighting the individual models according to their past performance. The proposed scheme is evaluated on real electrical load data sensed from a high voltage/medium voltage substation and is shown to be highly effective, as it results in R^2 coefficient values ranging from 0.99 to 0.79 for prediction horizons ranging from 15 min to 24 h ahead, respectively. The method is compared to several state-of-the-art machine-learning approaches, as well as a different ensemble method, producing highly competitive results in terms of prediction accuracy.

Keywords: load forecasting; ensemble learning; neural networks; sparse representation; support vector regression



Citation: Giamarelos, N.; Papadimitrakīs, M.; Stogiannos, M.; Zois, E.N.; Livanos, N.-A.I.; Alexandridis, A. A Machine Learning Model Ensemble for Mixed Power Load Forecasting across Multiple Time Horizons. *Sensors* **2023**, *23*, 5436. <https://doi.org/10.3390/s23125436>

Academic Editors: Yuh-Shyan Chen, Paolo Visconti and Wei Yi

Received: 24 April 2023

Revised: 26 May 2023

Accepted: 2 June 2023

Published: 8 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The modernization of the communication infrastructure of the electrical grid, featuring smart sensors, IoT, and edge computing [1], as well as the deregulation of the electric power markets, has enabled the proliferation of distributed generation, mainly from renewable energy sources (RES) [2]. This new paradigm, while actualizing the much sought-after decarbonization of energy generation, has jeopardized the stability of the distribution network due to the intermittency of the aforementioned resources, mainly in isolated power grids, such as non-interconnected islands. The effect of this intermittency is two-fold: From the distribution system operator perspective, uncertainty in RES generation compromises the ability to effectively plan short-term unit commitment scheduling [3–6], while from the energy market bidder perspective, stochasticity severely constrains bidding strategy and thus, reduces profit margins [7]. An important development in the field of energy transactions is the participation of the energy market in the distribution grid through ancillary services, which is expected to be established in the upcoming years [8].

These shortcomings underline the importance of the application of effective electric load prediction models in the context of multiple operational aspects of the smart grid, such as power stability and security. Especially in the case of micro-grids, storage management is critical and cannot be accomplished without the aid of accurate short-term load forecasts for load shifting and balancing operations [9,10]. Moreover, the grid extension and the

increasing exploitation of smart meters affect the efficient operation of the grid, leading to a complex and multifaceted framework [11,12]. As regards the distribution network on the substation level, load forecasting up to one day ahead, could be a valuable asset in the grid's optimization tasks [13]. Such actions can be carried out, not only by controlling the on-load tap changer (OLTC) and capacitor bank movements, which is currently the industry standard, but also by operation scheduling of batteries in the near future. Load forecasting with multiple time horizons participates in different, interdependent levels of operation of a power grid and thus can make a significant contribution to addressing the aforementioned challenges. A pivotal feature of smart grid is the bidirectional power flow and communication through administrators of generation, transmission, distribution, and end-users. As a result, the corresponding energy data contain mixed power-load (hereby referred to as 'mixed load'). While the majority of the load forecasting models found in the literature predict the electric load production or consumption, the differences between them are frequently studied as well. The forecasting of the so-called net load proves useful for tackling load volatility due to increasing RES penetration [14–18]. These net energy load prediction models utilize historic distribution grid load data as well as measurements of weather features on a substation level in order to infer the net active power (AP) demand of the distribution grid.

To this end, the field of computational intelligence, and more specifically, the branch of machine-learning (ML) [19], has proved to be an invaluable source providing a multitude of approaches to solving the aforementioned problem. ML methodologies are capable of extracting knowledge from historical data in order to develop black-box models and avoid the computationally intensive use of first-principle equations. Such algorithms can exhibit a number of important advantages like efficiency, increased prediction accuracy, robustness, etc., but require a number of suitable data to do so. The most common ML methodologies used in the context of load forecasting are mentioned in the following literature review.

Linear regression (LR) belongs to the methods originally used for load forecasting. An approach based on the regression analysis of decomposed time series is presented in [20] for modeling both voltage and electricity demand volatility. A probabilistic approach for handling the uncertainties of power load data is proposed in [21], employing quantile regression, while in [22], weather conditions and electricity prices are also considered. The influence of climatic variables on electricity demand forecasting is examined thoroughly in references [23,24]. The issue of improving the prediction of load consumption data of a set of smart meters is addressed in [25], featuring a periodic autoregressive model with exogenous variables (PARX), which include calendar and temperature information. An important research matter in load forecasting has been the presence of seasonal effects. To this end, different techniques have been proposed, presenting triple exponential smoothing [26], decomposition methods [27,28], or multiple equation time series [29]. Similarly, the maximum weekly load consumption is forecasted for a one-year horizon in [30], where the different components of the decomposed load are modeled by ARIMAX and ARIMA models. These models incorporate previous forecasting errors in the regression equation and thus outperform the simpler AR models in general. The authors of [31] make use of an ARIMA model for electric vehicle charging demand forecasting, the outputs of which are used to formulate a stochastic day-ahead scheduling problem.

Over the last few years, sparsity has emerged as a general principle for signal modeling. Sparse coding refers to the modeling of data signals as the sum of a few basic elements. Although it was primarily used in image processing, recently, there has been considerable interest in electrical signals. The use of sparse coding for modeling and forecasting individual household electricity loads was studied in [32] by implementing the alternating direction method of multipliers (ADMM) algorithm for solving the dictionary learning problem. A number of papers based on sparse Bayesian learning (SBL) have been published during the last decade, featuring weighted SBL [33,34] or combined kernels SBL [35]. More recently, a hierarchical sparsity approach has been proposed [36] for hourly load forecasting,

achieving remarkable results and outperforming both well-known sparse techniques and rival linear and non-linear methodologies.

Load forecasting using support vector machines for regression has been gaining popularity in recent years, due to the ability of this method to model the non-linearities present in the electric load prediction problem. The support vector regression (SVR) model in [37] deals with peak load forecasting. A polynomial kernel function is chosen, the parameters of which are optimized through multiple cross-validation. In contrast to conventional SVR models, the idea of pooling information from different multiple individual models with different kernel functions is introduced in [38,39]. The hourly load consumption prediction is attempted in [40], proposing an SVR model which exploits the empirical mode decomposition method to disaggregate a time series into two sets of components, respectively describing the trend and the local oscillations of the energy consumption measurements. A hybrid model [41] has been proposed recently for half-hourly demand forecasting using a modified fire-fly optimization algorithm for tuning the SVR hyperparameters.

Owing to their capability of capturing non-linear correlations, feed-forward neural networks [42] have been extensively applied for the prediction of both load consumption and demand. For the scope of this work, multi-layer perceptron (MLP) and radial basis function (RBF) networks are investigated, and their underlying structure is presented in the next section. The following papers propose the implementation of feed-forward neural networks for hourly load forecasting using historical data on hourly consumption. Particular calendar indices are also used by [43,44], while temperature and humidity data by [45]. These variables are common in most models, as the positive impact of exogenous variables on load forecasting has been confirmed by a number of articles, e.g., [46–49]. In many cases, the Levenberg–Marquardt algorithm is selected for artificial neural network training showing a better performance over other algorithms [50,51]. According to [52], the proposed multi-layered feed-forward neural network, which is optimized by grey wolf optimizer, demonstrates superior forecasting accuracy compared to simple LR as well as MLP combined with popular metaheuristic methods. An advanced artificial neural network following a novel connection between layers, called the dense average connection, has been proposed recently [53], showing satisfactory results. Aiming to improve the training procedure of a three-layer feed-forward neural network, [54] proposes an advanced backpropagation algorithm. The authors of [55] propose a methodology for short-term bus load forecasting, where an MLP neural network is combined with a bus-clustering algorithm, achieving reduced computational time. A multi-agent system architecture is designed in [56], including the identification of 24-h demand patterns, the classification of days according to these patterns, and finally, a set of MLPs for demand forecasting. Regarding the use of RBF networks, it occurs on a lesser scale. To be more specific, electric load demand forecasting models have been developed featuring short [57] and long-term [58] prediction horizons.

Random forests (RF) constitute an ensemble machine-learning method presenting very good predictive accuracy and have been used in a number of applications, including load forecasting. Several regression models are evaluated in [59], with random forest regressor to provide better short-term load demand predictions than k -nearest neighbor regressor and linear regressor in terms of MAPE. A probabilistic load forecasting model based on quantile regression forest is developed in [60] and is enhanced by recursive feature elimination for the purpose of feature selection. Moreover, the authors propose an alternative quantile determination method to alleviate the reliability issue of direct prediction interval construction.

It is apparent from the literature that the problem of electric load forecasting has been addressed by multiple machine-learning methods, but without any of them achieving universal superiority in terms of performance. This observation is confirmed not only by studying the individual research results but also by assessing various benchmark comparisons in the literature [61–63]. The inability of universal prediction effectiveness of the

aforementioned models is to be expected taking into account the undesirable characteristics of the load forecasting problem, which include non-linearities and high levels of noise in the associated data. Furthermore, load time series are not statistically static [64], due to the volatile, rapidly changing nature of the weather conditions that affect their power generation component. Different classes of machine-learning methods can cope better with some of these issues but usually underperform with respect to others, e.g., linear models are more robust to noise but cannot capture the non-linearities present in the load forecasting problem. To make things worse, though all of these problems are inherent to load forecasting, their mixture composition changes depending on the time horizon one tries to predict for, making it impossible to single out a unique machine-learning method that could outperform the others across different prediction horizons, e.g., linear methods are often found to perform better in short-term horizons, where data tend to be noisy, but the non-linearities can usually be adequately approximated by linear models, but mostly fail in longer time horizons, where the role of the non-linearities is dominant. It should be noted that the previous observation about the inability of a single method to beat all the others is not only tied to the context of load forecasting, but reveals a more generic concept in machine-learning and optimization, as expressed by the “no free lunch” theorem [65].

To remedy this predicament, one could resort to using a multi-model approach [66], combining multiple machine-learning methods. Unfortunately, in a real-time deployment scenario, an important practical consideration arises for multi-model schemes: How does one select the most suitable model from a pool of trained models for the next prediction timestep? One solution is to employ a rule-based decision system that uses a priori available knowledge, such as the time of day and measured weather conditions at the substation level. This presents a significant impediment. Not only are the rules of such a system difficult to conceptualize, but they also offer no guarantee of continuously optimal model selection. Doing away with a decision system altogether is also problematic since the individually generated predictions do not offer any actionable insight by themselves. Whether a distribution system operator technician or a RES aggregator, a practitioner requires a single forecast value in order to develop their operations strategy. A practical workaround is to discard such selection rules and instead employ a weighting system that assesses models only by using their past prediction performance [67]. The weighting of the output results of basic forecasting LSTM models in [68] is based on the similarity degree between target and identified standard values of load consumption. Two different approaches for determining the weights of multiple forecasters are followed in [69,70], using a novel incremental ensemble weight updating strategy and the minimum-error method, respectively. Alternatively, an extreme learning machine can be employed for combining the outputs of a pool of forecasts, as in [71]. An intelligent decision-making support scheme, including predictive performance evaluation, model properties analysis, structure and fusion strategy optimization, and optimal model preference selection, is incorporated with an evolutionary ensemble learning method proposed in [72] for short-term load forecasting (STLF) problems. Finally, an automated system is established in [73] based on hidden Markov chains for extracting similar day profiles to obtain the best model from a library of available forecasting models. Differently from the previous works, the output neural network (NN) models result from multiple training cycles based on snapshots [74] or the hidden features of a Random Vector Functional Link network [75].

It has become clear that the necessity of providing mixed load forecasts, and indeed for multiple short-term horizons, is a factor of paramount importance in the upcoming transition to smart electricity grids. Moreover, according to the preceding literature review, it is evident that in order to enhance the predictive capability of a model, it should incorporate more than one machine-learning methodologies, which of course should be able to handle the complex dynamic behavior of the mixed load. Finally, such a methodology is necessary to be applicable in an online implementation, which means that the final predictions should be provided in a reasonable amount of time and respond to the behavior of the load through a dynamic decision mechanism.

Realizing the aforementioned requirements and seeking to fill the corresponding research gaps, in this work, we present a novel forecasting scheme that is able to efficiently address the diverse and adverse characteristics of the load forecasting problem for various prediction horizons. The proposed method seeks to create an ensemble of prediction models based on multiple machine-learning techniques comprising different beneficial characteristics that have only been used individually for load forecasting before. Indeed, the sparse coding method introduced in the proposed model has been published very recently and used for the first time in ensemble schemes. As the participating techniques excel in different aspects of the load forecasting problem, their combined usage introduced in this work provides the ensemble with the ability to outperform each individual method in all the horizons tested. In order to efficiently combine the different machine-learning techniques, the proposed method employs an error-based metric on a rolling window of past predictions. This approach enhances the novelty of the proposed method as it does away with the adversity exhibited by complex, rule-based model selection systems. By combining the beneficial characteristics of the aforementioned techniques, the proposed scheme demonstrates superior performance in terms of prediction accuracy, compared to all the submodels, as well as a recently proposed MLP model ensemble from the literature [76], through a wide range of different prediction horizons, spanning from 15 min to 24 h-ahead. Thus, reliable forecasts can be obtained for: (a) One hour ahead or less, which are valuable for various applications at the transmission and distribution network, (b) one day ahead, contributing to the scheduling of generation sources and (c) intra-day forecasting, so as to achieve better optimization results. As a result, the introduced model ensemble can become a powerful tool for administrators and participants in the energy market, easily exploitable in both operational and managerial tasks of smart grids. It should be noted that, at least to the authors' best knowledge, no machine-learning approach that is able to handle this range of prediction horizons has been proposed in the literature. Furthermore, the proposed approach expands the existing literature by using mixed power-load data, i.e., data that include renewable generation measurements. Although there is an abundance of work in forecasting the net power load, the literature on mixed-load forecasting is very scarce. It should be pointed out that the employment of mixed measurements is aligned with the requirements of modern smart grids, where the penetration of renewable resources is a key feature.

The paper is structured as follows: Section 2.1 provides a short description of the different ML methods exploited for building the proposed pool of models. In Section 2.2, the proposed approach is presented analytically and then follows the application of the multi-model scheme upon a certain case study in Section 3, where information about the data and the training process and finally results are given in Section 4. Subsequently, in Section 5, the obtained results are discussed and explained. Finally, conclusions and guidelines for future work are outlined in Section 6.

2. Materials and Methods

2.1. Machine-Learning Methods Short Description

As mentioned earlier, multiple machine-learning methods are involved in the proposed approach. In this subsection, a short description of each one of these methods is provided. Here, we provide a short description of each one of them.

2.1.1. Linear Regression

LR is considered a standard method for addressing problems such as time series prediction, outlier detection, reliability analysis, and feature selection. The regression analysis method is basically a curve-fitting problem. Given a training dataset, $(y_n, x_n), y_n \in \mathbb{R}, x_n \in \mathbb{R}^l, n = 1, 2, \dots, l$, where $y_n \in \mathbb{R}$ represents the output or dependent variable and $x_n \in \mathbb{R}^l, n = 1, 2, \dots, l$ represents the input vector or regressor [77], the aim is to find a function, f , which fits the data. Subsequently, when an unknown data point x^*

appears, we can use this function in order to calculate/predict the respective output y^* [19]. Equation (1) describes the relation between the input and output variables.

$$y = \theta^T x + \eta \quad (1)$$

The objective of a regression problem is the estimation of regression coefficients vector θ which arises through the solution of a least squares problem.

$$\hat{\theta} = (x^T x)^{-1} x^T y \quad (2)$$

Although more modern and advanced methods have been developed, LR is still used due to its simplicity and robustness, which are of great importance, especially in online implementation of load forecasting. However, the inability of the method to extract the non-linear behavior of the load is an important disadvantage.

2.1.2. Sparse Coding

When addressing a linear system, for example $x = D\alpha$, $x \in \mathbb{R}^{m \times n}$, $\alpha \in \mathbb{R}^{k \times n}$, the number of predictors, p , can be extremely large. Thus, it is impossible to fit a linear model when $p < m$, or even when $p \approx m$ without overfitting (depending on the noise level), but it may still be possible to fit a sparse linear model that only depends on a reduced number of predictors s , where $s < p$. The dictionary $D \in \mathbb{R}^{m \times k}$ is underdetermined, and therefore, the linear system has infinite possible solutions. The sparse regression (SR) problem is defined as the search for the sparsest solution, i.e., the one with the fewest non-zeros, and is described by Equation (3) [78]:

$$\min_{\alpha \in \mathbb{R}^{k \times n}} \|\alpha\|_0 \text{ subject to } x = D\alpha \quad (3)$$

where $\|\alpha\|_0$ is the ℓ_0 norm, which counts the non-zero components of α . Although the problem in question is NP-hard, it can often be solved using approximation methods, such as greedy algorithms: Orthogonal matching pursuit [79], thresholding algorithm [80], or relaxation algorithms such as basis pursuit [81] are commonly used for this purpose. SR is a methodology of low complexity that carries the disadvantage of linear correlation assumption between mixed load features, but on the other hand is able to prevent overfitting compared to more complicated ML approaches. A key point stage in the sparse representation procedure is the so-called dictionary learning, which consists of finding the elements of the dictionary (atoms). Dictionary learning can be formulated as a joint unconstrained optimization problem [82], given in the form of (4).

$$\min_{D \in C, \alpha \in \mathbb{R}^{k \times n}} \sum_{i=1}^n \|x_i - Da_i\|_2^2 + \lambda \|\alpha_i\|_0 \quad (4)$$

where $C = \{D \in \mathbb{R}^{m \times k} \text{ s.t. } \forall j = 1, \dots, k, d_j^T d_j \leq 1\}$, denotes the feasible space of dictionary D . The role of the parameter λ is to regulate the sparsity level of the coefficient vector.

Dictionary learning can be accomplished using several algorithms, like online dictionary learning [83], method of directions [84], K-SVD [85], stochastic gradient descent [86], or LASSO [83].

2.1.3. Support Vector Regression

The next method to be included in the pool of models constitutes an extension of support vector machines in regression and is called SVR. The basic idea of this methodology is the use of a non-linear transformation $\varphi(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^{n_h}$ that maps the real data into a multi-dimensional space [87] and the subsequent application of LR. According to this approach, a linear function s is supposed to exist in the multi-dimensional space, which

models the non-linear relation between the input and output data of the initial space [88]. Such an equation is given in (5).

$$s(x) = w^T \varphi(x) + b \quad (5)$$

where $\varphi(x)$ denotes the kernel function and $w^T \in \mathbb{R}^{n_h}$, $b \in \mathbb{R}$ are the regression coefficients. The problem of calculating the variables w and b is reduced to the minimization of the structural risk functional.

$$R = \min_w \frac{1}{2} \|w\|_2^2 + C \sum_{i=1}^n |s(x_i) - y_i|_\epsilon \quad (6)$$

where y contains the real measurements and C is a penalty term used to balance between data fitting and overfitting.

The employment of the kernel trick allows SVR to acknowledge the presence of non-linearity in mixed load series. However, when a separating hyperplane in a given dimension cannot be found, then it is required to move in a higher dimension. In this case, the computational cost will increase as well. Furthermore, the use of support vectors makes the method sensitive to noisy data and outliers.

2.1.4. Neural Networks

Neural networks (NNs) constitute an important family of black-box modeling techniques. NNs are very accurate, robust, fault-tolerant, and flexible to adapt to any process given a suitable number of quality data. The proposed model ensemble includes two representative NN architectures, namely the MLP and the RBF. MLPs identify the process dynamics and form the model by guiding the input data through weighted successive layers of non-linear functions (threshold, sigmoid, etc.) called activation functions or nodes. The input (activity) $\mu_l(x)$ to each one of the L nodes is the weighted sum of all N input variables to that node.

$$\mu_l(x) = x \cdot w \quad (7)$$

where w are the weights corresponding to each variable of the input vector x . The intermediate layers between the input and the output are called hidden layers. The schematic of a typical MLP NN with 2 hidden layers is presented in Figure 1. The prediction produced by an MLP is the weighted sum of the final layer outputs. Due to the existence of non-linear characteristics in mixed-power load data, MLPs certainly seem like a promising method for the problem at hand.

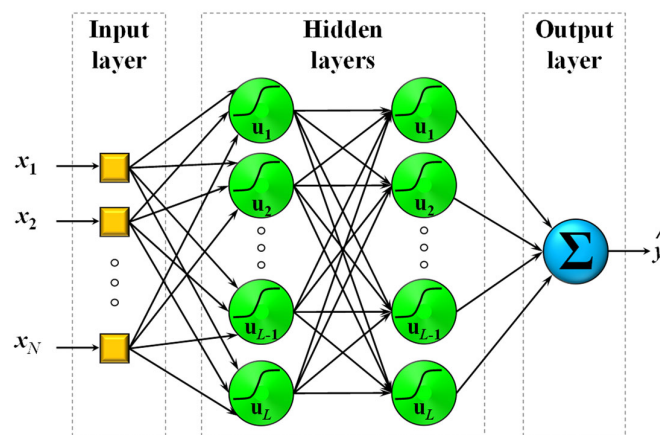


Figure 1. A typical fully connected multi-layer perceptron (MLP) neural network (NN) structure comprising of N inputs, $x_1, \dots, x_n$, 2 hidden layers of L neurons each, and one-dimensional output, $\hat{y}$.

On the downside, MLP training is usually performed by some form of backpropagation technique which usually requires more than one intertwined iterative procedure to

fully optimize the involved parameters, i.e., the number of layers and nodes, the weights, etc. Depending on the size and architecture of the MLP network, and the input space, this procedure may become computationally intensive, and thus it is commonly performed offline. A quite critical drawback of all backpropagation-based techniques is that they get easily trapped in local minima. In this case, the provided solution may not be satisfactory, a fact which leads to a tedious retraining procedure.

RBFs are similar to the MLPs in the sense that data are fed through the input layer and follow a straight path to the output layer but they differ in that there exists only one hidden layer which comprises radially symmetric activation functions (Gaussian, quadratic, thin plate spline, etc.). A typical RBF NN using Gaussian activation functions can be seen in Figure 2. The input layer distributes the data of the N inputs to the L nodes of the hidden layer, which are positioned to a specific point of the input space through a process of training. The activity $\mu_l(\mathbf{u}_k)$ of each node is calculated using the Euclidian distance between the input data $\mathbf{u}_k$ and the center $\mathbf{c}_l$ of each node.

$$\mu_l(\mathbf{u}_k) = \|\mathbf{u}_k - \mathbf{c}_l\|_2 = \sqrt{\sum_{i=1}^N (u_{i,k} - c_{i,l})^2}, l = 1, 2, \dots, L, k = 1, 2, \dots, K \quad (8)$$

where K is the number of training data. The chosen RBF g_k receives the activity value and calculates the node output. The linear combination of all hidden layer node outputs provides the NNs prediction $\hat{y}_k$

$$\hat{y}_k = \mathbf{g}_k \cdot \mathbf{w} \quad (9)$$

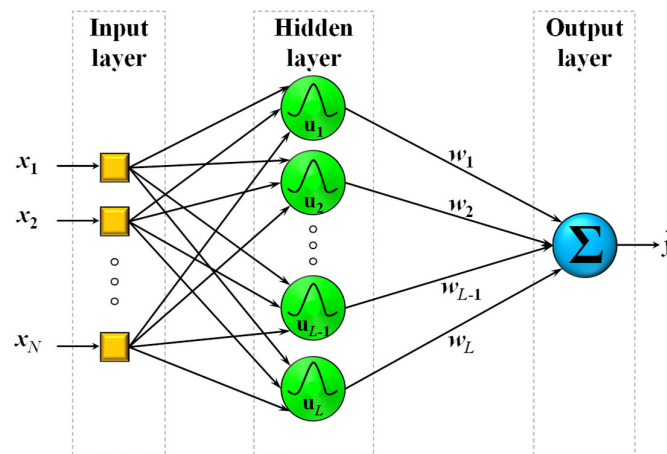


Figure 2. A typical Gaussian-based radial basis function (RBF) neural network (NN) structure comprising of N inputs, $x_1, \dots, x_n$, L neurons in the hidden layer, and one-dimensional output, $\hat{y}$.

The training algorithm for an RBF NN is usually broken down into two phases, the first of which discovers the optimal number and location of the hidden node centers in the input space, while the second one calculates the weights $\mathbf{w}$ usually through simple LR. Due to the fact that the training process is broken into two phases, RBF NNs are able to use very fast algorithms. In fact, some of the current RBF training techniques [89] are deterministic and non-iterative, requiring only a single pass of the data to converge, in contrast to other NN architectures which (a) are epoch-based requiring multiple passes of the data and (b) are stochastic requiring multiple runs to overcome their sensitivity to initial conditions. RBF networks provide very strong interpolation tools, usually outperforming other NN-based techniques provided that dense and good quality training data are available. In the absence of adequate training data though, their performance may become rather poor. Therefore, their application, in combination with the other models of the pool, can make a significant contribution to mixed-power load prediction.

2.1.5. Random Forests

The last method involved in the proposed approach is RF [90]. As the name suggests, an RF is a tree-based ensemble, with each tree depending on a collection of random variables. More formally, for a p -dimensional random vector $X = (X_1, \dots, X_p)^T$ representing the real-valued input or predictor variables and a random variable Y representing the real-valued response, we assume an unknown joint distribution $P_{XY}(X, Y)$. The goal is to find a prediction function $f(X)$ for predicting Y . The prediction function is determined by a loss function $L(Y, f(X))$ and defined to minimize the expected value of the loss

$$E_{XY}(L(Y, f(X))) \quad (10)$$

where the subscripts denote expectation with respect to the joint distribution of X and Y . Intuitively, $L(Y, f(X))$ is a measure of how close $f(X)$ is to Y . It penalizes values of $f(X)$ that are a long way from Y . As for simple LR, squared error loss could be a typical choice of L . Ensembles construct f in terms of a collection of linear estimators of x , the so-called “base learners” $h_1(x), h_2(x), \dots, h_J(x)$, where J denotes the number of trees and is user-specified, according to the following iterative procedure. Let $\mathcal{D} = \{(x_1, x_1), \dots, (x_N, x_N)\}$ denote the training data, with $x_i = (x_{i,1}, \dots, x_{i,p})^T, i = 1, \dots, N$. For each $j \in (1, J)$, a bootstrap sample $\mathcal{D}_j$ of size N is extracted from $\mathcal{D}$ and a corresponding tree $h_j(X, \Theta_j)$ is derived implementing the binary recursive partitioning [91]. The prediction extraction using a standard RF regressor is depicted in Figure 3. For each unsplit node of the tree, the best binary split among all binary splits on the $m \in (1, p)$ predictors, is found. The component Θ_j is used to inject randomness first by bootstrap sampling and second by the random subset of m predictors. Once the base learners are found, a prediction at a new point x is given by

$$f(x) = \frac{1}{J} \sum_{j=1}^J h_j(x) \quad (11)$$

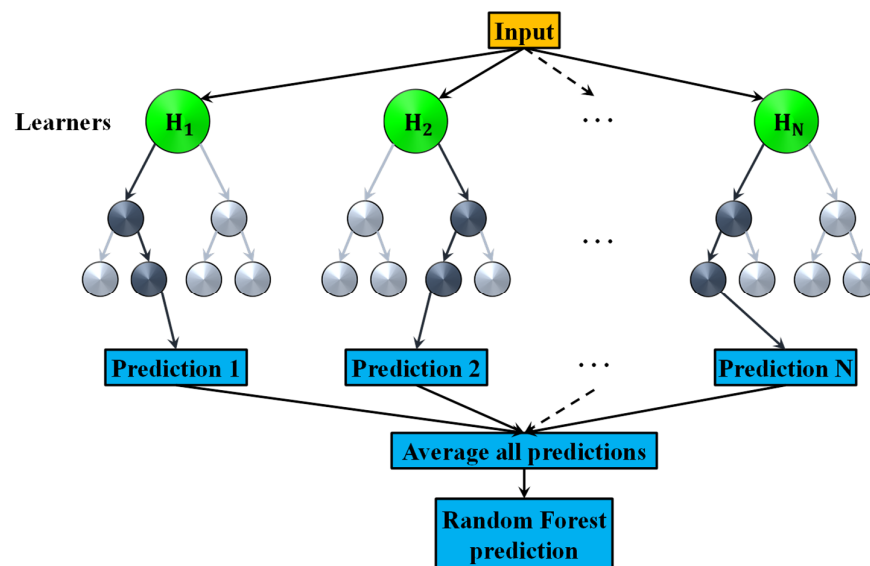


Figure 3. A typical random forest architecture comprising of N tree learners, $H_1, H_2, \dots, H_N$. The prediction set for each learner is averaged to produce the final predictions.

RF is a simple and reliable forecasting tool. Its main limitation is the trade-off between performance and the number of trees. Increasing this parameter can lead to more accurate predictions and prevent overfitting but can also make the algorithm too slow and ineffective for real-time predictions. It is, therefore, understandable that this methodology may prove suitable in specific areas of the dataset.

2.2. Machine-Learning Model Ensemble

Recognizing the individual advantages and disadvantages of the machine-learning methods described in Section 2.1, the proposed scheme seeks to create an ensemble that will successfully combine their merits in a single approach. For example, neural-network-based models such as RBF do exhibit superior prediction performance only as long as the input data point lies well within the domain of the input training dataset, otherwise it fails. On the other hand, linear and sparse prediction models, in general, show much better extrapolative performance, even though they are unable to capture more complex, non-linear dynamics. In other words, by toggling between the robust linear models and the more sensitive but also more effective non-linear ones, a superior approach to load time series prediction can be constructed.

In order to obtain the best possible performance of each sub-model, their optimal training configuration has to be determined. Starting with the simpler methods used, a linear and an SR model are trained by least squares and fast iterative shrinkage thresholding algorithm, respectively, the latter being a faster implementation of the corresponding iterative shrinkage thresholding algorithm used for load forecasting [36]. In the case of the sparse coding approach, sparsity is induced by the ℓ_2 norm and the regularization parameter was set by trial and error to 0.01. Subsequently, a random forest regressor is employed, where the number of decision trees is selected to be 15 so as to keep the training time at a reasonable level without reducing its predictive ability. As regards the non-linear methods, an SVR model with Gaussian kernel function was developed [92], using sequential minimal optimization for training and Bayesian optimization to optimize the model's hyperparameters [93]. Two NN models are also introduced, featuring two different architectures. The first one is a two-layered MLP network trained by the Levenberg–Marquardt backpropagation algorithm [94], following a 10-fold cross-validation. The neurons of each layer are chosen by trial and error as 20 and 10. It is noted that, in order to compensate for the performance dependence of the MLP training methods to initialization, the training procedure was conducted 10 different times, with different randomly initialized weights of the network. The second NN uses an RBF architecture and is trained using the fuzzy means technique [95], an algorithm that has found many successful applications due to the increased accuracy it provides [96–98] combined with fast training times [99]. In this work, the FM algorithm has been tested for a range of fuzzy sets between 4 and 15. When deployed online, the proposed approach evaluates a MAE metric on a rolling window of past predictions coming from a pool of trained models in order to create a weight vector for the next timestep prediction. An important item of the proposed method to be specified is the length of the rolling window. It can be easily inferred that this depends not only on the prediction horizon but also on the statistical properties of the predicted variable (a more volatile, non-stationary time series would require shorter rolling window horizons). Once the model pool has been populated by trained models, the optimum length of the rolling window is calculated in an exhaustive search manner over the same validation data in the range of 3–15 regressive timesteps. The proposed method operates as follows: For each timestep k , all trained models in the pool are evaluated concurrently. Their current prediction performance is assessed by applying the MAE metric on their previous predictions up to a rolling time window of length h_w

$$MAE_i(k) = \frac{\sum_{j=0}^{h_w-1} |\hat{y}_i(k-j) - y(k-j)|}{h_w} \quad (12)$$

where $\hat{y}_i(k)$ are the predictions of the i -th model and y are the actual values of the timeseries at timestep k . Then, the MAE metric is used to calculate the prediction weight of each model for the next timestep $k + 1$.

$$w_i(k+1) = \frac{MAE_i^{-1}(k)}{\sum_{i=1}^N MAE_i^{-1}(k)} \quad (13)$$

where MAE_i is the MAE of the i -th prediction model, N is the total number of models in the model pool, and w_i is the prediction weight for the next timestep. The prediction output of the proposed method is calculated as the weighted sum of the model predictions $\hat{y}_i$

$$\hat{y}(k+1) = \sum_{i=1}^N w_i(k+1) \hat{y}_i(k+1) \quad (14)$$

A snapshot of a two-model example version of the proposed method is shown in Figure 4. Note that the proposed method combines the strengths of the individual models by placing greater weight on the current better-performing model for the time window of length h_w . At first, both $\hat{y}_1$ and $\hat{y}_2$ models appear ineffective as individual predictors of the y time series. However, after closer inspection, $\hat{y}_2$ performs better for the first half of y , while $\hat{y}_1$ for the second half. By placing greater weight on the model with the best past prediction performance within the horizon h_w , the proposed method is able to toggle towards the best available model for the current circumstance. The result is an overall superior prediction performance.

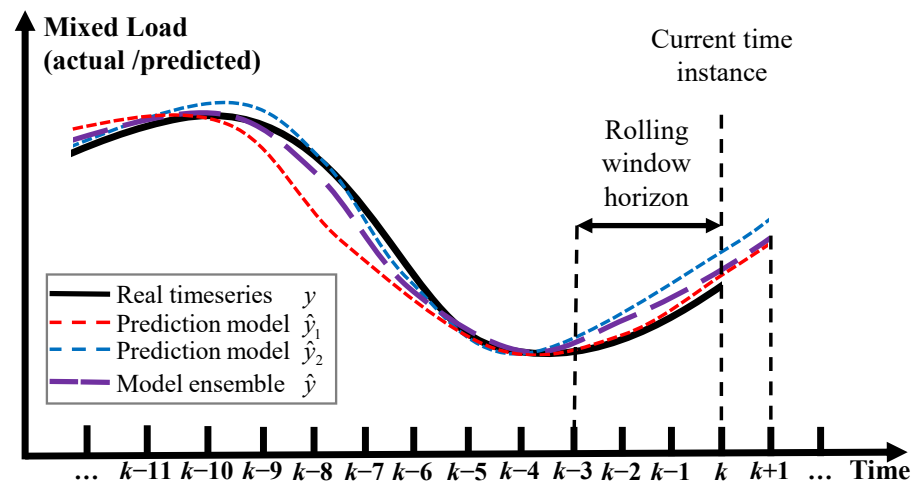


Figure 4. Schematic for a two-model version of the proposed method, where y denotes the real load, $\hat{y}_i$ the prediction of the i -th model, $\hat{y}$ the weighted prediction and k the current timestep. The ensemble model recognizing the superiority of $\hat{y}_1$ over $\hat{y}_2$, within the rolling window adapts its weights accordingly, achieving highly accurate prediction for the next timestep $k+1$.

3. Case Study

3.1. Problem and Data Description

The main goal of this paper is to develop a methodology in order to implement a load forecasting tool able to provide accurate mixed load predictions over several different time horizons and in particular 15 min, 1-h, 2-h, 3-h, 6-h, and 24-h. This case study makes use of real data from a high voltage/medium voltage substation located in mainland Europe, measured during the years 2017–2018. The MV distribution network contains multiple photovoltaic sites. As a result, the data measurements in question constitute mixed power-load recordings, which correspond to the mixed AP demand of the distribution grid from the transmission grid. The load measurements have been recorded every minute and contain the mixed AP demand, as well as cloud coverage, wind speed, humidity, and temperature, as measured from the substation's weather station. Due to practical concerns, individual power generation or weather data from the aforementioned photovoltaic sites should not be taken into account for the creation of the input dataset since these will normally not be available for a real-life implementation. In short, in this work, we rely on the substation's historical measurements of load and weather conditions in order to create a prediction model of the mixed AP demand of the grid.

3.2. Data Preprocessing and Model Training

Unavoidably, the substation measurements contain large periods of missing or corrupt data owing to sensor downtime or malfunction. For the scope of this case study, no missing data imputation has been performed—instead, corrupted data and outlier removal was the main focus of the preprocessing operation. Due to the sheer size of the dataset, manual preprocessing was impossible, mandating the creation of a bad data detection routine. Corrupted values were decidedly easy to detect since the corresponding AP signal exhibited unusually low variance around a constant value. However, outlier values on mixed load data were a challenge to successfully handle—a review of the challenges of this topic, as well as effective techniques, is available on [100]. The chosen technique must be sufficiently effective at classifying outliers in data, while avoiding false positives. In this case study, a rolling median window threshold approach is used, as it was found to compromise well between the aforementioned points. A two-day snapshot from the application of this algorithm to raw electrical load data is presented in Figure 5. The outliers usually originate from noisy sensor readings [101]. As part of data preprocessing, a resampling step also took place, where each sample was defined as an average of 15 one-minute measurements.

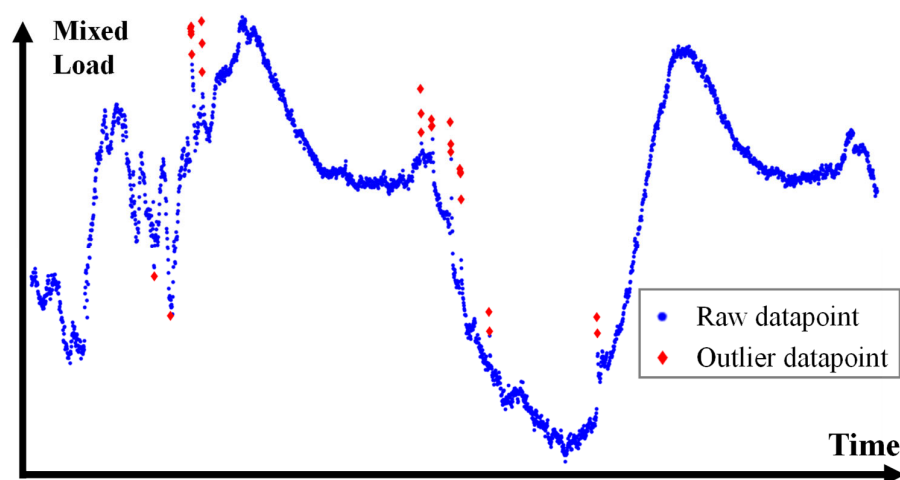


Figure 5. Operation of the rolling median threshold outlier detection algorithm. The data points marked as outliers exceed the median value of the time window multiplied by a user-specified threshold factor.

The task of input variable selection is closely related to the prediction horizon. All models developed in the context of this study are considered autoregressive with exogenous variables, as they use inputs that consist of previous values of the output and weather data. A set of inputs was initially constructed for each prediction horizon based on the literature. Subsequently, the contribution of these variables to the prediction accuracy improvement was examined by trial and error, sometimes leading to shorter input sets for some of the horizons. Alternatively, other approaches, such as gradient boosting decision tree and Pearson correlation coefficient [102], attention mechanism [103], or Exploratory Data Analysis [104], are considered to have an effective contribution during input features reduction and selection. However, it is important to note that for each horizon, inputs remain the same for all machine-learning methods used in the present study.

The selected input variables which all models accept could be divided into 4 categories, as described in Table 1, namely (a) current and past AP values, (b) difference between current and past AP values, (c) average of past AP values, and (d) weather measurements. It has to be noted that $p^{(t)}$ values contain the current and past, average and difference measures of the AP values, $\hat{p}^{(t+s)}$ is the output, i.e., the mixed power load s fifteen-minute intervals ahead, whereas $w^{(t)}$ components contain the respective weather-related inputs of cloud coverage, wind speed, humidity, and temperature, respectively.

Table 1. Description of training variables of the forecasting models for the different prediction horizons examined in the case study. Each row of the table refers to the different groups of input variables, whereas the last row refers to the output variable.

Prediction Horizon	15 min ($t+1$)	1 h ($t+4$)	2 h ($t+4$)	3 h ($t+12$)	6 h ($t+24$)	24 h ($t+96$)
Current and past AP measures	$p^{(t-i)},$ $i = 0, 95, 671$	$p^{(t-i)},$ $i = 0, 4, 92, 668$	$p^{(t-i)},$ $i = 0, 8, 88, 664$	$p^{(t-i)},$ $i = 0, 12, 84, 660$	$p^{(t-i)},$ $i = 0, 24, 72, 648$	$p^{(t-i)},$ $i = 0, 96, 576$
Average AP measures	$\frac{\sum_{n=0}^3 p^{(t-n)}}{4}$	$\frac{\sum_{n=0}^3 p^{(t-n)}}{4}$	$\frac{\sum_{n=0}^7 p^{(t-n)}}{8}$	$\frac{\sum_{n=0}^{11} p^{(t-n)}}{12}$	$\frac{\sum_{n=0}^{23} p^{(t-n)}}{24}$	$\frac{\sum_{n=0}^{95} p^{(t-n)}}{96}$
Difference AP measures	$p^{(t)} - p^{(t-i)},$ $i = 1$	$p^{(t)} - p^{(t-i)},$ $i = 4$	$p^{(t)} - p^{(t-i)},$ $i = 8$	$p^{(t)} - p^{(t-i)},$ $i = 12$	$p^{(t)} - p^{(t-i)},$ $i = 24$	$p^{(t)} - p^{(t-i)},$ $i = 96$
Weather measures	$w^{(t+i)},$ $i = 0$	$w^{(t+i)},$ $i = 4$	$w^{(t+i)},$ $i = 4, 8$	$w^{(t+i)},$ $i = 4, 8, 12$	$w^{(t+i)},$ $i = 16, 20, 24$	$w^{(t+i)},$ $i = 88, 92, 96$
Future AP forecasts (output variable)	$\hat{p}^{(t+1)}$	$\hat{p}^{(t+4)}$	$\hat{p}^{(t+8)}$	$\hat{p}^{(t+12)}$	$\hat{p}^{(t+24)}$	$\hat{p}^{(t+96)}$

The choice of the particular set of input variables can be justified as follows: The fact that electric load time series presents a strong dependency on previous values [47] strengthens the selection of such input variables in the form of (a). Trying to capture the trend of electrical load, differences (b) between current and previous AP values are frequently employed [31]. The implementation of past value averages (c) is also quite important, according to the literature [48]. Finally, the introduction of weather data (d) is undoubtedly an improving factor in the predictions [23,24,105]. At this point, it is important to mention that during the training stage of the forecasting model, the weather inputs $w^{(t)}$ are introduced as measured values of actual weather data acquired at the t time index. On the contrary, in an online implementation of the model, future weather data $w^{(t+s)}$ will be unknown and replaced by weather predictions, therefore introducing additional uncertainty.

Once the preprocessing stage has been completed and input variables have been selected, the dataset was partitioned in a yearly manner in order to select the training datasets. At this point, an important consideration should be made. As mentioned in the introductory section, the load time series consists of a load and generation component. The statistical properties of both of these components are not static in relation to time, especially on a long-term scale. The network physically expands, incorporating more consumers as well as RES generators, each with different load and generation profiles, respectively. Therefore, it makes sense to select training datasets as close to the actual prediction interval as possible. Since the available data concern two successive years, the data corresponding to 2017 were selected as the training subset, and the data corresponding to 2018 were selected as the testing dataset. A point worth mentioning is that no permutation step is taking place before training. This means that the data used for testing are considered completely unseen for the proposed model, yielding a more reliable forecasting model. Due to confidentiality reasons, the real and predicted mixed load values have been normalized in order to be presented. Finally, it should be noted that models that require a validation step during training, namely models based on MLP and RBF NNs, do so using 10-fold cross-validation, while in the case of models that require multiple training runs for each training seed (see MLP), the best-performing model on the validation data is kept. An overview of the implementation of the proposed model is provided in Figure 6, which illustrates, in the form of a block diagram, the entire sequence of steps that take place, starting from the acquisition of the raw AP data from the substation to the derivation of the final forecasts. It has to be highlighted that this figure is generic and does not refer to a particular prediction horizon.

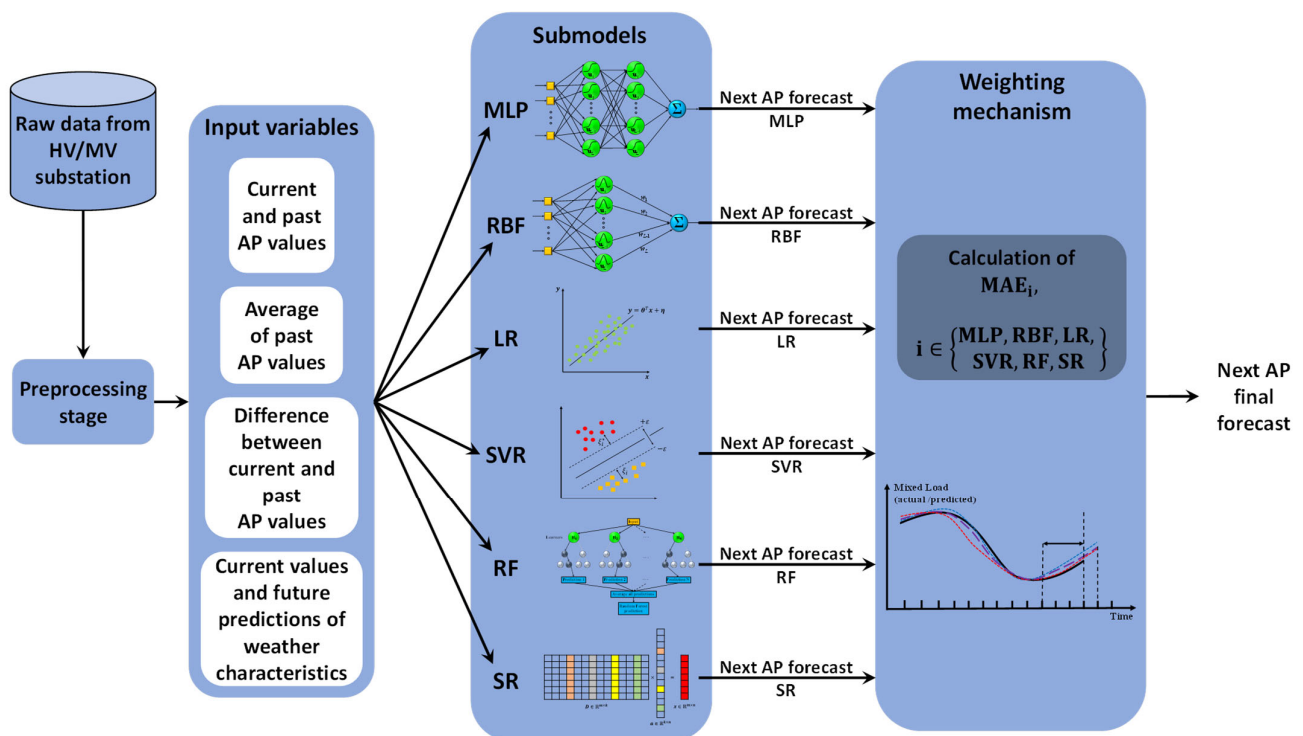


Figure 6. Overview of the proposed model ensemble. Its application in mixed load forecasting comprises a series of steps, i.e. raw data acquisition, data preprocessing, collection of input variables, splitting of the dataset in a training and a testing subset, training of submodels, generation of the next AP forecast by each submodel, weighting of the individual predictions, and, lastly, calculation of the next AP final forecast.

At this point, it should be mentioned that in order to evaluate the accuracy of the proposed method, it was considered appropriate to compare it with a model ensemble from the literature. To be more specific, we employed a method proposed for load forecasting based on an ensemble of multiple MLP neural networks [76]. Consequently, following the experimental protocol described in this work, a number of feed-forward NNs, with a single hidden layer, were trained on 14 different random initializations of the weights. For each initialization, the number of neurons in the hidden layer ranged from 3 to 50. The hyperbolic tangent sigmoid function was selected as the transfer function among the NNs' layers, while all NNs were trained using the resilient backpropagation algorithm. The neural networks were arranged in ascending order with respect to the MAPE error on a common validation set, which, in this case, was defined as 20% of the training dataset. Then, the networks corresponding to the first 5 MAPE errors were selected, and the final forecasts were obtained by averaging the individual forecasts of these 5 models.

4. Results

In this section, the results of extensive simulations of the proposed model are presented. A set of scatterplots is shown in Figure 7a–f, representing the actual versus the predicted values mixed load values for 1, 2, 3, 6, and 24-h-ahead horizons, respectively, through the whole testing dataset. The diagonal line implies a complete match between real values and forecasts. The axes are presented in units of normalized AP.

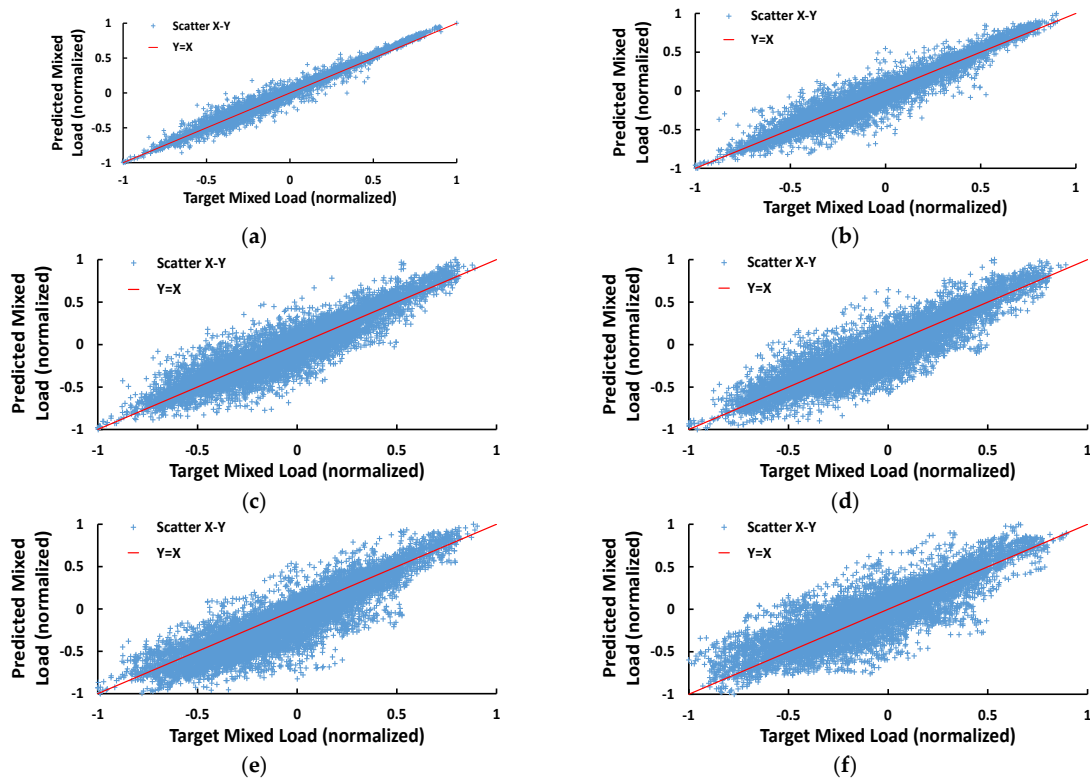


Figure 7. Scatterplots of actual versus predicted mixed load for (a) 15-min, (b) 1-h, (c) 2-h, (d) 3-h, (e) 6-h, and (f) 24-h ahead prediction. The predicted values residing on the diagonal line are identical to the actual values. Each mark refers to a data point and shows the deviation of its predicted value from its actual value.

Additional results are provided in Table 2, which contains information about the forecasting performance of the proposed method in comparison to the individual machine-learning methods comprising the model pool. In order to distinguish the results for different prediction time horizons, the table is divided into sections. The accuracy of model predictions is evaluated through the correlation coefficient (R^2), RMSE and MAE, considering them as representative and efficient criteria [106]. For comparative reasons, the table also contains the values of the indices for all submodels, as well as their percentage of ranking in the first place. This quantity, labeled as “Rank 1” in Table 2, denotes how many times each submodel scored the 1st rank among all submodels, i.e., achieved the lowest MAE.

The aforementioned form of ranking of the submodels can be seen graphically in Figure 8. More specifically, each one of Figure 8a–f refers to 15 min, 1, 2, 3, 6, and 24-h prediction horizons, respectively. Each one of these subfigures contains 6 pie charts, denoting 1st to 6th rank for the models. To be more specific, each pie chart shows the percentages corresponding to how many times each submodel ranked in the respective place, according to its weighted MAE. For example, the 2nd pie of Figure 8a implies that for 15 min-ahead forecasting, the MLP submodel ranked in the 2nd place among all models with a percentage of 17%, the SR submodel with a percentage of 21%, etc. Finally, analytical graphs are provided for each prediction time horizon, with Figure 9a1–f1 to depict forecasts of 15 min, 1, 2, 3, 6, and 24 h-ahead, respectively, where a randomly chosen 12-h time window (from 09:00 to 21:00) of real AP values and the respective predictions are shown for an arbitrarily chosen day belonging to the testing subset (the same day and the same window is used for all horizons). These graphs are accompanied by Figure 9a2–f2, which indicates which submodel has the largest weight for every predicted data point using a bar plot.

Table 2. Performance of the proposed multi-model scheme, the MLP model ensemble of [76], and individual machine-learning models for each prediction horizon. The values of MAE, RMSE, and R^2 , achieved by each model, are presented, as well as the percentage that each submodel achieved the lowest MAE among all submodels.

Method	R^2	MAE	RMSE	Rank1
15 min				
Proposed	0.98613	0.26120	0.4703	-
MLP ensemble	0.9852	0.2760	0.4869	-
MLP	0.98568	0.26936	0.4782	19.25%
RBF	0.98574	0.27095	0.4773	19.94%
LR	0.98562	0.26700	0.4793	10.49%
SVR	0.98541	0.26931	0.4829	14.23%
RF	0.98373	0.29531	0.5071	24.27%
SR	0.98561	0.26715	0.4795	11.82%
1 h				
Proposed	0.93793	0.60224	0.9946	-
MLP ensemble	0.9344	0.6330	1.0240	-
MLP	0.91697	0.66794	1.1500	21.56%
RBF	0.93253	0.64235	1.0374	20.77%
LR	0.93168	0.64174	1.0438	8.91%
SVR	0.93008	0.65376	1.0562	10.70%
RF	0.92912	0.67311	1.0614	20.79%
SR	0.93045	0.64079	1.0532	17.27%
2 h				
Proposed	0.88147	0.88279	1.3767	-
MLP ensemble	0.8838	0.8965	1.3721	-
MLP	0.84455	0.99479	1.5854	20.32%
RBF	0.87052	0.96255	1.4472	18.32%
LR	0.87233	0.93356	1.4377	11.20%
SVR	0.86949	0.93765	1.4537	11.38%
RF	0.86653	0.96596	1.4675	22.94%
SR	0.86953	0.93189	1.4534	15.92%
3 h				
Proposed	0.84143	1.0599	1.5871	-
MLP ensemble	0.8359	1.0859	1.6192	-
MLP	0.78486	1.2504	1.8538	18.08%
RBF	0.82483	1.1367	1.6727	20.14%
LR	0.82241	1.1270	1.6843	8.74%
SVR	0.81914	1.1512	1.6997	11.82%
RF	0.81895	1.1391	1.7006	23.67%
SR	0.81893	1.1229	1.7007	17.54%
6 h				
Proposed	0.83251	1.1144	1.6462	-
MLP ensemble	0.8272	1.1951	1.6888	-
MLP	0.83036	1.1758	1.6733	20.77%
RBF	0.80289	1.2848	1.8037	21.31%
LR	0.77800	1.3308	1.9141	10.34%
SVR	0.75400	1.4300	2.0150	16.42%
RF	0.81341	1.2119	1.7549	21.08%
SR	0.77373	1.3413	1.9325	10.08%
24 h				
Proposed	0.78474	1.1835	1.8174	-
MLP ensemble	0.7827	1.2372	1.8468	-
MLP	0.78073	1.2313	1.8553	21.93%
RBF	0.73576	1.4119	2.0367	21.83%
LR	0.75712	1.3188	1.9526	11.16%
SVR	0.73669	1.3031	2.0331	16.82%
RF	0.76487	1.2694	1.9212	16.71%
SR	0.74761	1.3419	1.9905	11.56%

Note: The quality metrics of the proposed methodology are denoted by text in colour.



Figure 8. Pie charts depicting the ranking of the submodels included in the proposed model ensemble for (a) 15-min, (b) 1-h, (c) 2-h, (d) 3-h, (e) 6-h, and (f) 24-h ahead prediction. Each pie chart refers to a ranking position and shows the percentage that each submodel was ranked in that position. Each submodel is represented by a different color and pattern.

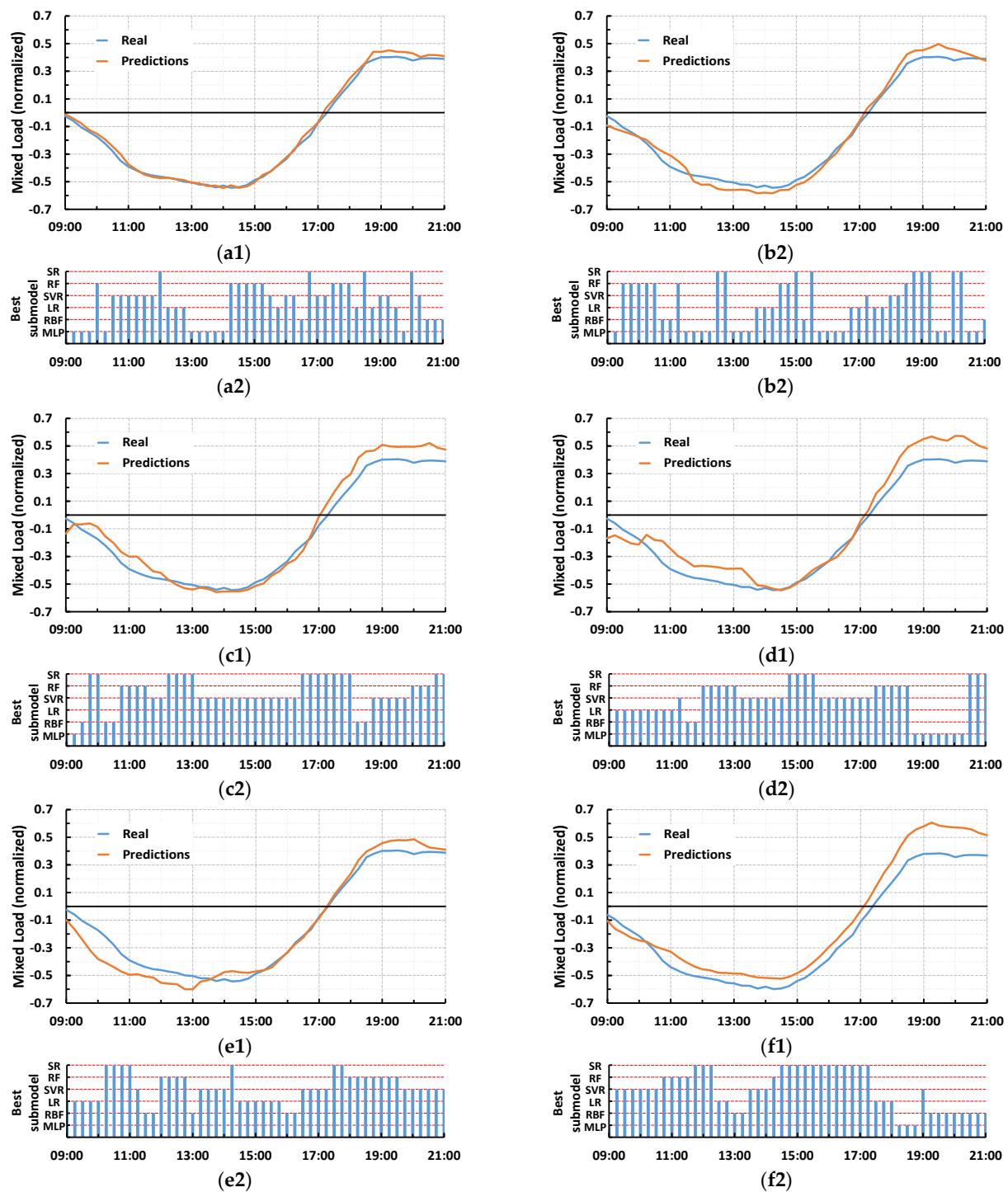


Figure 9. Results for a randomly selected 12-h window for (a) 15-min, (b) 1-h, (c) 2-h, (d) 3-h, (e) 6-h, and (f) 24-h ahead predictions. Subgraphs labeled 1 depict actual and predicted value results, whereas subgraphs labeled 2 depict the best submodel performance results.

5. Discussion

In the context of the case study, multiple experiments were conducted, and the results are explained and discussed here. At this point, we should point out that providing accurate predictions is indeed a challenging task due to both grid and data-related reasons. First, the system's expandability can be a limiting factor for the accuracy of future forecasts. At the same time, this is reinforced by inherent characteristics of the load time series, such as non-linearity and uncertainty. In the face of these challenges, the proposed method

seems to be quite effective, providing reliable predictions. From Figure 7a–f, we can draw conclusions about the quality of predictions. When the prediction time horizon is too short (Figure 7a), the forecast error is distributed close to the diagonal line, which implies quite accurate predictions. While we are trying to increase the prediction horizon, the forecasts are getting less accurate (Figure 7b–f), as obviously, the pairs of real and predicted values are scattered further from the ideal line.

Looking at Table 2, we observe that the proposed model outmatches all individual submodels, and the competitive MLP model ensemble in terms of MAE, and R^2 and RMSE. Moreover, this conclusion applies to all prediction time horizons. As the prediction horizon gets longer, the forecasting error increases, which is absolutely reasonable. The only exceptions are the R^2 and RMSE values obtained by the MLP model ensemble for 2 h prediction horizon, which slightly exceeds those of the proposed model. However, these differences cannot be considered significant as they are marginal, while on the other hand, the corresponding value of the MAE index clearly favors the proposed method. A result worth mentioning is the improvement of the multi-model performance over the current best sub-model that occurs in most cases while the horizon is getting longer. More specifically, the reduction of MAE that the proposed approach achieves over the best of the individual models ranges from 0.03411 to 0.3156. Such an improvement in performance could be partly explained by the occurrence of uncertainty in the load time series. As the prediction horizon is getting longer, the level of uncertainty is also increased, which is better addressed by the ensemble model than each individual submodel alone.

Regarding the efficiency of the individual models of the pool, the results of MAE, RMSE, and R^2 show that there is not just one model to prevail over the others in all cases. For the shorter prediction horizons and, more specifically, up to 3 h, LR and SR appear to achieve marginally smaller forecasting errors than their non-linear counterparts. Although the non-linearities are an intrinsic characteristic of mixed load [107], this behavior becomes more apparent as the prediction horizon is getting longer. As a result, models which are based on LR are able to provide robust results for very short-term forecasts. On the other hand, one major advantage of neural networks is their capability of modelling non-linear systems. An important observation is that neural networks appear to perform better for longer prediction horizons, and this can be attributed to the fact that, as the prediction horizon is getting longer, the non-linear properties of the load are becoming more dominant. Therefore, when predictions for longer horizons are required, MLP neural networks take the lead. However, the same does not apply to RBF networks. As stated above, in order for RBF networks to perform well, dense and suitable data are required. Consequently, their performance is reduced for 24-h prediction horizons, where the input information is poorer due to the resampling process. Although the remaining models of the pool, SVR and RF networks, present a moderate predictive capability, they contribute positively to the overall performance of the proposed model. This conclusion confirms our claim of the need to use multiple models in order to enhance the reliability of load predictions.

Looking at the results of actual and predicted values in Figure 9 we confirm that, as the time horizon increases, accurate load forecasting becomes more and more difficult. Continuing with the subfigures of Figure 9 that show the alternation between submodels in order to maintain the accuracy of predictions, we conclude that the weighting mechanism of the proposed model seems to perform adequately regardless of the time horizon. It can easily be seen that quite reliable forecasts are obtained during the steady rise or fall of the actual AP values. On the contrary, predictions become less accurate when the AP presents great fluctuation.

Several quite interesting conclusions can also be drawn from the pie charts in Figure 8. Each percentage in the pies represents the degree to which the respective model yielded the highest weight or equivalently the lowest MAE. The highest percentages of the first rank (above 18%) belong to MLP, RBF, and RF, and this applies for all horizons except that of 24 h, where SVR takes the place of RF. RF, in particular, scores lower MAE most of the time when the prediction horizon does not exceed 3 h. Beyond that point, RBF neural networks

outperform the rest of the submodels. An interesting observation is that the aforementioned models have equally high percentages in the sixth rank. Thus, these methods either achieve very good or poor performance. This observation is quite significant and strongly enhances the usefulness and effectiveness of our proposed method. The percentages of the rest of the pool models are, in most cases, divided into the intermediate rankings, with the exception of the high percentage of SVR in the sixth rank for the 6-h horizon.

6. Conclusions and Prospects

Achieving reliable electric load forecasts is of paramount importance for the smooth operation of electric power grids. However, the intrinsic volatility of the electric load makes its prediction particularly hard. Therefore, we assume that the load behavior is influenced by multiple input variables, which differ depending on the data to be predicted.

The mixed load forecasting task has been addressed by a variety of machine-learning methods, and it has been observed that none is able to provide equally accurate results for any testing dataset. In the present study, a mixed power-load forecasting model is introduced, which employs the predictions coming from several individual models, namely MLP and RBF neural networks, LR, SVR, RF, and SR. These forecasts are weighted based on how accurate they have been and then added to calculate the final forecast value. The proposed model provides predictions for different time horizons, spanning from 15 min to 24 h. The extended results presented using real data sensed from a high voltage/medium voltage substation show the superiority of this novel approach compared to all the individual models as well as an MLP model ensemble for every prediction horizon tested. Thus, the proposed multi-model forecasting scheme constitutes a powerful method capable of greatly enhancing the operation of the modern electricity grid, with potential practical applications in network planning, operation, and management.

It should be noted here that a limitation of the present study is that it did not involve predictions for long-term horizons. Although investigating longer prediction horizons is outside the scope of this work, we believe the proposed model ensemble could serve as the basis for designing such a tool. On the other hand, it is quite probable that a different set of input variables, presenting higher correlation with the long-term evolution of the mixed load would be needed in this case.

Driven by the remarkable performance of the proposed methodology in mixed load forecasting, its application could be extended to other critical sectors of the smart grid, such as forecasting the electricity price and the production from RES, in order to more efficiently schedule conventional sources. A fruitful application would also be to forecast the residential demand or the aggregated load corresponding to several substations. Another promising direction for future research towards this direction includes the integration of Graph Neural Networks, which have been proved to be a promising candidate due to their ability to successfully interpret spatiotemporal features of the input data.

Author Contributions: Conceptualization, E.N.Z., N.-A.I.L. and A.A.; methodology, N.G., M.P., E.N.Z., N.-A.I.L. and A.A.; software, N.G. and M.P.; validation, N.G., M.P., M.S. and N.-A.I.L.; formal analysis, E.N.Z. and A.A.; investigation, N.G., M.P. and M.S.; resources, N.-A.I.L., E.N.Z. and A.A.; data curation, N.G. and M.P.; writing—original draft preparation, N.G., M.P., M.S., E.N.Z., N.-A.I.L. and A.A.; writing—review and editing, N.G., M.P., M.S., E.N.Z., N.-A.I.L. and A.A.; visualization, N.G. and M.S.; supervision, E.N.Z. and A.A.; project administration, A.A.; funding acquisition, N.-A.I.L. and A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH—CREATE—INNOVATE (project code: T1EDK-00244).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are not publicly available due to confidentiality and privacy reasons.

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

Nomenclature

AP	active power
LR	linear regression
ML	Machine-learning
MAE	mean absolute error
MLP	multi-layer perceptron
NN	neural network
RBF	radial basis function
RF	random forests
RES	renewable energy sources
SBL	sparse Bayesian learning
SR	sparse regression
SVR	support vector regression

References

1. Livanos, N.-A.I.; Hammal, S.; Giamarelos, N.; Alifragkis, V.; Psomopoulos, C.S.; Zois, E.N. OpenEdgePMU: An Open PMU Architecture with Edge Processing for Future Resilient Smart Grids. *Energies* **2023**, *16*, 2756. [\[CrossRef\]](#)
2. Chen, H.; Xuan, P.; Wang, Y.; Tan, K.; Jin, X. Key Technologies for Integration of Multitype Renewable Energy Sources—Research on Multi-Timeframe Robust Scheduling/Dispatch. *IEEE Trans. Smart Grid* **2016**, *7*, 471–480. [\[CrossRef\]](#)
3. Papadimitrakis, M.; Giamarelos, N.; Stogiannos, M.; Zois, E.N.; Livanos, N.A.; Alexandridis, A. Metaheuristic Search in Smart Grid: A Review with Emphasis on Planning, Scheduling and Power Flow Optimization Applications. *Renew. Sustain. Energy Rev.* **2021**, *145*, 111072. [\[CrossRef\]](#)
4. Salkuti, S.R. Day-Ahead Thermal and Renewable Power Generation Scheduling Considering Uncertainty. *Renew. Energy* **2019**, *131*, 956–965. [\[CrossRef\]](#)
5. Gong, L.; Wang, X.; Tian, M.; Yao, H.; Long, J. Multi-Objective Optimal Planning for Distribution Network Considering the Uncertainty of PV Power and Line-Switch State. *Sensors* **2022**, *22*, 4927. [\[CrossRef\]](#)
6. Fotopoulou, M.; Rakopoulos, D.; Blanas, O.; Psomopoulos, S.; Munteanu, R.A.; Agavanakis, K. Day Ahead Optimal Dispatch Schedule in a Smart Grid Containing Distributed Energy Resources and Electric Vehicles. *Sensors* **2021**, *21*, 7295. [\[CrossRef\]](#)
7. Zhang, N.; Kang, C.; Xia, Q.; Ding, Y.; Huang, Y.; Sun, R.; Huang, J.; Bai, J. A Convex Model of Risk-Based Unit Commitment for Day-Ahead Market Clearing Considering Wind Power Uncertainty. *IEEE Trans. Power Syst.* **2015**, *30*, 1582–1592. [\[CrossRef\]](#)
8. Divenyi, D.; Polgari, B.; Raisz, D.; Sleisz, A.; Sores, P. Special Session on Proposal of a New European Co-Optimized Energy and Ancillary Service Market Design—Part II. In Proceedings of the International Conference on the European Energy Market, EEM, Porto, Portugal, 6–9 June 2016; Volume 2016.
9. Tayab, U.B.; Zia, A.; Yang, F.; Lu, J.; Kashif, M. Short-Term Load Forecasting for Microgrid Energy Management System Using Hybrid HHO-FNN Model with Best-Basis Stationary Wavelet Packet Transform. *Energy* **2020**, *203*, 117857. [\[CrossRef\]](#)
10. Abdelsalam, A.A.; Zedan, H.A.; ElDesouky, A.A. Energy Management of Microgrids Using Load Shifting and Multi-Agent System. *J. Control Autom. Electr. Syst.* **2020**, *31*, 1015–1036.
11. Basir, R.; Qaisar, S.; Ali, M.; Aldwairi, M.; Ashraf, M.I.; Mahmood, A.; Gidlund, M. Fog Computing Enabling Industrial Internet of Things: State-of-the-Art and Research Challenges. *Sensors* **2019**, *19*, 4807. [\[CrossRef\]](#)
12. Zhang, J.; Han, H. A Lightweight and Privacy-Friendly Data Aggregation Scheme against Abnormal Data. *Sensors* **2022**, *22*, 1452. [\[CrossRef\]](#)
13. Papadimitrakis, M.; Kapnopoulos, A.; Tsavartzidis, S.; Alexandridis, A. A Cooperative PSO Algorithm for Volt-VAR Optimization in Smart Distribution Grids. *Electr. Power Syst. Res.* **2022**, *212*, 108618. [\[CrossRef\]](#)
14. Haupt, S.E.; Dettling, S.; Williams, J.K.; Pearson, J.; Jensen, T.; Brummet, T.; Kosovic, B.; Wiener, G.; McCandless, T.; Burghardt, C. Blending Distributed Photovoltaic and Demand Load Forecasts. *Sol. Energy* **2017**, *157*, 542–551. [\[CrossRef\]](#)
15. Pan, Y.; Zheng, J.; Yang, Y.; Zhu, R.; Zhou, C.; Shi, T. An Electricity Load Forecasting Approach Combining DBN-Based Deep Neural Network and NAR Model for the Integrated Energy Systems. In Proceedings of the 2019 IEEE International Conference on Big Data and Smart Computing, BigComp, Kyoto, Japan, 27 February–2 March 2019.
16. Kaur, A.; Nonnenmacher, L.; Coimbra, C.F.M. Net Load Forecasting for High Renewable Energy Penetration Grids. *Energy* **2016**, *114*, 1073–1084. [\[CrossRef\]](#)
17. Van der Meer, D.W.; Munkhammar, J.; Widén, J. Probabilistic Forecasting of Solar Power, Electricity Consumption and Net Load: Investigating the Effect of Seasons, Aggregation and Penetration on Prediction Intervals. *Sol. Energy* **2018**, *171*, 397–413. [\[CrossRef\]](#)
18. Qin, J.; Zhang, Y.; Fan, S.; Hu, X.; Huang, Y.; Lu, Z.; Liu, Y. Multi-Task Short-Term Reactive and Active Load Forecasting Method Based on Attention-LSTM Model. *Int. J. Electr. Power Energy Syst.* **2022**, *135*, 107517. [\[CrossRef\]](#)

19. Theodoridis, S. *Machine Learning: A Bayesian and Optimization Perspective*; Academic Press: Cambridge, MA, USA, 2015; ISBN 9780128017227.
20. Wang, C.; Grozev, G.; Seo, S. Decomposition and Statistical Analysis for Regional Electricity Demand Forecasting. *Energy* **2012**, *41*, 313–325. [\[CrossRef\]](#)
21. Gerossier, A.; Girard, R.; Kariniotakis, G.; Michiorri, A. Probabilistic Day-Ahead Forecasting of Household Electricity Demand. *CIREN Open Access Proc. J.* **2017**, *2017*, 2500–2504. [\[CrossRef\]](#)
22. Yang, Y.; Li, S.; Li, W.; Qu, M. Power Load Probability Density Forecasting Using Gaussian Process Quantile Regression. *Appl. Energy* **2018**, *213*, 499–509. [\[CrossRef\]](#)
23. Vu, D.H.; Muttaqi, K.M.; Agalgaonkar, A.P. Assessing the Influence of Climatic Variables on Electricity Demand. In Proceedings of the 2014 IEEE PES General Meeting | Conference & Exposition, National Harbor, MD, USA, 27–31 July 2014; pp. 1–5.
24. Xie, J.; Chen, Y.; Hong, T.; Laing, T.D. Relative Humidity for Load Forecasting Models. *IEEE Trans. Smart Grid* **2018**, *9*, 191–198. [\[CrossRef\]](#)
25. Alzate, C.; Sinn, M. Improved Electricity Load Forecasting via Kernel Spectral Clustering of Smart Meters. In Proceedings of the IEEE International Conference on Data Mining, ICDM 2013, Dallas, TX, USA, 7–10 December 2013; pp. 943–948.
26. Rossi, M.; Brunelli, D. Electricity Demand Forecasting of Single Residential Units. In Proceedings of the 2013 IEEE Workshop on Environmental, Energy and Structural Monitoring Systems, EESMS, Trento, Italy, 11–12 September 2013.
27. Damrongkulkarnjorn, P.; Churueang, P. Monthly Energy Forecasting Using Decomposition Method with Application of Seasonal ARIMA. In Proceedings of the 2005 International Power Engineering Conference, Singapore, 29 November–2 December 2005. [\[CrossRef\]](#)
28. Ould Mohamed Mahmoud, M.; Mhamdi, F.; Jaïdane-Saïdane, M. Long Term Multi-Scale Analysis of the Daily Peak Load Based on the Empirical Mode Decomposition. In Proceedings of the 2009 IEEE Bucharest PowerTech, Bucharest, Romania, 28 June–2 July 2009. [\[CrossRef\]](#)
29. Clements, A.E.; Hurn, A.S.; Li, Z. Forecasting Day-Ahead Electricity Load Using a Multiple Equation Time Series Approach. *Eur. J. Oper. Res.* **2016**, *251*, 522–530. [\[CrossRef\]](#)
30. Elrazaz, Z.S.; Mazi, A.A. Unified Weekly Peak Load Forecasting for Fast Growing Power System. *IEE Proc. C Gener. Transm. Distrib.* **2010**, *136*, 29. [\[CrossRef\]](#)
31. Amini, M.H.; Kargarian, A.; Karabasoglu, O. ARIMA-Based Decoupled Time Series Forecasting of Electric Vehicle Charging Demand for Stochastic Power System Operation. *Electr. Power Syst. Res.* **2016**, *140*, 378–390. [\[CrossRef\]](#)
32. Yu, C.N.; Mirowski, P.; Ho, T.K. A Sparse Coding Approach to Household Electricity Demand Forecasting in Smart Grids. *IEEE Trans. Smart Grid* **2017**, *8*, 738–748. [\[CrossRef\]](#)
33. Yang, D.; Xu, L.; Gong, S.; Li, H.; Peterson, G.D.; Zhang, Z. Joint Electrical Load Modeling and Forecasting Based on Sparse Bayesian Learning for the Smart Grid. In Proceedings of the 2011 45th Annual Conference on Information Sciences and Systems, CISS 2011, Baltimore, MD, USA, 23–25 March 2011.
34. Sun, X.; Wang, X.; Wu, J.; Liu, Y. Hierarchical Sparse Learning for Load Forecasting in Cyber-Physical Energy Systems. In Proceedings of the Conference Record—IEEE Instrumentation and Measurement Technology Conference, Minneapolis, MN, USA, 6–9 May 2013.
35. Duan, Q.; Sheng, W.X.; Ma, Y.; Ma, K. Sparse Bayesian Learning Using Combined Kernels for Medium Term Load Forecasting. In Proceedings of the 2nd IET Renewable Power Generation Conference (RPG 2013), Beijing, China, 9–11 September 2013.
36. Giamarellos, N.; Zois, E.N.; Papadimitrakis, M.; Stogiannos, M.; Livanos, N.A.-I.; Alexandridis, A. Short-Term Electric Load Forecasting with Sparse Coding Methods. *IEEE Access* **2021**, *9*, 102847–102861. [\[CrossRef\]](#)
37. Azad, M.K.; Uddin, S.; Takruri, M. Support Vector Regression Based Electricity Peak Load Forecasting. In Proceedings of the 11th International Symposium on Mechatronics and its Applications, ISMA 2018, Sharjah, United Arab Emirates, 4–6 March 2018; pp. 1–5.
38. Che, J.; Wang, J. Short-Term Load Forecasting Using a Kernel-Based Support Vector Regression Combination Model. *Appl. Energy* **2014**, *132*, 602–609. [\[CrossRef\]](#)
39. Li, Y.; Che, J.; Yang, Y. Subsampled Support Vector Regression Ensemble for Short Term Electric Load Forecasting. *Energy* **2018**, *164*, 160–170. [\[CrossRef\]](#)
40. Ghelardoni, L.; Ghio, A.; Anguita, D. Energy Load Forecasting Using Empirical Mode Decomposition and Support Vector Regression. *IEEE Trans. Smart Grid* **2013**, *4*, 549–556. [\[CrossRef\]](#)
41. Hafeez, G.; Khan, I.; Jan, S.; Shah, I.A.; Khan, F.A.; Derhab, A. A Novel Hybrid Load Forecasting Framework with Intelligent Feature Engineering and Optimization Algorithm in Smart Grid. *Appl. Energy* **2021**, *299*, 117178. [\[CrossRef\]](#)
42. Korovesis, N.; Kandris, D.; Koulouras, G.; Alexandridis, A. Robot Motion Control via an Eeg-Based Brain–Computer Interface by Using Neural Networks and Alpha Brainwaves. *Electronics* **2019**, *8*, 1387. [\[CrossRef\]](#)
43. Akarslan, E.; Hocaoglu, F.O. Electricity Demand Forecasting of a Micro Grid Using ANN. In Proceedings of the 2018 9th International Renewable Energy Congress, IREC, Hammamet, Tunisia, 20–22 March 2018; pp. 1–5.
44. Pandey, A.K.; Sahay, K.B.; Chandra, D.; Tripathi, M.M. Day Ahead Load Forecast in ISO New England Market and Ontario Market Using a Novel ANN. *Int. J. Res. Emerg. Sci. Technol.* **2015**, *2*, 30–40.
45. Bala, A.; Yadav, N.K. Load Forecasting For Weekend Load Using ANN Technique in Deregulated Environment. *IOSR J. Electr. Electron. Eng.* **2014**, *9*, 1–6.

46. Dilhani, M.H.M.R.S.; Jeenanunta, C. Daily Electric Load Forecasting: Case of Thailand. In Proceedings of the 7th International Conference on Information Communication Technology for Embedded Systems 2016, IC-ICTES, Bangkok, Thailand, 20–22 March 2016; pp. 25–29.
47. Raza, M.Q.; Baharudin, Z.; Badar-Ul-Islam; Azman Zakariya, M.; Khir, M.H.M. Neural Network Based STLF Model to Study the Seasonal Impact of Weather and Exogenous Variables. *Res. J. Appl. Sci. Eng. Technol.* **2013**, *6*, 3729–3735. [\[CrossRef\]](#)
48. Sahay, K.B.; Tripathi, M.M. Day Ahead Hourly Load Forecast of PJM Electricity Market and Iso New England Market by Using Artificial Neural Network. In Proceedings of the ISGT 2014, Istanbul, Türkiye, 12–15 October 2014; pp. 1–5.
49. Roldán-Blay, C.; Escrivá-Escrivá, G.; Álvarez-Bel, C.; Roldán-Porta, C.; Rodríguez-García, J. Upgrade of an Artificial Neural Network Prediction Method for Electrical Consumption Forecasting Using an Hourly Temperature Curve Model. *Energy Build.* **2013**, *60*, 38–46. [\[CrossRef\]](#)
50. Rodrigues, F.; Cardeira, C.; Calado, J.M.F. The Daily and Hourly Energy Consumption and Load Forecasting Using Artificial Neural Network Method: A Case Study Using a Set of 93 Households in Portugal. *Energy Procedia* **2014**, *62*, 220–229. [\[CrossRef\]](#)
51. Sahay, K.B.; Sahu, S.; Singh, P. Short-Term Load Forecasting of Toronto Canada by Using Different ANN Algorithms. In Proceedings of the 2016 IEEE 6th International Conference on Power Systems (ICPS), New Delhi, India, 4–6 March 2016; pp. 1–6.
52. Alhמוד, L.; Abu Khurma, R.; Al-Zoubi, A.M.; Aljarah, I. A Real-Time Electrical Load Forecasting in Jordan Using an Enhanced Evolutionary Feedforward Neural Network. *Sensors* **2021**, *21*, 6240. [\[CrossRef\]](#)
53. Liao, Z.; Pan, H.; Huang, X.; Mo, R.; Fan, X.; Chen, H.; Liu, L.; Li, Y. Short-Term Load Forecasting with Dense Average Network. *Expert Syst. Appl.* **2021**, *186*, 115748. [\[CrossRef\]](#)
54. Ding, M.; Wang, L.; Bi, R. An ANN-Based Approach for Forecasting the Power Output of Photovoltaic System. *Procedia Environ. Sci.* **2011**, *11*, 1308–1315.
55. Salgado, R.M.; Ohishi, T.; Ballini, R. A Short-Term Bus Load Forecasting System. In Proceedings of the 2010 10th International Conference on Hybrid Intelligent Systems, Atlanta, GA, USA, 23–25 August 2010; pp. 55–60.
56. Hernandez, L.; Baladron, C.; Aguiar, J.; Carro, B.; Sanchez-Esguevillas, A.; Lloret, J.; Chinarro, D.; Gomez-Sanz, J.; Cook, D. A Multi-Agent System Architecture for Smart Grid Management and Forecasting of Energy Demand in Virtual Power Plants. *IEEE Commun. Mag.* **2013**, *51*, 106–113. [\[CrossRef\]](#)
57. Alamin, Y.I.; Álvarez, J.D.; del Mar Castilla, M.; Ruano, A. An Artificial Neural Network (ANN) Model to Predict the Electric Load Profile for an HVAC System. *IFAC PapersOnLine* **2018**, *51*, 26–31. [\[CrossRef\]](#)
58. González-Romera, E.; Jaramillo-Morán, M.Á.; Carmona-Fernández, D. Monthly Electric Energy Demand Forecasting Based on Trend Extraction. *IEEE Trans. Power Syst.* **2006**, *21*, 1946–1953. [\[CrossRef\]](#)
59. Johannesen, N.J.; Kolhe, M.; Goodwin, M. Relative Evaluation of Regression Tools for Urban Area Electrical Energy Demand Forecasting. *J. Clean. Prod.* **2019**, *218*, 555–564. [\[CrossRef\]](#)
60. Zhang, W.; Quan, H.; Srinivasan, D. Parallel and Reliable Probabilistic Load Forecasting via Quantile Regression Forest and Quantile Determination. *Energy* **2018**, *160*, 810–819. [\[CrossRef\]](#)
61. Lusi, P.; Khalilpour, K.R.; Andrew, L.; Liebman, A. Short-Term Residential Load Forecasting: Impact of Calendar Effects and Forecast Granularity. *Appl. Energy* **2017**, *205*, 654–669. [\[CrossRef\]](#)
62. Vantuch, T.; Vidal, A.G.; Ramallo-Gonzalez, A.P.; Skarmeta, A.F.; Misak, S. Machine Learning Based Electric Load Forecasting for Short and Long-Term Period. In Proceedings of the 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, 5–8 February 2018; pp. 511–516.
63. Bessec, M.; Fouquau, J. Short-Run Electricity Load Forecasting with Combinations of Stationary Wavelet Transforms. *Eur. J. Oper. Res.* **2018**, *264*, 149–164. [\[CrossRef\]](#)
64. Mughees, N.; Mohsin, S.A.; Mughees, A.; Mughees, A. Deep Sequence to Sequence Bi-LSTM Neural Networks for Day-Ahead Peak Load Forecasting. *Expert Syst. Appl.* **2021**, *175*, 114844. [\[CrossRef\]](#)
65. Adam, S.P.; Alexandropoulos, S.A.N.; Pardalos, P.M.; Vrahatis, M.N. No Free Lunch Theorem: A Review. *Approx. Optim.* **2019**, *145*, 57–82. [\[CrossRef\]](#)
66. Ahmia, O.; Farah, N. Multi-Model Approach for Electrical Load Forecasting. In Proceedings of the 2015 SAI Intelligent Systems Conference (IntelliSys), London, UK, 10–11 November 2015; pp. 87–92. [\[CrossRef\]](#)
67. Zeng, P.; Jin, M.; Elahe, M.F. Short-Term Power Load Forecasting Based on Cross Multi-Model and Second Decision Mechanism. *IEEE Access* **2020**, *8*, 184061–184072. [\[CrossRef\]](#)
68. 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. [\[CrossRef\]](#)
69. Li, S.; Zhong, Y.; Lin, J. AWS-DAIE: Incremental Ensemble Short-Term Electricity Load Forecasting Based on Sample Domain Adaptation. *Sustainability* **2022**, *14*, 14205. [\[CrossRef\]](#)
70. Yu, L.; Wang, S.; Lai, K.K. A Novel Nonlinear Ensemble Forecasting Model Incorporating GLAR and ANN for Foreign Exchange Rates. *Comput. Oper. Res.* **2005**, *32*, 2523–2541. [\[CrossRef\]](#)
71. De Mattos Neto, P.S.G.; De Oliveira, J.F.L.; Bassetto, P.; Siqueira, H.V.; Barbosa, L.; Pereira Alves, E.; Marinho, M.H.N.; Rissi, G.F.; Li, F.; Oliveira, J.F.L.; et al. Energy Consumption Forecasting for Smart Meters Using Extreme Learning Machine Ensemble. *Sensors* **2021**, *21*, 8096. [\[CrossRef\]](#)
72. Hu, Y.; Qu, B.; Wang, J.; Liang, J.; Wang, Y.; Yu, K.; Li, Y.; Qiao, K. Short-Term Load Forecasting Using Multimodal Evolutionary Algorithm and Random Vector Functional Link Network Based Ensemble Learning. *Appl. Energy* **2021**, *285*, 116415. [\[CrossRef\]](#)

73. Duan, Q.; Liu, J.; Zhao, D. Short Term Electric Load Forecasting Using an Automated System of Model Choice. *Int. J. Electr. Power Energy Syst.* **2017**, *91*, 92–100. [[CrossRef](#)]
74. Chen, Q.; Zhang, W.; Zhu, K.; Zhou, D.; Dai, H.; Wu, Q. A Novel Trilinear Deep Residual Network with Self-Adaptive Dropout Method for Short-Term Load Forecasting. *Expert Syst. Appl.* **2021**, *182*, 115272. [[CrossRef](#)]
75. Gao, R.; Du, L.; Suganthan, P.N.; Zhou, Q.; Yuen, K.F. Random Vector Functional Link Neural Network Based Ensemble Deep Learning for Short-Term Load Forecasting. *Expert Syst. Appl.* **2022**, *206*, 117784. [[CrossRef](#)]
76. Saviozzi, M.; Massucco, S.; Silvestro, F. Implementation of Advanced Functionalities for Distribution Management Systems: Load Forecasting and Modeling through Artificial Neural Networks Ensembles. *Electr. Power Syst. Res.* **2019**, *167*, 230–239.
77. Montgomery, D.C.; Peck, E.A.; Vining, G.G. *Introduction to Linear Regression Analysis*, 5th ed.; Balding, D.J., Cressie, N.A.C., Fitzmaurice, G.M., Eds.; John Wiley & Sons: Hoboken, NJ, USA, 2012; ISBN 978-0-470-54281-1.
78. Zhang, Z.; Xu, Y.; Yang, J.; Li, X.; Zhang, D. A Survey of Sparse Representation: Algorithms and Applications. *IEEE Access* **2015**, *3*, 490–530. [[CrossRef](#)]
79. Pati, Y.C.; Rezaifar, R.; Krishnaprasad, P.S. Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition. In Proceedings of the 27th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 1–3 November 1993.
80. Beck, A.; Teboulle, M. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM J. Imaging Sci.* **2009**, *2*, 183–202. [[CrossRef](#)]
81. Chen, S.S.; Donoho, D.L.; Saunders, M.A. Atomic Decomposition by Basis Pursuit. *SIAM Rev.* **2001**, *43*, 129–159. [[CrossRef](#)]
82. Mairal, J.; Bach, F.; Ponce, J.; Sapiro, G.; Zisserman, A. Supervised Dictionary Learning. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 7–10 December 2009.
83. Mairal, J.; Bach, F.; Ponce, J.; Sapiro, G. Online Dictionary Learning for Sparse Coding. In Proceedings of the ACM International Conference Proceeding Series, Athens, Greece, 29–31 October 2009.
84. Engan, K.; Aase, S.O.; Husoy, J.H. Method of Optimal Directions for Frame Design. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99, Phoenix, AZ, USA, 15–19 March 1999. [[CrossRef](#)]
85. Aharon, M.; Elad, M.; Bruckstein, A. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Trans. Signal Process.* **2006**, *54*, 4311–4322. [[CrossRef](#)]
86. Robbins, H.; Monro, S. A Stochastic Approximation Method. *Ann. Math. Stat.* **1951**, *22*, 400–407. [[CrossRef](#)]
87. Kyriakides, E.; Polycarpou, M. Short Term Electric Load Forecasting: A Tutorial. *Trends Neural Comput.* **2006**, *35*, 391–418. [[CrossRef](#)]
88. Hong, W.C. *Intelligent Energy Demand Forecasting*; Springer: London, UK, 2013; Volume 10.
89. Alexandridis, A.; Chondrodima, E.; Giannopoulos, N.; Sarimveis, H. A Fast and Efficient Method for Training Categorical Radial Basis Function Networks. *IEEE Trans. Neural Networks Learn. Syst.* **2017**, *28*, 2831–2836. [[CrossRef](#)]
90. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
91. Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. *Classification and Regression Trees*; John Wiley and Sons: Hoboken, NJ, USA, 2017; ISBN 9781351460491.
92. Tan, Z.; Zhang, J.; He, Y.; Zhang, Y.; Xiong, G.; Liu, Y. Short-Term Load Forecasting Based on Integration of SVR and Stacking. *IEEE Access* **2020**, *8*, 227719–227728. [[CrossRef](#)]
93. Zulfiqar, M.; Gamage, K.A.A.; Kamran, M.; Rasheed, M.B. Hyperparameter Optimization of Bayesian Neural Network Using Bayesian Optimization and Intelligent Feature Engineering for Load Forecasting. *Sensors* **2022**, *22*, 4446. [[CrossRef](#)]
94. Marquardt, D.W. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *J. Soc. Ind. Appl. Math.* **1963**, *11*, 431–441. [[CrossRef](#)]
95. Alexandridis, A.; Chondrodima, E.; Sarimveis, H. Radial Basis Function Network Training Using a Nonsymmetric Partition of the Input Space and Particle Swarm Optimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 219–230. [[CrossRef](#)]
96. Papadimitrakis, M.; Alexandridis, A. Active Vehicle Suspension Control Using Road Preview Model Predictive Control and Radial Basis Function Networks. *Appl. Soft Comput.* **2022**, *120*, 108646. [[CrossRef](#)]
97. Karamichailidou, D.; Alexandridis, A.; Anagnostopoulos, G.; Syriopoulos, G.; Sekkas, O. Modeling Biogas Production from Anaerobic Wastewater Treatment Plants Using Radial Basis Function Networks and Differential Evolution. *Comput. Chem. Eng.* **2022**, *157*, 107629. [[CrossRef](#)]
98. Chondrodima, E.; Georgiou, H.; Pelekis, N.; Theodoridis, Y. Particle Swarm Optimization and RBF Neural Networks for Public Transport Arrival Time Prediction Using GTFS Data. *Int. J. Inf. Manag. Data Insights* **2022**, *2*, 100086. [[CrossRef](#)]
99. Karamichailidou, D.; Kaloutsas, V.; Alexandridis, A. Wind Turbine Power Curve Modeling Using Radial Basis Function Neural Networks and Tabu Search. *Renew. Energy* **2021**, *163*, 2137–2152. [[CrossRef](#)]
100. Sun, L.; Zhou, K.; Zhang, X.; Yang, S. Outlier Data Treatment Methods Toward Smart Grid Applications. *IEEE Access* **2018**, *6*, 39849–39859. [[CrossRef](#)]
101. Martín, P.; Moreno, G.; Rodríguez, F.J.; Jiménez, J.A.; Fernández, I. A Hybrid Approach to Short-Term Load Forecasting Aimed at Bad Data Detection in Secondary Substation Monitoring Equipment. *Sensors* **2018**, *18*, 3947. [[CrossRef](#)]
102. Wang, Y.; Sun, S.; Chen, X.; Zeng, X.; Kong, Y.; Chen, J.; Guo, Y.; Wang, T. Short-Term Load Forecasting of Industrial Customers Based on SVM and XGBoost. *Int. J. Electr. Power Energy Syst.* **2021**, *129*, 106830. [[CrossRef](#)]

103. Wang, S.; Wang, X.; Wang, S.; Wang, D. Bi-Directional Long Short-Term Memory Method Based on Attention Mechanism and Rolling Update for Short-Term Load Forecasting. *Int. J. Electr. Power Energy Syst.* **2019**, *109*, 470–479. [[CrossRef](#)]
104. Javed, U.; Ijaz, K.; Jawad, M.; Khosa, I.; Ahmad Ansari, E.; Shabih Zaidi, K.; Nadeem Rafiq, M.; Shabbir, N. A Novel Short Receptive Field Based Dilated Causal Convolutional Network Integrated with Bidirectional LSTM for Short-Term Load Forecasting. *Expert Syst. Appl.* **2022**, *205*, 117689. [[CrossRef](#)]
105. Román-Portabales, A.; López-Nores, M.; Pazos-Arias, J.J. Systematic Review of Electricity Demand Forecast Using ANN-Based Machine Learning Algorithms. *Sensors* **2021**, *21*, 4544. [[CrossRef](#)]
106. Rashidi, M.M.; Alhuyi Nazari, M.; Mahariq, I.; Ali, N. Modeling and Sensitivity Analysis of Thermal Conductivity of Ethylene Glycol-Water Based Nanofluids with Alumina Nanoparticles. *Exp. Tech.* **2023**, *47*, 83–90. [[CrossRef](#)]
107. Zhou, B.; Meng, Y.; Huang, W.; Wang, H.; Deng, L.; Huang, S.; Wei, J. Multi-Energy Net Load Forecasting for Integrated Local Energy Systems with Heterogeneous Prosumers. *Int. J. Electr. Power Energy Syst.* **2021**, *126*, 106542. [[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.