

ORIGINAL RESEARCH

Open access

Fine-Grained Peak Load Prediction Leveraging Generalized Additive Models and Deep Neural Network Architectures

George Brown¹, Michael Taylor^{1*}, Sarah Wilson², Olivia Harris¹

Abstract

This study investigates the prediction of fine-grained electricity peak demand using coarser-resolution data. Specifically, it addresses whether sparse high-resolution monitoring can inform forecasts of future high-resolution peaks when detailed measurements are unavailable, a question of particular importance for network operators evaluating the replacement of costly monitoring systems with predictive models. We develop models capable of estimating half-hourly maxima and minima from one-minute interval electricity load data using inputs aggregated to 30-minute intervals. Our methodology combines generalized additive models (GAMs) and deep neural networks (DNNs), both widely applied in load forecasting. The analysis includes an in-depth assessment of input feature importance, focusing on load patterns, meteorological variables, and seasonal trends. The proposed approach achieved first place in a data competition organized by Western Power Distribution (WPD), a UK electricity distribution operator. Beyond the competition, we conduct an extended evaluation over an additional 11 months, confirming that our combined model significantly outperforms the benchmark in out-of-sample root mean squared error (RMSE), achieving a 57.4% reduction compared to the competition reference.

Keywords Electricity peak load, Generalized additive models, Artificial neural networks, Prediction, Combination, Weather effects

*Correspondence:

Michael Taylor

michael.taylor@auckland.ac.nz

¹ Department of Smart Grid Systems, Faculty of Engineering, University of Auckland, Auckland, New Zealand

² Department of Energy Optimization Analytics, Faculty of Engineering, University of Otago, Dunedin, New Zealand

Introduction

As energy systems become increasingly decentralized, accurate electricity demand forecasting is critical to prevent stress on distribution networks caused by sudden demand spikes. The anticipated growth of low-carbon technologies such as electric vehicles and heat pumps further heightens this challenge. While continuous high-resolution monitoring could help, it entails substantial installation and maintenance costs, motivating network operators to explore predictive alternatives.

This paper presents a high-performing approach for forecasting high-resolution peak demand by integrating generalized additive models (GAMs) and deep neural networks (DNNs). The models were developed for a data competition organized by Western Power Distribution (WPD), covering the Midlands, South West, and Wales. The competition task required forecasting minute-level maximum and minimum loads using only low-resolution (30-minute) electricity measurements combined with weather information. This framework evaluates whether limited high-resolution data can suffice to predict future peaks without continuous monitoring, offering a cost-effective alternative to constant high-resolution measurements [1, 2]. If results generalize across substations, only low-resolution inputs would be necessary for high-resolution feature estimation.

Peak load forecasts serve different purposes depending on scale. National-level predictions support climate planning and resource allocation [3, 4], whereas regional forecasts are more relevant to distribution operators, which is the focus of this study. Load management strategies, such as peak shaving, are also of interest [5, 6] but lie beyond the core scope of this paper. A brief review of relevant literature follows.

Previous studies address short-term load prediction at varying scales, including individual households [7, 8], office buildings [9], low-voltage feeders [10], cities [11], and nations [4]. Results consistently indicate that weather variables, such as temperature, humidity, wind speed, and atmospheric pressure, are important predictors of peak electricity demand [12–15], with wind effects analyzed in detail by Acaroğlu and García Márquez [16]. Selection of appropriate weather stations is discussed by Hong *et al.* [17], while Haben *et al.* [10] find limited temperature influence when deterministic factors, such as day of week, are included.

The range of methods used for peak load forecasting is broad, encompassing time-varying autoregressive models (ARTV) [18], recurrent and convolutional neural networks (RNN, CNN) [14], long short-term memory networks (LSTM) [15], gradient boosting [19], and multi-step forecasting approaches [20–22]. GAMs are widely applied to both short-term [23–25] and high-resolution smart-meter load forecasting [26]. Neural network-based methods are even more prevalent, including multilayer perceptrons (MLPs), CNNs, and RNNs, used for both peak [27, 28] and general load forecasting [29–36].

The contributions of this work are summarized as follows:

- Proposing a high-resolution load prediction framework using only low-resolution data.
- Conducting a detailed descriptive analysis of load and weather inputs to optimize predictive performance.
- Developing a hybrid model combining GAMs and DNNs that outperforms competing approaches.
- Performing a rigorous validation study to confirm model robustness.
- Investigating the explanatory power of features and analyzing the effects of hyperparameter tuning.

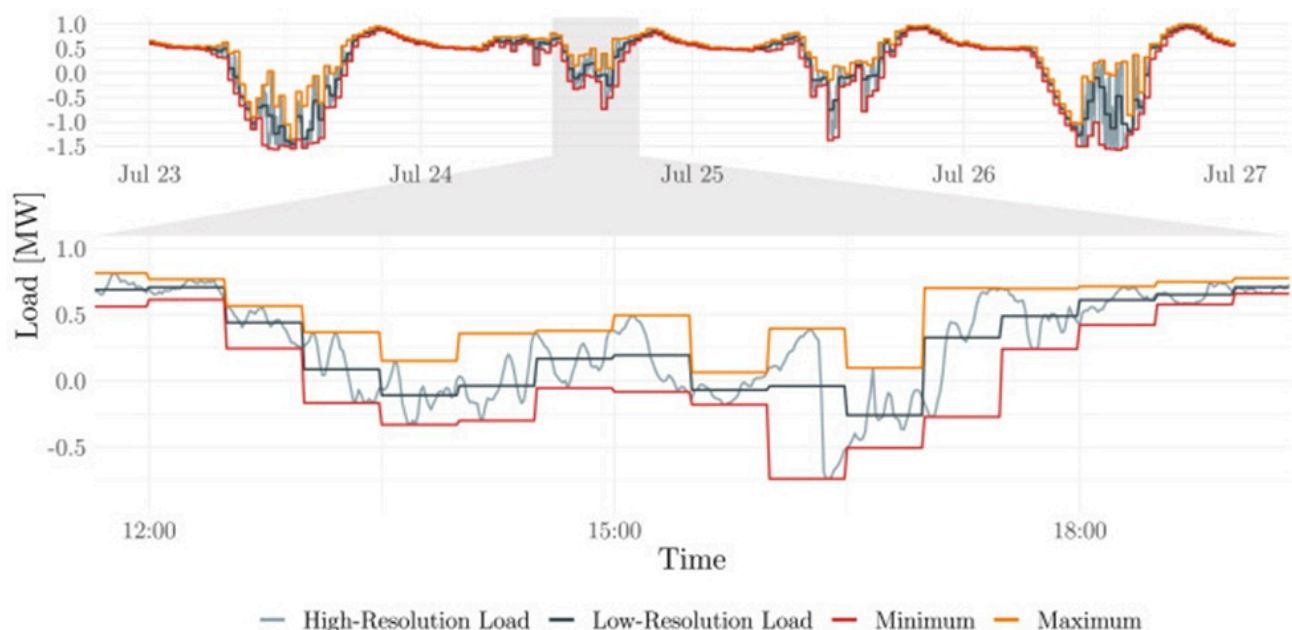


Figure 1. Example of the fine-grained peak load forecasting challenge using data from four days in July 2021. The task involves estimating the minimum and maximum load levels for each 30-minute window (highlighted in red and orange) from the detailed 1-minute-resolution load series (shown in gray).

These predictions must be generated using only the available 30-minute aggregated load readings (black line) (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article).

Problem Setup, Dataset, and Initial Data Exploration

The analysis relies on a comprehensive dataset from the high-resolution peak load estimation competition, covering the period from November 2019 through September 2021 — almost 2 years. WPD supplied this load information. Alongside the electricity demand series depicted in **Figure 1**, supplementary meteorological data from the MERRA-2 reanalysis product were made available to enhance forecast accuracy. The weather variables were collected from five different sites situated near the substation (**Figure 2**) and include temperature, solar radiation, meridional (north-south) wind, zonal (east-west) wind, atmospheric pressure, and specific humidity. In the modeling process, these six weather parameters were computed as simple averages across the five locations.

Originally, the competition required entrants to forecast the minimum and maximum load values for every half-hour interval during the entire month of September 2021. For a more thorough assessment of the method's reliability, the current study instead adopts a rolling-window validation framework that evaluates performance over 12 consecutive months, from October 2020 to September 2021. Because the winning entry in the competition was judged using a specialized skill metric derived from squared errors, the models are designed to predict the expected (mean) values of the half-hourly minima and maxima. This approach is necessary to maximize the skill score [37].

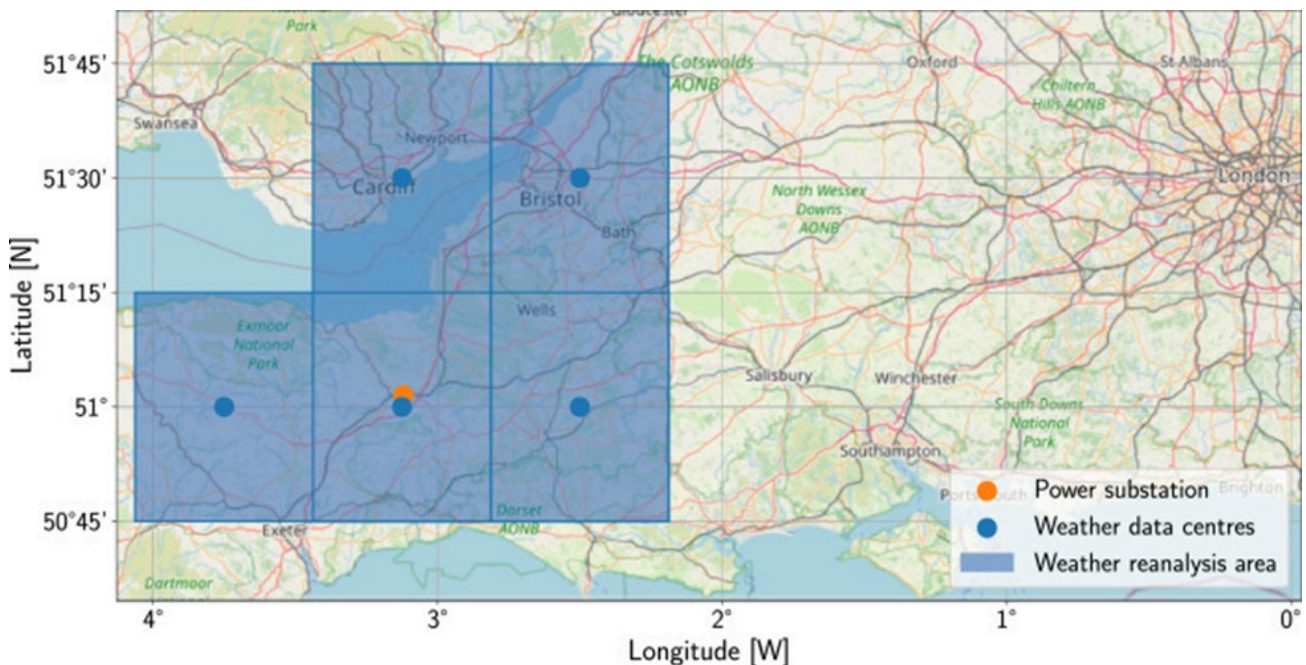


Figure 2. Position of the examined power substation together with the weather reanalysis data, situated in the south-west of England. The map was created with OpenStreetMap.

Let L_t represent the half-hourly load value, shown as the black line in **Figure 1**. Additionally, let L_{tmin} and L_{tmax} denote the minimum and maximum load values from the high-frequency data, which serve as our prediction targets. Since L_t is already known and satisfies $L_{tmin} < L_t < L_{tmax}$, it is logical to model the differences $\Delta_{tmin} = L_{tmin} - L_t$ and $\Delta_{tmax} =$

$L_{tmax} - L_t$. **Figure 3** displays these minimum and maximum load differences, Δt_{min} and Δt_{max} , across two chosen time periods. In this figure, we have also included the discrete second-order central difference (DSOCD).

$$L_t'' = L_{t-1} - 2L_t + L_{t+1} \quad (1)$$

Of the load L_t , along with its negative counterpart $-L_t''$. We notice that the absolute values of Δt_{min} and Δt_{max} tend to increase when L_t'' (or nearby values such as L_{t-1}'' or L_{t+1}'') exhibits a large absolute magnitude. Consequently, it is reasonable to treat L_t'' as a significant feature during model development.

A more detailed examination of the lower panel in **Figure 3** reveals notably high absolute values of L_t'' during winter, specifically at the very end and start of a day (namely at 23:30 and 00:00). This pattern is confined to a restricted interval within the winter months only. Nevertheless, we are unable to account for these particular values. Unlike other substantial readings, they do not correspond to large absolute values in Δt_{min} and Δt_{max} . For this reason, we chose to substitute these values to avoid any distortion from these anomalous observations in our forecasting process. To correct for this issue, we introduce a modified discrete second-order central difference, denoted as $\tilde{L}_t''$, in which the affected values are replaced via linear interpolation. This adjustment is formally expressed as:

$$\tilde{L}_t'' = \begin{cases} L_t'' & \text{if } t \bmod 48 = 1, \dots, 46 \\ \frac{1}{3}L_{t-2}'' + \frac{2}{3}L_{t+1}'' & \text{if } t \bmod 48 = 0 \\ \frac{2}{3}L_{t-2}'' + \frac{1}{3}L_{t+1}'' & \text{if } t \bmod 48 = 47. \end{cases} \quad (2)$$

We establish three deterministic components for subsequent initial-data examination to account for possible seasonal patterns in the dataset. These include daily, weekly, and annual variables, labeled as D_t , W_t , and A_t , respectively. D_t and W_t represent the cumulative hours elapsed within a day or a week, beginning from Monday. Similarly, A_t indicates the cumulative hours within a meteorological year consisting of 365.24 days, commencing at 00:00 on 1 January 2020.



Figure 3. Display of the minimum and maximum load deviations, defined as $\Delta t_{min} = L_{tmin} - L_t$ and $\Delta t_{max} = L_{tmax} - L_t$, alongside both the positive and negative versions of the discrete second-order central difference (DSOCD) of the load L_t . The plot covers four consecutive days during summer (23–26 July 2020, top panel) and four days in winter (5–8 December 2020, bottom panel).

Figure 4 presents a correlation matrix that combines all the main variables discussed so far. These consist of the load-related series Δt_{min} , Δt_{max} , L_t , the adjusted second-order difference $\tilde{L}_t''$, the original second-order difference L_t'' , the six weather-related time series, and the three deterministic seasonal components D_t , W_t , and A_t . Pearson's correlation coefficients are shown in the lower triangle of the matrix, while distance correlation values are presented in the upper triangle. Pearson's coefficient is the familiar linear correlation measure that ranges between -1 and 1 . Distance correlation, by contrast, is a non-linear measure bounded between 0 and 1 that can identify situations of statistical independence [38].

The matrix suggests that both L_t and the adjusted difference $\tilde{L}_t''$ are likely to play a key role in accounting for variations in Δt_{min} and Δt_{max} . In addition, the adjusted version $\tilde{L}_t''$ appears slightly more promising than the unadjusted L_t'' for predictive modeling. Among all weather variables, solar irradiance shows the strongest association with both load differences. This finding is understandable, as photovoltaic generation is expected to have a substantial impact on substation load. Even so, the true overall importance of the weather inputs is not yet clear, especially because they tend to overlap strongly with the deterministic seasonal factors. When looking at the seasonal components, their Pearson correlations with Δt_{min} and Δt_{max} are very weak, ranging from -0.04 to 0.05 . This suggests virtually no linear relationship between the seasonal variables and the target differences. However, the distance correlations for the daily and annual components are markedly higher. Given that the near-zero Pearson values already rule out meaningful linear relationships, these elevated distance correlations indicate that the links between the daily/annual seasonal inputs and $\Delta t_{min}/\Delta t_{max}$ are predominantly non-linear in character.

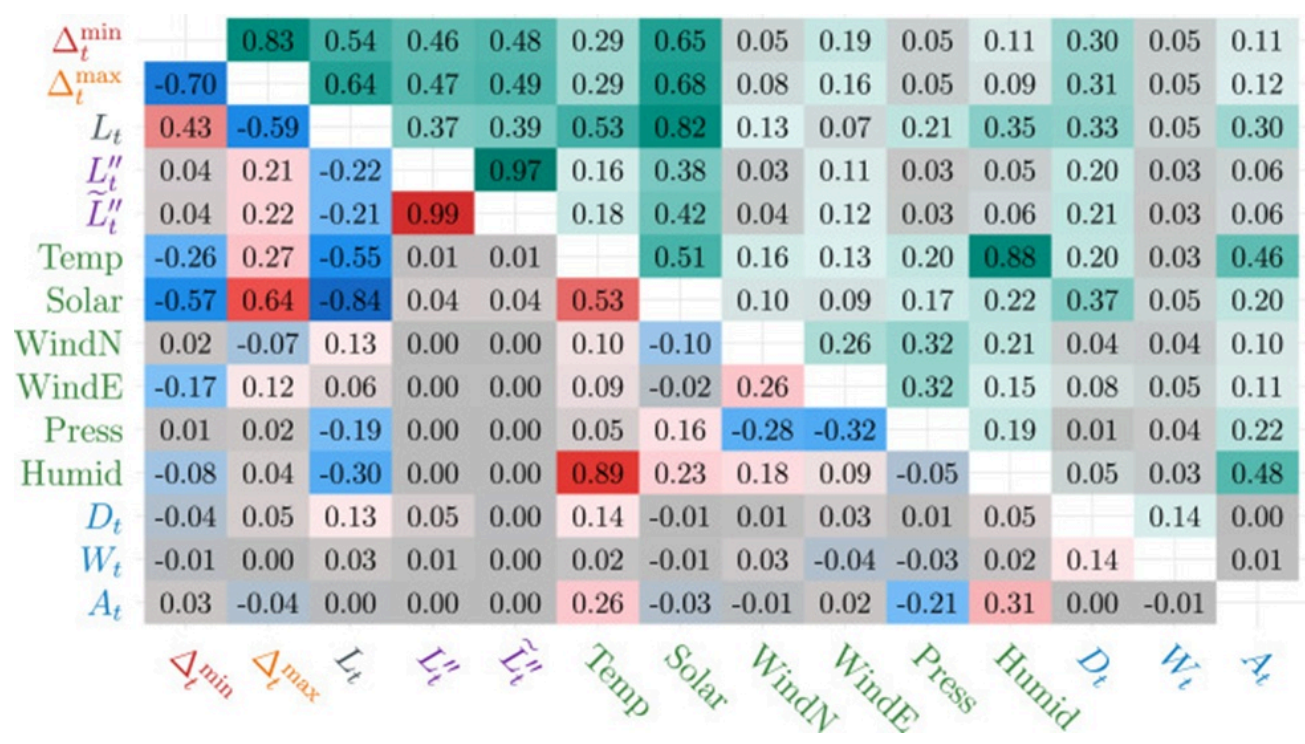


Figure 4. Correlation matrix displaying Pearson’s correlation coefficients in the lower triangle and distance correlation values in the upper triangle.

Models

To ensure a fair and consistent comparison, both the GAM and DNN models were constructed using an identical input matrix X. This matrix consists of n = 21 columns, which are detailed in Table 1. It is important to note once more that the challenge focused on accurate in-sample predictions rather than true out-of-sample forecasts of future values. As a result, unlike conventional forecasting tasks, the use of positive (non-negative) lags of the load and the discrete second-order central difference (DSOCD) of the load is permitted here to capture high-frequency peak load patterns better. Accordingly, such positive lags are incorporated into the input features.

Table 1. Table of all 3+9+6+3 = 21 inputs.

Variable type	Included feature	Number
Lagged load	Lt-1, Lt, Lt+1	3
Lagged DSOCD load	L~t-4'',...,L~t+4''	9
Weather inputs	Temp, Solar, WindN, WindE, Press, Humid	6
Seasonal inputs	Dt, Wt, At	3

GAM models

A generalized additive model (GAM) consists of M additive components. In the context of modeling our targets $\Delta_{\min}$ and $\Delta_{\max}$, the general form can be expressed as:

$$\Delta_t^m = \sum_{i=1}^M f_i(X_{t,1}, \dots, X_{t,N}) + \epsilon_t \quad (3)$$

Where $m \in \{\min, \max\}$.

Each additive term f_i can take various forms. While linear terms and smoothing splines are the classical choices for these components, more sophisticated elements such as gradient boosting machines or deep neural networks have also been employed in recent studies [39, 40].

For this particular application, we simplify the GAM structure defined in Eq. 3 by restricting it to linear terms and pairwise (2-way) interactions. Higher-order interactions (3-way or above) are not considered. Furthermore, we adopt the conventional GAM approach based on splines—specifically, cubic B-splines. The full GAM that incorporates all possible 2-way interactions is specified as

$$\Delta_t^m = \sum_{i=1}^N b_{k_0}(X_{t,i}) + \sum_{i=1}^N \sum_{j=1, j>i}^N b_{k_1, k_2}(X_{t,i}, X_{t,j}) + \epsilon_t \quad (4)$$

Where b_{k_0} represents univariate splines with k_0 knots, and b_{k_1, k_2} denotes bivariate tensor product interaction splines with (k_1, k_2) knots that capture only the joint interaction effects. We set $k_0 = 27$ for the univariate terms and $k_1 = k_2 = 9$ for the bivariate terms. This configuration results in 27 parameters per linear term and 81 per bivariate interaction.

Parameter estimation is carried out using the `bam` function from the R package `mgcv` [41], which is well-suited for efficient computation with large datasets. To further reduce computational burden, the input data can be discretized before applying the fast restricted maximum likelihood (fREML) method. The fREML procedure assumes a Gaussian distribution with constant variance, effectively corresponding to ordinary least squares estimation. Model selection is guided by minimizing the Bayesian information criterion (BIC), accounting for the effective degrees of freedom contributed by the smoothing components. For additional technical and theoretical background on spline-based GAMs, refer to Wood [41].

In addition to the full model (4), we define a reduced GAM variant, GAM.red, that retains only a limited set of 2-way interactions. Specifically, this version includes interactions exclusively involving the key load variable L_t , the adjusted second derivative $\tilde{L}_t''$, and the daily seasonal component Solar_t . The reduced model is given by

$$\begin{aligned} \Delta_t^m = & \sum_{i=1}^N b_{k_0}(X_{t,i}) + \sum_{i=2}^N b_{k_1, k_2}(L_t, X_{t,i}) \\ & + \sum_{i=3}^N b_{k_1, k_2}(\tilde{L}_t'', X_{t,i}) + \sum_{i=4}^N b_{k_1, k_2}(\text{Solar}_t, X_{t,i}) + \epsilon_t \end{aligned} \quad (5)$$

Using the same knot settings: $k_0 = 27$ and $k_1 = k_2 = 9$. For notational simplicity in (5), we assume that the first three columns of the input matrix X correspond to L_t , $\tilde{L}_t''$, and Solar_t , respectively. Estimating the parameters of GAM.red requires considerably less computation time than the full model (4). On our eight-core system, the complete training and prediction process for either the minimum or maximum peak model drops from approximately 3 hours to roughly 2 minutes.

We also tested the main GAM variants (GAM.full and GAM.red) after replacing the north–south and east–west wind components with absolute wind speed.³ The resulting performance differences were negligible (below 0.05% in RMSE),

so we retained the original wind vector components and do not report the absolute wind speed results for conciseness. In any case, the strong non-linear modeling capacity of the DNNs enables them to learn such transformations internally; therefore, we used only the original directional wind components throughout.

To highlight the importance and non-linear influence of the three central variables, L_t , $\tilde{L}_t''$, and $Solar_t$ (beyond what was observed in the exploratory analysis), we additionally examine a simple GAM model, GAM.simple, defined as

$$\Delta_t^m = b_{k_0}(L_t) + b_{k_0}(\tilde{L}_t'') + b_{k_0}(Solar_t) + \varepsilon_t \quad (6)$$

With $k_0 = 27$. The BIC-optimized fit of model (6) is visualized in **Figure 5**. The plots clearly reveal strong non-linear relationships, particularly for the load L_t and the adjusted second derivative $\tilde{L}_t''$. Notably, the spline fit for $\tilde{L}_t''$ near zero resembles an absolute-value shape, which aligns with the pattern already evident in **Figure 3**.

Finally, we evaluate the contribution of the weather variables by fitting the full GAM (4) while excluding all meteorological inputs entirely. This variant is denoted as GAM.noWeather.

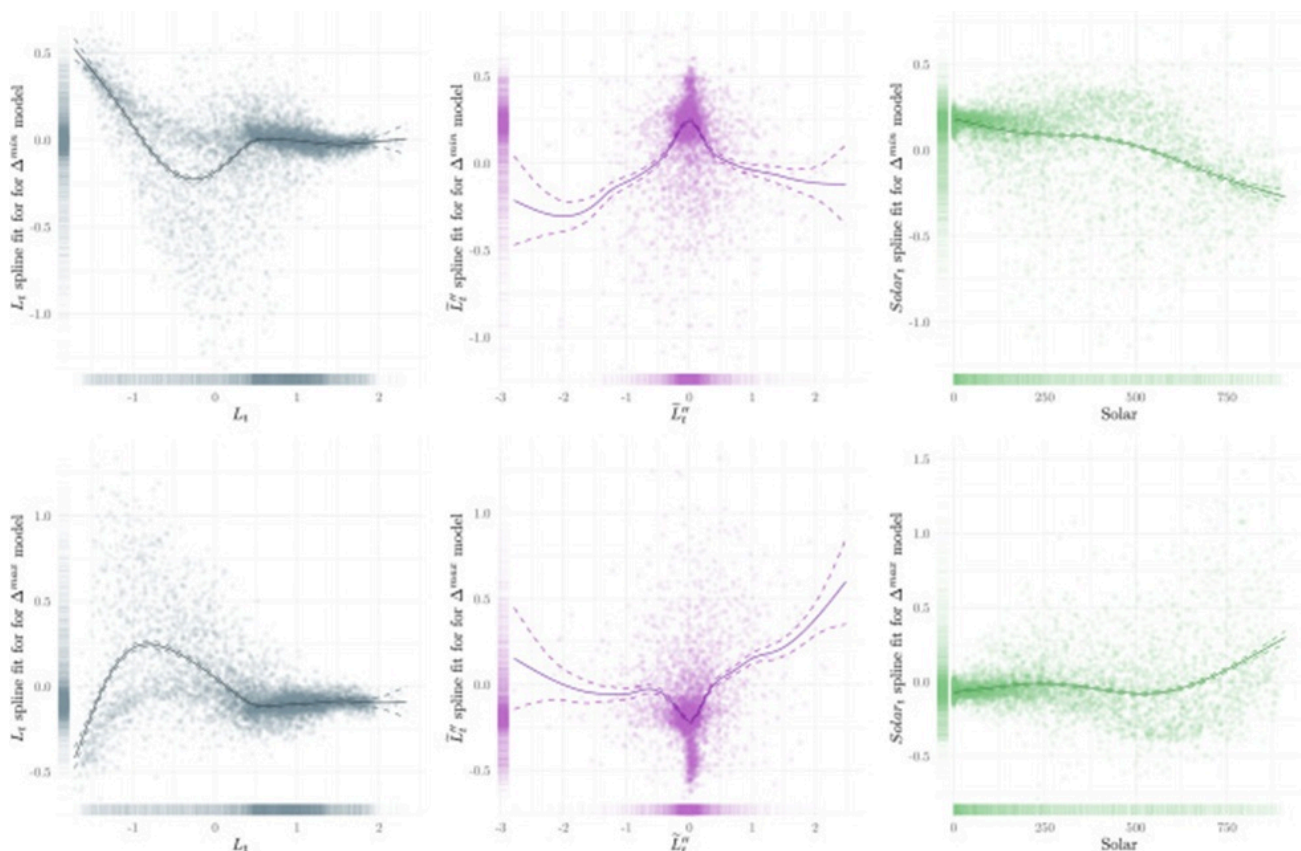


Figure 5. Estimated cubic B-splines obtained from the basic GAM model (6) for the minimum load deviation Δ_{tmin} (top row) and the maximum load deviation Δ_{tmax} (bottom row), displayed for the variables L_t (left), $\tilde{L}_t''$ (center), and $Solar_t$ (right).

Deep neural network model

The deep neural network follows a standard multilayer perceptron (MLP) design, depicted in **Figure 6**. In mathematical terms, the network for layers $i = 1$ to l is expressed as **Eq. 7**.

$$H_{t,i} = a_i H_{t,i-1} W_i + b_i \quad (7)$$

Here, $H_{t,i}$ denotes the activation output of the i -th hidden layer. In contrast, W_i , b_i , and a_i denote the weight matrix, bias term, and activation function associated with that layer. The input layer is defined as $H_{t,0} = X_t$, which consists of the full feature vector ($X_{t,1}, \dots, X_{t,N}$). The final output layer produces $H_{t,l+1} = \Delta_t$, which contains the two prediction targets, Δ_{tmin} and Δ_{tmax} . The architecture includes either two or three hidden layers ($l = 2$ or $l = 3$) and simultaneously generates predictions for both the minimum and maximum load differences.

Overfitting is mitigated through a combination of techniques: selective feature inclusion, dropout in the hidden layers, and L1 regularization of the network weights. These regularization strategies, along with several other modeling choices, are optimized jointly. The tunable parameters include the total number of hidden layers, the type of activation function used, the size of each hidden layer (i.e., the number of neurons), and the learning rate. All hyperparameters to be optimized are summarized in the list below.

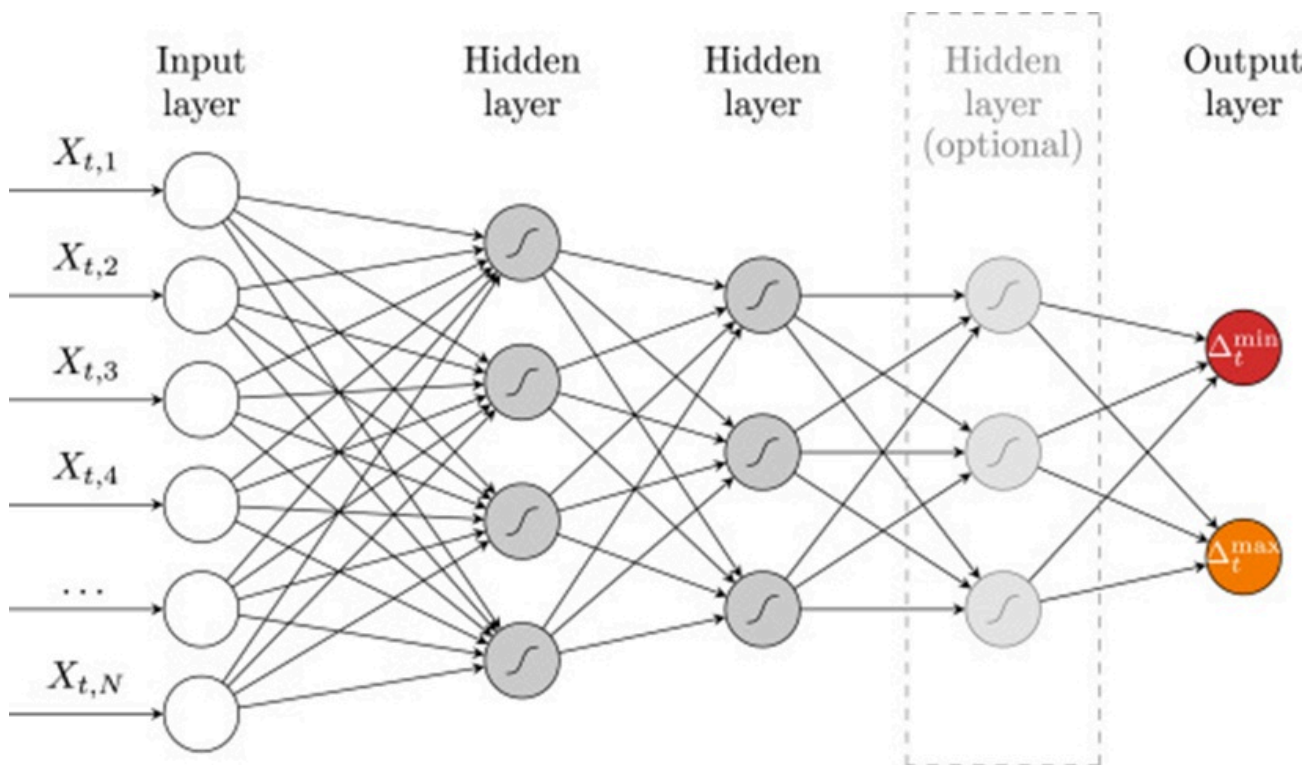


Figure 6. Representative architecture of the multilayer perceptron (MLP) used in the study.

The following hyperparameters are optimized during the tuning process:

- Selection of input features according to the 21 variables presented in [Table 1](#) (21 hyperparameters).
- Choice between 2 or 3 hidden layers (1 hyperparameter).
- Application of a dropout layer immediately after the input layer, including the decision whether to include it and, if so, the dropout probability sampled uniformly from the (0,1) range (up to 2 hyperparameters).
- Type of activation function applied in the hidden layers, chosen from elu, relu, sigmoid, softmax, softplus, and tanh (1 hyperparameter for each layer).

- Size of each hidden layer, i.e., the number of neurons, sampled on an exponential scale within the [4, 128] interval (1 hyperparameter per layer).
- Use of L1 regularization on the hidden layers and associated weights, including the decision to apply it and, if used, the strength parameter drawn on an exponential scale from the interval $[10^{-5}, 10]$ (up to 4 hyperparameters per layer).
- Learning rate of the Adam optimizer, sampled on an exponential scale from the interval $[10^{-5}, 10^{-1}]$ (1 hyperparameter).

Altogether, these choices yield up to 43 tunable hyperparameters. Given the vast number of possible combinations, an exhaustive search is impractical. Instead, hyperparameter optimization is performed using the Optuna library [42], which employs Bayesian optimization to identify promising configurations efficiently. A total of 1000 iterations is conducted. Within each iteration, a compact three-step expanding-window evaluation is performed. First, the model is fitted to the in-sample data excluding the final three months, and then it generates predictions for the next month. In the second step, the window advances by one month, incorporating the previously predicted period, and the fitting and prediction cycles are repeated. The third step mirrors the second. This procedure produces predictions for the latest three months of in-sample data under the tested hyperparameter set. Although restricting the validation to only the most recent month would more closely resemble the actual forecasting setup, using three months enhances robustness. It helps prevent excessive overfitting of the network. After each iteration, the mean squared error (MSE) of these predictions is computed, and Optuna selects the hyperparameter combination that minimizes it. The complete tuning routine is executed once per month, immediately before constructing the model for the following month, resulting in 12 independent tuning runs over the year.

Implementation and training of the network rely on the TensorFlow [43] and Keras [44] libraries. Beyond the optimized hyperparameters, several settings are fixed in advance: the validation split is set to 25% of the available data, the maximum number of training epochs is 1500, and the batch size is set to $7 \times 48 = 336$. The Adam optimizer is used, mean squared error serves as the loss function, and an early stopping mechanism with a patience of 50 epochs is applied. All input features undergo standardization before being fed into the model.

Owing to the inherent randomness in training deep neural networks, outcomes from identical architectures can vary across different runs. To mitigate this instability, each final prediction is computed as the average of an ensemble of 10 separate training runs for each of the five top-performing hyperparameter sets found during tuning. This approach produces an ensemble of 50 predictions, from which the mean is taken as the definitive output. This ensemble averaging method has been successfully used in both electricity price forecasting [45] and load forecasting [46].

Empirical Study and Results

Parameter significance in the GAM models

Table 2 reports the significance statistics for the reduced GAM model GAM.red defined in Eq. 5. For conciseness, the table includes only the main effects and pairwise interactions whose cumulative effective degrees of freedom (EDF) surpass 3 for individual terms and 6 for interactions. It also provides the F-statistic from an F-test that assesses the overall significance of each term relative to the hypothesis of no contribution. Corresponding p-values are not shown, as all listed terms exhibit extremely low values. The combination of large EDF counts and high F-statistics underscores the substantial influence of these particular model components.

Table 2. Effective degrees of freedom and F-statistic of selected model terms of the model GAM.red.

Minimum	Maximum

Single terms	EDF	F	Single terms	EDF	F
Lt	9.5	8.9	Lt	10.8	17.6
$L\sim t-1''$	8.8	4.4	$L\sim t-2''$	6.7	4.9
$L\sim t+1''$	8.4	4.8	Lt-1	6.3	6.9
$L\sim t''$	8.2	4.6	Lt+1	5.4	5.0
Humid	6.1	3.7	Dt	4.8	3.1
WindE	5.1	8.9	Temp	4.1	11.5
At	4.7	4.9	Solar	4.0	5.5
WindN	4.2	8.5	WindE	3.8	5.8
Temp	3.5	3.6	$L\sim t+4''$	3.2	2.1
Lt-1	3.3	1.3	$L\sim t+1''$	3.1	2.2
Interactions	EDF	F	Interactions	EDF	F
Lt, At	26.7	7.8	Lt, At	37.8	28.1
Lt, Solar	20.7	11.2	Lt, Dt	25.1	20.6
Lt, Dt	18.3	8.2	Solar, At	15.4	6.4
Lt, Lt+1	17.1	4.0	$L\sim t''$, $L\sim t-1''$	14.9	6.5
$L\sim t''$, $L\sim t-1''$	13.5	2.5	Solar, Dt	11.0	8.5
Lt, Temp	13.3	3.1	$L\sim t''$, $L\sim t+1''$	11.0	2.4
Lt, Lt-1	13.0	4.9	$L\sim t''$, $L\sim t+2''$	10.7	2.8
$L\sim t''$, $L\sim t+1''$	12.1	2.3	Lt, Solar	10.0	2.9
Lt, Wt	11.7	3.0	$L\sim t''$, Temp	9.5	3.4
$L\sim t''$, $L\sim t+2''$	11.4	2.9	Lt, WindE	9.3	2.5
Lt, Humid	10.1	2.6	Lt, $L\sim t-4''$	9.1	2.1
$L\sim t''$, $L\sim t-3''$	9.9	3.1	$L\sim t''$, $L\sim t-3''$	9.0	2.2
Lt, WindN	9.9	4.4	Solar, WindE	9.0	2.5
$L\sim t''$, Humid	9.8	1.8	Lt, Wt	8.9	1.9
$L\sim t''$, $L\sim t-2''$	9.3	2.1	Lt, $L\sim t-3''$	8.7	1.5
Lt+1, Solar	8.3	1.3	Lt, Lt+1	8.4	1.3
Solar, Dt	8.2	2.0	Lt, $L\sim t+3''$	7.4	1.3
Lt, WindE	7.6	2.5	Lt, $L\sim t+1''$	6.8	1.5
Solar, WindE	6.9	1.6	Lt, $L\sim t-1''$	6.5	1.4
$L\sim t''$, WindE	6.9	1.5	Lt, $L\sim t+4''$	6.1	1.0
Lt, $L\sim t+1''$	6.2	4.2			

The results indicate that L_t is the dominant input variable, both as a standalone term and in interaction effects. This pattern appears consistently in the models for both the minimum and maximum load deviations. In addition, several tensor-product interaction terms that capture non-linear relationships exhibit notably high effective degrees of freedom (EDF). Although the minimum and maximum models show several differences overall, the (lagged and adjusted) second derivatives of the load, denoted $\tilde{L}''_{t+k}$, attain high EDF values in both cases. A comparable observation applies to the meteorological variables and their interactions with other inputs. Among the weather-related features, solar irradiance emerges as the clearest contributor, while temperature, wind, and humidity also demonstrate meaningful influence. These meteorological impacts are consistent with findings reported in existing load forecasting studies [10, 12, 15].

DNN hyperparameter tuning outcomes

Hyperparameter tuning for the deep neural network was performed 12 times, with the five best configurations from each run retained, yielding a total of 60 hyperparameter sets. Figure 7 illustrates how frequently each input feature was selected across these optimized sets.

Figure 8 shows the activation functions in the best 60 DNN hyperparameter sets. The most commonly chosen features are L_t , $\tilde{L}''_{t+1}$, and Solar. This ranking closely mirrors the importance patterns observed in the reduced GAM model (Table 2). By contrast, the least frequently selected variables are $\tilde{L}''_{t-4}$ and WindN. Regarding network depth, three hidden layers were preferred in 60% of configurations, while two were used in the remaining 40%. On average, the first hidden layer contained 66 neurons, whereas the second and third layers (when present) each contained 38 neurons. Figure 9 explores the relationship between neuron counts across layers. Notably, networks with only two hidden layers generally use fewer neurons overall than those with three layers.

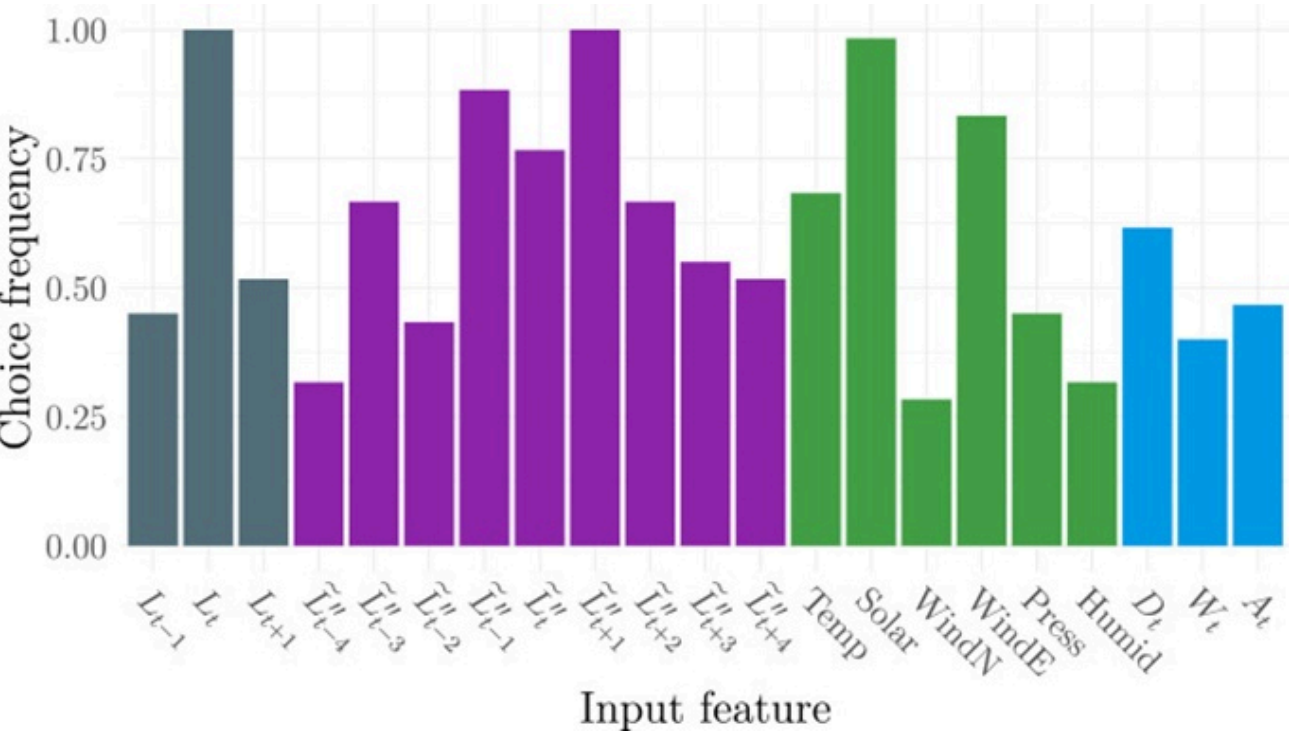


Figure 7. Input choice frequency in the best 60 DNN hyperparameter sets.

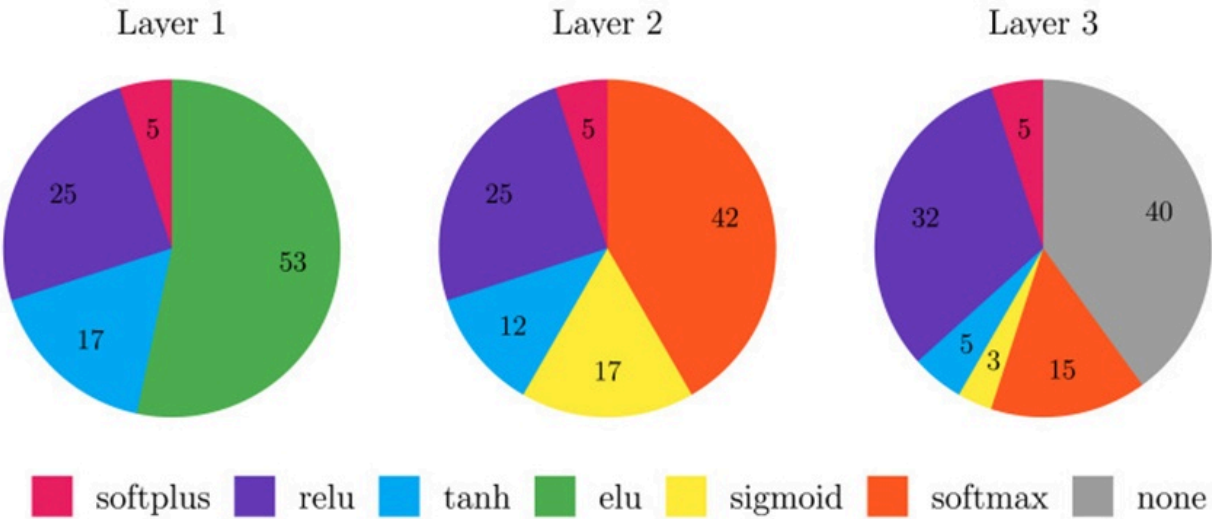


Figure 8. Activation functions in the best 60 DNN hyperparameter sets.

J. Berrisch et al.

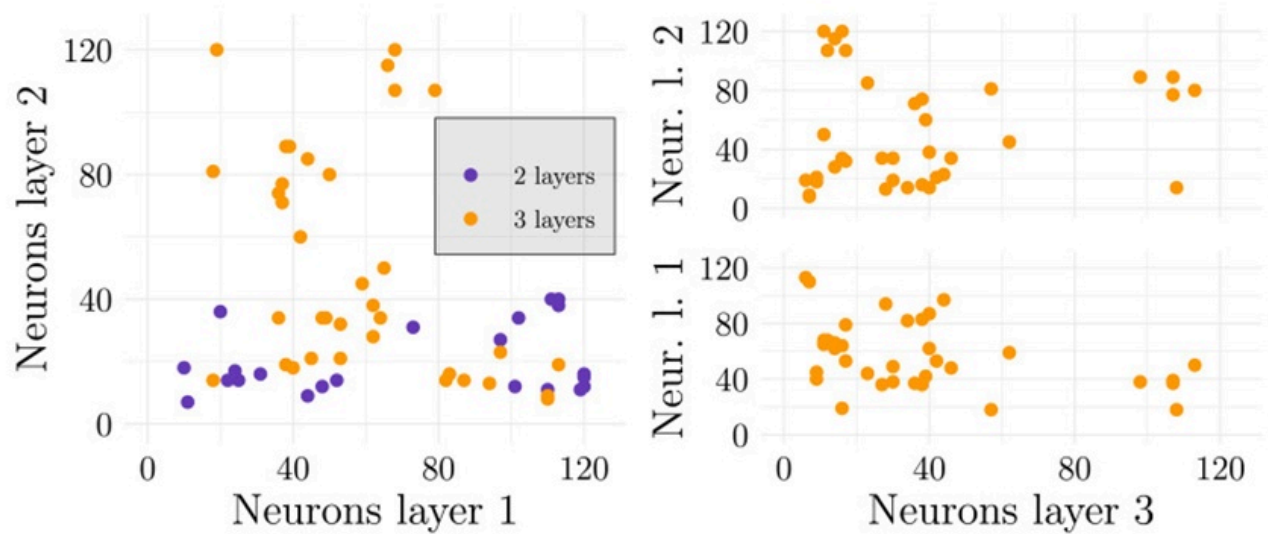


Figure 9. Neurons per layer in the best 60 DNN hyperparameter sets.

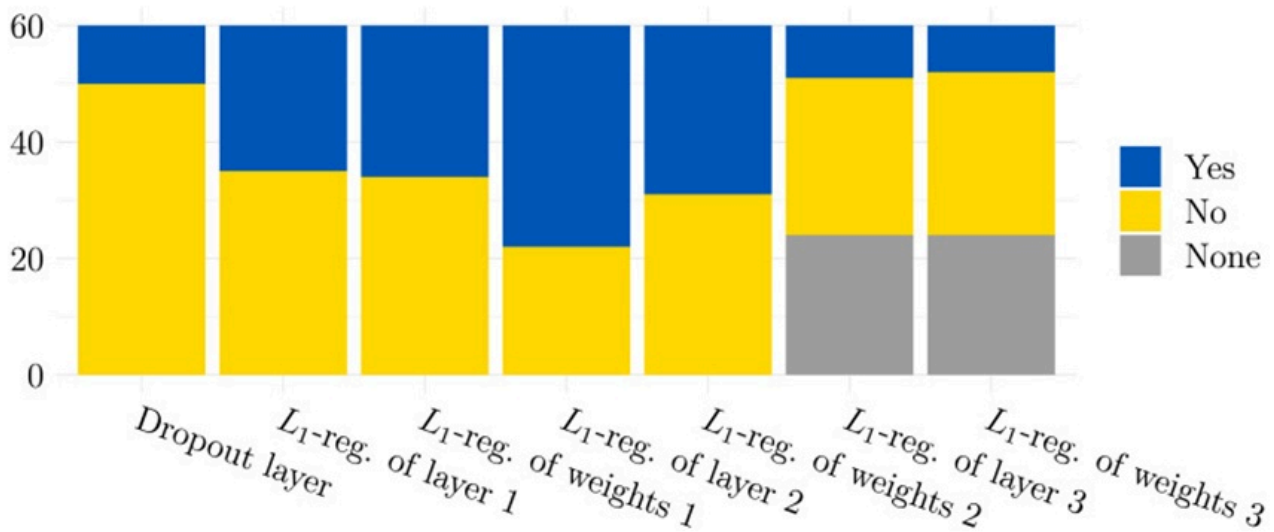


Figure 10. Regularization frequency in the best 60 DNN hyperparameter sets.

Figure 8 displays the selection frequency of different activation functions across the hidden layers. The elu activation was selected for approximately half of the first hidden layers, yet it was never chosen for the second or third layers. In contrast, the relu function appears evenly across the first two layers and becomes the dominant choice in the third layer. For the second hidden layer, softmax emerges as the most frequently selected activation function.

The extent to which dropout and L1 regularization were applied in the optimized networks is shown in **Figure 10**. In the majority of cases, the best-performing configurations did not incorporate either form of regularization. The only exception is L1 regularization on the second hidden layer, which was included in more than half of the tuned networks.

Study design

The performance of the proposed models is assessed through a comprehensive empirical evaluation. The main competing approaches are the complete GAM model GAM.full (4), the reduced GAM model GAM.red (5), and the deep neural network model DNN (7). Since combining different sophisticated models can often enhance overall accuracy, we also examine a simple uniform average of the predictions from GAM.full, GAM.red, and DNN. This naive ensemble assigns equal weights to the three individual models.

For benchmarking purposes, we further include the basic GAM model GAM.simple (6), the full GAM specification without any meteorological inputs GAM.noWeather (4), and a straightforward naive benchmark. The naive benchmark is defined by setting the predicted maximum load deviation to $\tilde{L}_t^{\max} = L_t$ and the minimum to $\tilde{L}_t^{\min} = L_t$.

As previously noted, model performance is evaluated using a rolling-window approach over the 12 months from October 2020 to September 2021. Predictive accuracy is quantified via the root mean squared error (RMSE) of the generated predictions. We compute both an overall joint RMSE and separate RMSE values for the minimum and maximum deviations:

$$\text{RMSE} = \sqrt{\sum_{t \in T} (\tilde{L}_t^{\min} - L_t^{\min})^2 + (\tilde{L}_t^{\max} - L_t^{\max})^2} \quad (8)$$

$$\text{RMSE}_m = \sqrt{\sum_{t \in T} (\tilde{L}_t^m - L_t^m)^2} \quad (9)$$

Where $m \in \{\min, \max\}$ and T denotes the set of time indices corresponding to the evaluated month.

The skill score employed in the original competition is based on the relative improvement of each model's RMSE compared with the naive benchmark, calculated as: [Eq. 10](#)

$$\text{Score} = \text{RMSE}(\text{Model}) / \text{RMSE}(\text{naive}). \tag{10}$$

Table 3. RMSE values (in MW) for the peak load prediction task in the evaluation study, including average performance across the period (final column). Relative improvements over the Naive benchmark (in %) are shown in brackets beneath each entry.

Model	20/10	20/11	20/12	21/1	21/2	21/3	21/4	21/5	21/6	21/7	21/8	21/9
GAM.full	0.1239 (56.3)	0.0703 (58.5)	0.0497 (56.6)	0.0532 (57.6)	0.0988 (55.4)	0.1202 (57.8)	0.1447 (57.2)	0.1746 (56.7)	0.1368 (56.3)	0.1537 (53.9)	0.1383 (60.2)	0.1163 (56.9)
GAM.red	0.1241 (56.2)	0.0700 (58.7)	0.0496 (56.6)	0.0536 (57.3)	0.0994 (55.2)	0.1201 (57.8)	0.1467 (56.6)	0.1750 (56.6)	0.1385 (55.8)	0.1554 (53.4)	0.1374 (60.5)	0.1162 (56.9)
DNN	0.1230 (56.6)	0.0704 (58.4)	0.0507 (55.7)	0.0545 (56.6)	0.1027 (53.7)	0.1251 (56.1)	0.1514 (55.2)	0.1653 (59.0)	0.1474 (52.9)	0.1538 (53.9)	0.1422 (59.1)	0.1225 (54.6)
Combination	0.1221 (56.9)	0.0689 (59.3)	0.0491 (57.1)	0.0527 (58.0)	0.0979 (55.8)	0.1193 (58.1)	0.1443 (57.3)	0.1684 (58.2)	0.1365 (56.4)	0.1512 (54.6)	0.1371 (60.6)	0.1164 (56.9)
GAM.noWeather	0.1301 (54.1)	0.0707 (58.2)	0.0508 (55.6)	0.0533 (57.5)	0.1058 (52.3)	0.1253 (56.0)	0.1471 (56.5)	0.1822 (54.8)	0.1401 (55.3)	0.1584 (52.5)	0.1424 (59.1)	0.1249 (53.7)
GAM.simple	0.1531 (46.0)	0.0940 (44.5)	0.0584 (49.0)	0.0672 (46.5)	0.1267 (42.9)	0.1500 (47.3)	0.1797 (46.8)	0.2093 (48.1)	0.1733 (44.7)	0.1873 (43.8)	0.1671 (52.0)	0.1413 (47.6)
Naive	0.2833	0.1693	0.1144	0.1255	0.2217	0.2817	0.3380	0.4029	0.3131	0.3334	0.3478	0.2699

Evaluation results

The outcomes presented in [Table 3](#) highlight the strong forecasting capability of the more advanced approaches. Both the full GAM model (GAM.full) and its reduced counterpart (GAM.red) achieve very similar levels of accuracy, each reducing the naive benchmark's RMSE by roughly 57%. The deep neural network achieves a slightly lower gain of 56%. Despite this, the DNN produces better predictions than the GAM variants in certain months, notably October 2020 and May 2021. Thanks to this complementary behavior, the simple uniform ensemble of the three models delivers stable performance. It edges out the individual models with an overall improvement of approximately 57.5% over the naive benchmark.

Special attention should be given to September 2021, the official test month of the WPD challenge. In that period, the GAM models stood out, achieving a skill score of 43.1% — equivalent to a 56.9% reduction in RMSE relative to the naive model. This matched the score obtained by the more robust Combination model. For the competition submission, a modestly adjusted version of the ensemble was entered (including, for example, some manually selected three-way interactions in the GAM component). It attained a final skill score of 42.6%, corresponding to a 57.4% improvement over the naive benchmark. The teams finishing second and third (WOJJ and code_green) recorded scores of 43.6% and 43.7%. This means the proposed Combination model surpassed the other competitors by more than one percentage point in RMSE terms.

Looking at the GAM.noWeather results in [Table 3](#), it becomes clear that adding the weather variables lifts the skill score by an average of 1.6 percentage points. While this increment may seem small, it carries real weight compared to the roughly 10 percentage-point gap separating the simple GAM model from the full GAM specification.

[Table 4](#) breaks down the performance into separate errors for the minimum and maximum peak loads. Here, the picture differs depending on the target. When forecasting the minimum load deviation, the DNN clearly trails behind the GAM models and even falls short of the no-weather GAM variant. As a result, the Combination model provides only a modest extra gain beyond what the strong GAM.full and GAM.red models already achieve. For the maximum load deviation, however, the DNN performs comparably to the two GAM models. This balance allows the ensemble to deliver a more noticeable boost in accuracy through aggregation.

Table 4. RMSEmin and RMSEmax values (in MW) for minimum and maximum peak load predictions in the evaluation study, with the overall average shown in the last column.

	20/10	20/11	20/12	21/1	21/2	21/3	21/4	21/5	21/6	21/7	21/8	21/9	Avg
Min													
GAM.full	.1551	.0831	.0544	.0593	.1136	.1281	.1429	.1824	.1389	.1658	.1528	.1169	.1244
GAM.red	.1529	.0826	.0543	.0598	.1151	.1278	.1452	.1801	.1416	.1673	.1513	.1188	.1247
DNN	.1513	.0838	.0565	.0630	.1176	.1373	.1518	.1746	.1527	.1684	.1579	.1297	.1287
Combination	.1517	.0818	.0541	.0596	.1132	.1285	.1433	.1761	.1404	.1644	.1520	.1197	.1237
GAM.noWeather	.1635	.0827	.0545	.0592	.1181	.1306	.1463	.1869	.1434	.1694	.1571	.1243	.1280
GAM.simple	.1889	.1105	.0623	.0758	.1413	.1624	.1768	.2122	.1817	.1963	.1842	.1443	.1531
Naive	.3256	.1890	.1196	.1305	.2375	.3010	.3330	.4131	.3269	.3468	.3691	.2723	.2804
Max													
GAM.full	.0816	.0545	.0444	.0462	.0813	.1116	.1464	.1663	.1346	.1406	.1220	.1156	.1038
GAM.red	.0860	.0544	.0444	.0466	.0805	.1120	.1481	.1698	.1353	.1425	.1220	.1135	.1046
DNN	.0859	.0538	.0440	.0445	.0853	.1115	.1511	.1555	.1418	.1377	.1247	.1149	.1042
Combination	.0824	.0529	.0434	.0446	.0797	.1094	.1453	.1603	.1324	.1366	.1203	.1129	.1017
GAM.noWeather	.0842	.0561	.0469	.0466	.0918	.1197	.1478	.1774	.1367	.1466	.1259	.1255	.1088
GAM.simple	.1057	.0740	.0541	.0574	.1101	.1365	.1825	.2065	.1645	.1780	.1481	.1381	.1296
Naive	.2335	.1469	.1090	.1203	.2047	.2674	.3429	.3925	.2986	.3193	.3522	.2675	.2523

Conclusion

This study addresses the estimation of high-resolution peak electricity demand when only lower-resolution load data is available. To tackle the task, we developed predictive models using two modeling frameworks: generalized additive models (GAM) and deep neural networks (DNN). The ensemble that combines predictions from these two approaches achieved the highest score in the WPD data competition. The official test month for the competition was September 2021. To further assess the robustness of the proposed approach, we also evaluated performance over eleven additional months.

A detailed analysis was carried out of the various model components and their relative contributions. The findings consistently show that the most influential variables are the half-hourly load L_t , its adjusted second-order derivative L''_t , and solar irradiance. This holds for both the GAM and DNN architectures. Combining the two model classes yields better overall accuracy and produces more stable predictions than either approach used in isolation. The effectiveness of the method was confirmed through a comprehensive rolling-window evaluation covering a full 12-month period.

In summary, the combined model reduces RMSE by 1.2% relative to the strongest single model (GAM.full) and by 57.4% compared with the naive benchmark. Incorporating weather variables proved clearly advantageous. A comparison between GAM.full and GAM.noWeather reveals that the inclusion of meteorological inputs improves the skill score by 1.6 percentage points on average.

The methodology presented here can be readily transferred to other power substations. An important open question remains regarding how model performance would vary if the data originated from a different geographic region. Should the results prove generalizable, network operators could avoid the expense of collecting high-resolution measurements altogether. Instead, the required high-frequency peak load characteristics could be derived directly from lower-resolution data using the techniques developed in this work. Exploring the generalizability of the approach across regions constitutes a promising direction for future research.

Acknowledgements

None

Conflict of interest

None

Financial support

None

Ethics statement

None

Received: 17 Sep 2024

Revised: 24 Oct 2024

Accepted: 02 Dec 2024

Published online: 20 March 2025

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

Zheng J, Gao DW, Lin L. Smart meters in smart grid: An overview. In: 2013 IEEE Green Technologies Conference (GreenTech). IEEE; 2013. p. 57–64.

Sănduleac M, Ciornei I, Toma L, Plămnescu R, Dumitrescu A-M, Albu MM. High reporting rate smart metering data for enhanced grid monitoring and services for energy communities. *IEEE Trans Ind Inf.* 2021;18(6):4039–48.

Mughees N, Mohsin SA, Mughees A, Mughees A. Deep sequence to sequence Bi-LSTM neural networks for day-ahead peak load forecasting. *Expert Syst Appl.* 2021;175:114844.

Lee J, Cho Y. National-scale electricity peak load forecasting: Traditional, machine learning, or hybrid model? *Energy.* 2022;239:122366.

Uddin M, Romlie MF, Abdullah MF, Abd Halim S, Kwang TC, Shafiu Baharin SA. A review on peak load shaving strategies. *Renew Sustain Energy Rev.* 2018;82:3323–32.

Lissa P, Deane C, Schukat M, Seri F, Keane M, Barrett E. Deep reinforcement learning for home energy management system control. *Energy AI.* 2021;3:100043.

Sun M, Wang Y, Strbac G, Kang C. Probabilistic peak load estimation in smart cities using smart meter data. *IEEE Trans Ind Electron.* 2018;66(2):1608–18.

Chou J-S, Tran D-S. Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders. *Energy.* 2018;165:709–26.

Chen Y, Xu P, Chu Y, Li W, Wu Y, Ni L, et al. Short-term electrical load forecasting using the support vector regression (SVR) model to calculate the demand response baseline for office buildings. *Appl Energy.* 2017;195:659–70.

Haben S, Giasemidis G, Ziel F, Arora S. Short term load forecasting and the effect of temperature at the low voltage level. *Int J Forecast.* 2019;35(4):1469–84.

Guo Z, Zhou K, Zhang X, Yang S. A deep learning model for short-term power load and probability density forecasting. *Energy.* 2018;160:1186–200.

Xie J, Chen Y, Hong T, Laing TD. Relative humidity for load forecasting models. *IEEE Trans Smart Grid.* 2016;9(1):191–8.

Dehalwar V, Kalam A, Kolhe ML, Zayegh A. Electricity load forecasting for urban area using weather forecast information. In: 2016 IEEE International Conference on Power and Renewable Energy (ICPRE). IEEE; 2016. p. 355–9.

Cai M, Pipattanasomporn M, Rahman S. Day-ahead building-level load forecasts using deep learning vs. traditional time-series techniques. *Appl Energy.* 2019;236:1078–88.

Muzaffar S, Afshari A. Short-term load forecasts using LSTM networks. *Energy Procedia.* 2019;158:2922–7.

Acaroğlu H, García Márquez FP. Comprehensive review on electricity market price and load forecasting based on wind energy. *Energies.* 2021;14(22):7473.

Hong T, Wang P, White L. Weather station selection for electric load forecasting. *Int J Forecast.* 2015;31(2):286–95.

Vu DH, Muttaqi KM, Agalgaonkar AP, Bouzerdoum A. Short-term electricity demand forecasting using autoregressive based time varying model incorporating representative data adjustment. *Appl Energy.* 2017;205:790–801.

Aguilar Madrid E, Antonio N. Short-term electricity load forecasting with machine learning. *Information.* 2021;12(2):50.

Zhang X, Wang J, Zhang K. Short-term electric load forecasting based on singular spectrum analysis and support vector machine optimized by Cuckoo search algorithm. *Electr Power Syst Res*. 2017;146:270–85.

Sheng F, Jia L. Short-term load forecasting based on SARIMAX-LSTM. In: 2020 5th International Conference on Power and Renewable Energy (ICPRE). IEEE; 2020. p. 90–4.

Fan G-F, Yu M, Dong S-Q, Yeh Y-H, Hong W-C. Forecasting short-term electricity load using hybrid support vector regression with grey catastrophe and random forest modeling. *Util Policy*. 2021;73:101294.

Pierrot A, Goude Y. Short-term electricity load forecasting with generalized additive models. In: Proceedings of ISAP Power. Vol. 2011. 2011.

Goude Y, Nedellec R, Kong N. Local short and middle term electricity load forecasting with semi-parametric additive models. *IEEE Trans Smart Grid*. 2013;5(1):440–6.

Ziel F. Smoothed Bernstein online aggregation for short-term load forecasting in IEEE DataPort competition on day-ahead electricity demand forecasting: Post-COVID paradigm. *IEEE Open Access J Power Energy*. 2022.

Amato U, Antoniadis A, De Feis I, Goude Y, Lagache A. Forecasting high resolution electricity demand data with additive models including smooth and jagged components. *Int J Forecast*. 2021;37(1):171–85.

Tasre MB, Bedekar PP, Ghatte VN. Daily peak load forecasting using ANN. In: 2011 Nirma University International Conference on Engineering. IEEE; 2011. p. 1–6.

Pallonetto F, Jin C, Mangina E. Forecast electricity demand in commercial building with machine learning models to enable demand response programs. *Energy AI*. 2022;7:100121.

Hosein S, Hosein P. Load forecasting using deep neural networks. In: 2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT). IEEE; 2017. p. 1–5.

Amarasinghe K, Marino DL, Manic M. Deep neural networks for energy load forecasting. In: 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE). IEEE; 2017. p. 1483–8.

Khwaja AS, Anpalagan A, Naeem M, Venkatesh B. Joint bagged-boosted artificial neural networks: Using ensemble machine learning to improve short-term electricity load forecasting. *Electr Power Syst Res*. 2020;179:106080.

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.

Memarzadeh G, Keynia F. Short-term electricity load and price forecasting by a new optimal LSTM-NN based prediction algorithm. *Electr Power Syst Res*. 2021;192:106995.

Shaqour A, Ono T, Hagishima A, Farzaneh H. Electrical demand aggregation effects on the performance of deep learning-based short-term load forecasting of a residential building. *Energy AI*. 2022;8:100141.

Bashir T, Haoyong C, Tahir MF, Liqiang Z. Short term electricity load forecasting using hybrid prophet-LSTM model optimized by BPNN. *Energy Rep*. 2022;8:1678–86.

Khan ZA, Ullah A, Haq IU, Hamdy M, Maurod GM, Muhammad K, et al. Efficient short-term electricity load forecasting for effective energy management. *Sustain Energy Technol Assess*. 2022;53:102337.

Gneiting T. Making and evaluating point forecasts. *J Amer Statist Assoc*. 2011;106(494):746–62.

Székely GJ, Rizzo ML, Bakirov NK. Measuring and testing dependence by correlation of distances. *Ann Statist.* 2007;35(6):2769–94.

Ziel F. M5 competition uncertainty: Overdispersion, distributional forecasting, GAMLSS, and beyond. *Int J Forecast.* 2021.

März A, Kneib T. Distributional gradient boosting machines. 2022. arXiv:2204.00778.

Wood S. Generalized additive models: an introduction with R. 2nd ed. Chapman and Hall/CRC; 2017.

Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A next-generation hyperparameter optimization framework. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.* 2019. p. 2623–31.

Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-scale machine learning on heterogeneous systems. 2015. Available from: <https://www.tensorflow.org/>.

Chollet F, et al. Keras. 2015. Available from: <https://keras.io>.

Lago J, Marcjasz G, De Schutter B, Weron R. Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. *Appl Energy.* 2021;293:116983.

Takeda H, Tamura Y, Sato S. Using the ensemble Kalman filter for electricity load forecasting and analysis. *Energy.* 2016;104:184–98.