Computers and Electrical Engineering 116 (2024) 109145

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Industrial kitchen appliance consumption forecasting: Hour-ahead and day-ahead perspectives with post-processing improvements

Vasco Andrade [a], Hugo Morais [a,b], Lucas Pereira [a,c,*]

[a] Instituto Superior Técnico - IST, Universidade de Lisboa, Lisboa, 1049-001, Portugal
[b] INESC-ID — Instituto de Engenharia de Sistemas e Computadores-Investigacão e Desenvolvimento, 1049-001, Portugal
[c] Interactive Technologies Institute, LARSyS, Rua da Manutenção 71, Building F, S05, Lisboa, 1900-500, Portugal

ARTICLE INFO

Keywords:
Appliance consumption forecasting
Industrial kitchen
Day-ahead
Hour-ahead
Post-processing

ABSTRACT

Forecasting techniques have gained considerable prominence within the electric energy sector. Many studies have been documented in the literature, addressing various facets of the energy grid, ranging from power generation to end-user consumption. However, it is noteworthy that the prediction of individual appliance demand has remained relatively unexplored despite its increasing significance, particularly in modern power grids characterized by a dominant presence of distributed energy resources. In light of this research gap, this work focuses on developing and evaluating methodologies for forecasting active power consumption at the device level in the context of industrial kitchens. Three post-processing algorithms are also proposed to improve the forecasting accuracy by leveraging historical predictions. A comprehensive case study employing sub-metered data from 15 industrial kitchen devices was conducted to validate the proposed methods, spanning both hour-ahead and day-ahead scenarios. The results demonstrate the effectiveness of the proposed methods in both forecasting horizons, particularly of the post-processing techniques that show average improvements of over 30% in day-ahead and 50% in hour-ahead, compared to the original predictions.

1. Introduction

Energy efficiency has emerged as a highly debated subject due to the rising global energy demand and the goals set by various governments to curb carbon emissions. In the European Union (EU), energy production and utilization alone contribute to 80% of total Greenhouse Gas (GHG) emissions [1]. Data from the United Kingdom (UK) indicates low energy efficiency in the catering industry, with an excess in consumption estimated in 21 600 million kWh per year [2].

Studies indicate that in the EU, UK, and the United States (US), 30% of the energy consumed in Industrial Kitchens (IKs) is used in purely commercial establishments such as restaurants and snack bars [3]. Still, although the industry is large and prevalent, its contribution to the global pursuit of sustainable energy systems remains largely unexplored [4]. A study [5] examining the literature on sustainable restaurants from 1991 to 2015 revealed that most works only address certain aspects of sustainability, such as ecological factors, rather than taking a comprehensive approach to sustainability as a whole. The electrification of IKs has a significant potential to contribute to the energy transition targets by enabling the integration of Distributed Energy Resourcess (DERs) [6] and the participation of such spaces in electricity markets that are being created specifically from smaller consumers [7].

In this regard, load forecasting plays a crucial role due to the nature of the operation of IKs. These establishments heavily rely on energy-intensive equipment for meal preparation, including cooking appliances, refrigeration units, lighting systems, and hot

* Corresponding author at: Instituto Superior Técnico - IST, Universidade de Lisboa, Lisboa, 1049-001, Portugal.
E-mail address: lucas.pereira@tecnico.ulisboa.pt (L. Pereira).

https://doi.org/10.1016/j.compeleceng.2024.109145
Received 26 November 2023; Received in revised form 11 February 2024; Accepted 22 February 2024
Available online 1 March 2024
0045-7906/

water supplies [8]. Therefore, accurately predicting the consumption of electric devices is important to implement strategies that promote better coordination between using IKs' devices, hence increasing the benefits from the DER integration and participation in electricity markets.

Numerous approaches have been explored to predict electricity demand, as presented in several literature review papers [9–12], with works spanning different points of the grid, including transmission [13], distribution [14], and total household demand [15]. However, only a few works have considered forecasting the demand of individual appliances, e.g., [16,17], including what is, to the best of the authors' knowledge, the only work that considers IK devices [18].

Against this background, this work focuses on developing and evaluating methodologies for forecasting active power consumption at the device level in the context of IKs considering hour-ahead and day-ahead forecasting horizons. Day-ahead analysis, with one sample every 30 min, provides insights into overall daily load patterns. The hour-ahead analysis, 5-minute samples, focuses on the detailed behavior of individual loads. To improve the forecasts, three post-processing methods were investigated, and their individual performances were analyzed and benchmarked. To summarize, this work makes the following original research contributions:

1. A comprehensive benchmark of the performance of three well-known load forecasting algorithms – Extreme Gradient Boosting (XGBoost), Categorical Boosting (CatBoost), and Long-Short Term Memory (LSTM) deep neural networks – when applied to consumption data of 15 devices that were monitored from one IK during its daily operation. Before the comparison, all the models were optimized to select the best historical window size and the optimal hyper-parameters for each device under test. The comparison was done based on two aspects: the quality of the forecasts and the time required to train each model.
2. The implementation and evaluation of three post-processing techniques with target selection that leverage historical predictions to improve the performance of the forecast results: (1) Persistence-based Prediction (PbP), (2) Most Recent Prediction (MRP), and (3) Weight-based Prediction (WP). These techniques were applied to the forecasting results of the best algorithms that emerged from the previous benchmark. The resulting performances were compared with those without a post-processing step and from a baseline post-processing technique - Average Prediction (AbP).

The remainder of this paper is organized as follows. Section 2 reviews the state of the art on load forecasting. Related works in the field of power forecasting are presented. Section 3 presents in detail the methods proposed and implemented in this work. Section 4 presents the case study that was developed to evaluate the proposed methods, including a description of the appliance consumption data. The main results of this work are presented and discussed in Section 5. Finally, Section 6 presents the main conclusions of this work, highlighting its limitations and future work directions.

2. Related works

Although research into energy consumption in IKs is scarce, the forecasting of both aggregated loads and individual appliances has garnered significant attention from the research community. This section provides an overview of some of these efforts, including studies on post-processing methods for power forecasting.

2.1. Aggregated consumption forecasting

Numerous approaches have been explored to predict load demand curves. Authors in [19] combine the Autoregressive Integrated Moving Average (ARIMA) statistical model with clustering techniques to forecast a university load profile accurately. Load forecasting tends to present a more time-correlated behavior at the aggregate level [20] since the electricity demand in this scenario exhibits significant changes during the month but slow variations from day to day. For that reason, authors [20] consider a simple persistence model for load forecasting in a shopping mall. Despite using a simple model, the persistence exhibits good accuracy.

Machine Learning (ML) methods such as Support Vector Machines (SVMs) present good results in household short-term load forecasting [15], especially when compared with statistical methods such as the ARIMA. Despite that, the authors state that SVMs do not perform adequately when forecasting load consumption peaks. Another research [21] compares an SVM with an Artificial Neural Networks (ANNs). Using datasets with different business days, the authors conclude that SVMs can outperform ANNs in forecasting accuracy despite requiring approximately the same computational time. Authors in [22] propose a robust Support Vector Regression (SVR) model to forecast the electricity demand under data integrity attacks, introducing a weight function to calculate the relative importance of each observation in the load history.

To perform forecasts at the distribution level, authors in [23] suggest a short-term load forecasting method based on XGBoost that selects similar days to construct a feature map for load forecasting. The proposed approach outperformed the traditional implementation of XGBoost and LSTM models. At the transmission level, a combination of LSTM and XGBoost models is presented in [24] to perform day-ahead forecasts. In this work, XGBoost is presented as a tool to evaluate feature importance with similar day detection. This hybrid approach in forecasting outperformed the standard approaches of LSTM, ARIMA, and SVM.

2.2. Individual appliance consumption forecasting

Appliance-level forecasting has not received much research attention compared to aggregate-level forecasting. However, in recent years, a few works have been published exploring this area, especially due to the need to account for the stochastic nature of user behaviors and external factors that affect the consumption patterns of individual appliances [25].

In this regard, to make day-ahead predictions on the load profile of refrigerators, authors in [16] use clustering techniques alongside Fuzzy-Rule-based and SVM models. Here, researchers use historical data relative to several months to obtain and generate the daily profiles of 20 000 refrigerators, with the SVM model presenting a lower accuracy than the Fuzzy-Rule-based model.

The authors in [26] apply LSTM Deep Neural Network (DNN) and compare them with algorithms such as Feed-Forward Neural Networks (FFNNs) and Random Forests (RFs) to perform day-ahead forecasts for different household appliances such as TV, fridge, laptop, electric heater, cookers, and microwaves. Ultimately, the LSTM models outperformed the other two alternatives. Other DNN-based approaches have also been tested, but in the context of short-term forecasting. For example, [27] the authors proposed an LSTM-based Sequence to Sequence (seq2seq) model to forecast hour-ahead energy consumption with 10-minute resolution using historical data of the past day. The proposed model was evaluated against several appliances across four households, including dishwashers, glass washers, TVs, and refrigerators. The results show that the proposed LSTM-based seq2seq model outperforms other techniques, including a Convolutional Neural Network (CNN) and a 2-layer LSTM network.

In [17], the authors proposed a Conditional Hidden semi-Markov Model (CHSMM) to produce short-term forecasts (6-hours horizon with 15-minute data) of selected household loads (Air-Conditioning (AC), pool pump, water heater, Electric Vehicle (EV), and refrigerator) from 20 households. The proposed approach was benchmarked against a standard Hidden semi-Markov Model (HSMM), showing that the proposed method was able to outperform the baseline with respect to thermal (ACs, and refrigerators) and fixed scheduled loads (poop pumps). For the other loads, both algorithms have shown a similar performance. CHSMMs were also studied by the authors in [28] to explore how different training methods affect forecasting performance in the same 6-hours horizon. It was concluded that selecting the most adequate training method heavily depends on the characteristics of the appliance. Still, a limitation of this study is the fact that only two very different types of appliances have been considered (Refrigerators and AC).

Concerning the forecast of IK devices, [18] is the only published work to the best of the author's knowledge. In this work, the authors benchmarked three forecasting algorithms (Prophet, RF, and LSTM) to predict the hour-ahead demand of 17 devices (with 5-minute samples) as a way to evaluate if the sum of individual forecasts would outperform the forecasts of the aggregated demand (i.e., the sum of the 17 devices). Ultimately, the RF had the best overall performance for device load forecast, having the best forecasting performance in 10 out of 17 of the appliances. In contrast, the LSTM had the worst performance, outperforming the other models in only two appliances. However, this work also suffers from an important limitation, which is the fact that no hyper-parameter tuning was performed. Instead, all the models were trained with the default values independently of the device.

2.3. Post-processing methods in power forecasting

In the field of forecasting, post-processing methods aim at improving the model performance by changing the predictions according to some pre-defined criteria or technique [29]. In the energy field, such techniques are fairly common when it comes to forecasting production from Renewable Energy Sources (RESs), notably from Wind and Photovoltaic (PV), e.g., [29–31].

Of particular relevance for this research is the work in [30], where the authors evaluate six post-processing techniques to improve short-term (day ahead) off-shore wind power forecasting. Three of them relied on past predictions: (1) combining the present forecast with the one with the previous 24 h (D1/D2 Mix), (2) assuming persistence in the power production in the following day (Pers), and (3) a moving average with 24 h lead time (Smooth). Experimental results have demonstrated the feasibility of these approaches, with forecast improvements in the three methods (D1/D2: 30%, Pers: 20%, and Smooth: 29%).

To the best of the authors' knowledge, post-processing in electricity forecasting is not common in the literature. Instead, a common approach is to rely on meta-learners that select the most appropriate forecasting algorithms for a particular context (e.g., [32]). Still, besides the need to train different models, the main drawback of these techniques is that they are not dataset-agnostic. One of the few published papers in post-processing electricity forecasts is the work in [33], which proposes a rule-based post-processing method that blends human expertise and domain knowledge. This method was evaluated for day-ahead demand forecasting with post-processing applied to extreme events, and the results show increased performance. Still, the main drawback of such a method is its reliance on domain expertise, which varies greatly depending on the problem at hand.

3. Methods

The proposed methodology is depicted in Fig. 1. The first step consists of applying and benchmarking different forecasting algorithms using IK appliance data (Appliance Load Forecast block). Then, the best resulting models are post-processed using four different methods that leverage past predictions to improve the forecasting performance (Model Post-Processing block). The performance of the different proposed methods is assessed using different performance metrics for forecasting and computational time, considering each appliance and also the respective IK average (Performance Evaluation block). The following sections thoroughly explain the three major blocks in the proposed methodology.

3.1. Appliance load forecasting

3.1.1. Algorithms

In this work, the forecast algorithms used are XGBoost, CatBoost, and LSTM deep neural network. This selection was made based on the results observed in Section 2 where it is shown that boosting algorithms (XGBoost, CatBoost) and neural networks (LSTM) are among the most widely used algorithms for other forecasting studies. A brief description of the algorithms is provided next.

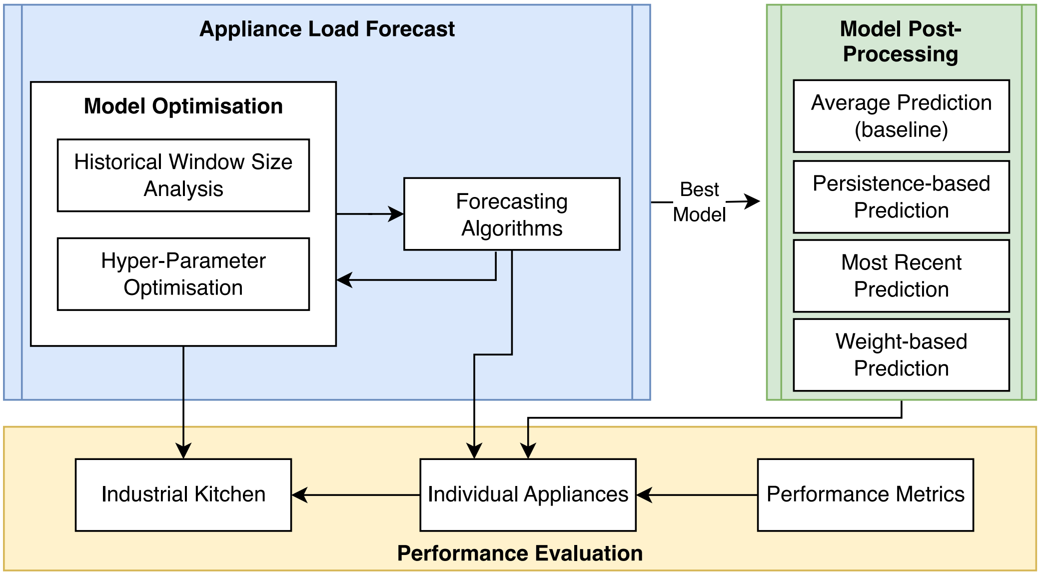


Fig. 1. Overview of the proposed methodology and the underlying methods.

XGBoost. Extreme Gradient Boosting uses gradient boosting algorithms combined with decision trees to achieve high accuracy and efficiency. It is particularly effective for high-dimensional datasets. In this algorithm, decision trees (also called weak learners) are created sequentially, perform a prediction, and are sequentially added to the model [34]. The final result is a strong, upgraded predictor. The mathematical formulation for XGBoost is detailed in Eq. (1):

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{1}$$

where, K is the number of trees, f is the functional space of F, being F the set of possible decision trees.

In XGBoost, each new learner is trained to minimize the residual error of the previous model. This is done by optimizing an objective function, which measures the difference between the predicted and actual values of the target variable [34]. One key feature of XGBoost is the use of regularization to prevent over-fitting. The objective function with regularization terms is presented in Eq. (2) [35], where the first term is the loss function and the second is the regularization parameter.

$$obj(\theta) = \sum_{i}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{2}$$

CatBoost. Categorical Boosting has several similarities with XGBoost, but also some significant differences, especially in how both handle categorical variables. XGBoost requires that categorical variables be pre-processed into numerical features before training, while CatBoost can handle categorical features directly [36]. CatBoost introduces two additional terms to handle these features: the categorical feature index j and the categorical feature value v. The split gain is calculated for each categorical feature, and the feature with the highest gain is selected for the split [37]. Since its proposal in 2018, this has been one of the most widely used algorithms for several forecasting problems, consistently reporting superior performance compared to other boosting strategies [38].

LSTM. Long Short-Term Memory networks were introduced to overcome the vanishing gradient problem in Recurrent Neural Networks (RNNs) by incorporating a special memory cell that can selectively remember or forget information over time. The memory cell comprises three gates [39]: input, forget, and output gates. The input to the LSTM is a sequence of vectors, such as a time series, and the output is also a sequence of vectors, the predicted values. The LSTM layers can be stacked to increase the depth of the network and improve its ability to capture complex dependencies in the input sequence. Still, it also increases the processing time of the network. A mathematical formulation of the nodes present in a cell is analytically described in Eqs. (3) to (7):

$$i_t = \sigma(W_{xi}X_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{3}$$

$$f_t = \sigma(W_{xf}X_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{4}$$

$$c_t = f_t c_{t-1} + i_t\, tanh(W_{xc}X_t + W_{hc}h_{t-1} + b_c) \tag{5}$$

$$o_t = \sigma(W_{xo}X_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \tag{6}$$

Table 1
Values tested for the historic window size.

	Day-ahead	Hour-ahead
Window size (hours)	24, 48, 72	2, 4, 6, 8, 12, 24
Number of samples	48, 96, 144	24, 48, 72, 96, 144, 288

Table 2
Hyper-parameters for optimization.

Algorithms	Parameters	Range of values
XGBoost/Catboost	Learning rate	0.005, 0.01, 0.02
	Maximum depth of each tree	2, 3, 4, 5
	Column sample by tree	[0.2, 0.8]
	Subsample	[0.2, 0.8]
	Regularization parameters λ, α	1, 5, 10, 50
LSTM	Number of neurons in each layer	16, 32, 64
	Number of LSTM layers	1, 2, 3
	Activation function	tanh, linear, sigmoid
	Dropout rate in each neuron	0, 0.1, 0.2

$$h_t = o_t \, tanh(c_t) \tag{7}$$

where c_t, i_t, f_t, and o_t are the state cell, input gate, forget gate, and output gate at the time instant t. Also, the final output from the LSTM unit is represented as h_t. x_t is the current sample, and the activation function σ is the *sigmoid* function. This algorithm can be computationally expensive, especially for long input sequences [40].

3.1.2. Model optimization

Three different aspects were considered when optimizing the forecasting models. First, the size of the historical (or context) window. Second, the size of the training dataset. Finally, the optimal values for the most relevant hyper-parameters of each forecasting algorithm were selected through Hyper-Parameter Optimization (HPO). A brief description of each step is provided next.

Historical window size analysis. The historical window size is crucial to feed the model data from which it can extract the patterns of load consumption. Authors in [41] perform a sensitivity analysis ranging from 1 to 30 days of historic window size and identify the optimal length as five days for day-ahead forecasts. Generally, the accuracy of the prediction model improves as the size of the training window increases, i.e., it is better to have sizeable historical data for training the prediction model so that it covers all possible patterns spanning time series [42]. However, a significant drawback of having large historical data is that if the data's behavior changes quickly, the trained model may not track or ignore those changes, resulting in inaccurate forecasts. Moreover, it is expected that due to the different types of appliances studied in this work, some may require a larger historical window to properly identify patterns. In contrast, other appliances might perform better considering small windows.

In this work, the maximum window size for the day-ahead scenario was set to three days, mainly due to the small size of the available datasets. However, for hour-ahead forecasts, there is the possibility of implementing the models with a wider range of historical periods since the prediction window is much smaller. This way, the values chosen to analyze the historical window size influence in the forecast performance are presented in Table 1.

Hyper-parameter optimization. ML and Deep Learning (DL) algorithms rely on hyper-parameters, which must be defined beforehand. Since these hyper-parameters can significantly affect performance, their values must be selected carefully [43]. To this end, HPO aims to find the combination of hyper-parameters that yields the best performance on the given dataset. This is typically done by performing a search over a range of possible values, using techniques such as grid search, random search, or Bayesian optimization [44]. The model's performance is evaluated using a validation set, and the hyper-parameter values that yield the best performance are selected.

In this work, Bayesian optimization is used to tune the models, leveraging the *Optuna* framework in Python [45]. This type of optimization involves constructing a probabilistic model of the objective function, building a surrogate for the objective, and quantifying the uncertainty in that surrogate using a Bayesian machine learning technique [46]. The optimized hyper-parameters and the respective range of values are summarized in Table 2. A total of 20 trials was executed for each algorithm, upon which the best performing model was selected and the respective hyper-parameters recovered. All the remaining hyper-parameters were set to the default values in the respective implementations (refer to Section 4.3 for additional details on the software versions.).

Apart from the parameters tuned using the *Optuna* library, for the boosting algorithms, the number of trees to be built was defined as $n\_estimators = 3000$. For the LSTM model, two deeply connected layers were also added, with the first layer presenting

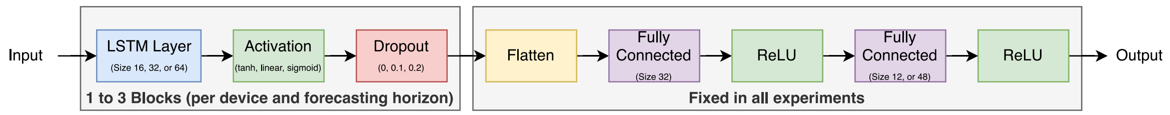


Fig. 2. Overall architecture of the LSTM Network used in this work.

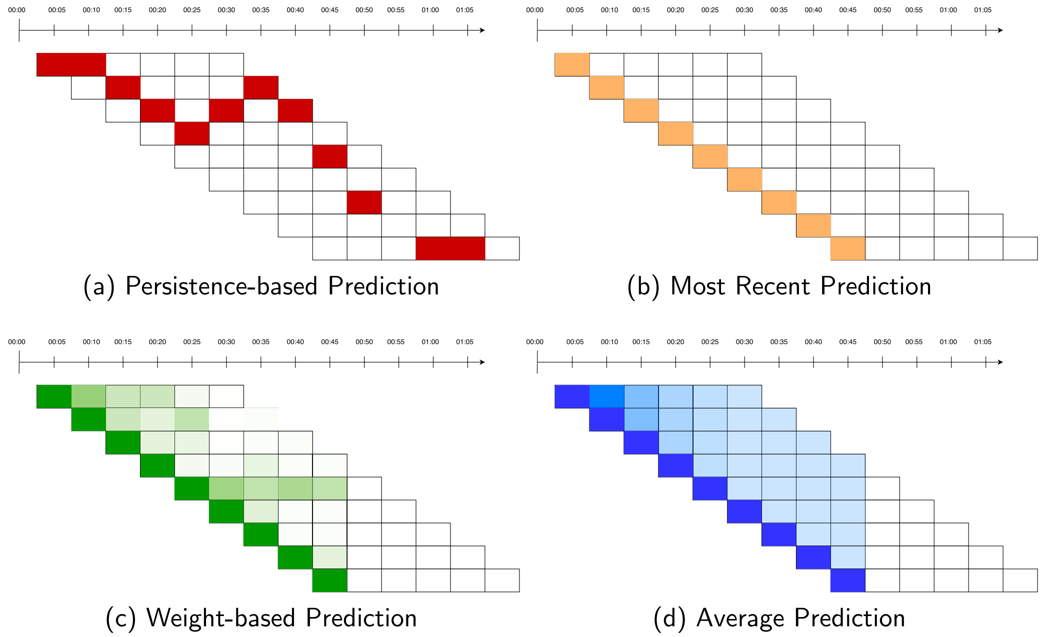


Fig. 3. Illustration of how the predicted value is updated in each of the four methods.

32 neurons and the last layer presenting a number of neurons corresponding to the number of targets of the analysis. An overview of the LSTM architecture used in this work is provided in Fig. 2. Note that the number of LSTM blocks layers will vary between 1 and 3, according to the HPO parameters defined in Table 2.

3.2. Model post-processing

Since the forecast algorithms are trained using a sliding window approach, the output of the forecast models also consists of multiple sliding windows, forming a 2D matrix where each line contains the forecast of the next W_{pred} values. This means that the predicted windows overlap, and consequently, there is more than one prediction for each timestamp. With this, updating the predicted values at each timestep t is possible, which, if done properly, can result in improved performances. Four methods are compared in this work.: (1) PbP, (2) MRP, (3) WP, and (4) AbP (baseline method). The four methods are illustrated in Fig. 3.

3.2.1. Persistence-based prediction

The optimization using a persistence model assumes that for each prediction window, the results of the forecasting model present a trend in the results and that the same iteration of prediction presents a similar error for the timestamps in the different windows. In other words, it assumes that, for example, the first prediction for the time instant t presents a similar error in relation to the first prediction for the time instant $t+W_{pred}$. Hence, this method analyzes the first W_{pred} timestamps and finds which prediction presents the lowest error for each t. For $W_{pred}+t$, it is assumed that the most accurate prediction is at the same relative distance in time as it was for the first W_{pred} points analyzed.

An example of how the predicted value is selected with this approach is presented in Fig. 3(a), considering a simple scenario with $W_{pred}=6$. Here, it is possible to understand the pattern selected in the predictions. With the first two timestamps being obtained by selecting the oldest values available (for 00:05 and 00:10), the first two timestamps after W_{pred} (00:35 and 00:40) are also obtained by selecting the oldest prediction available.

3.2.2. Most recent prediction

Using the most recent prediction for a timestamp is straightforward as it is assumed that the prediction for a point in time t is given by the last value predicted for the same instant t. In other words, it is the prediction made at the time instant $t-1$, which, in theory, represents the prediction with the most up-to-date information. An example of how the predicted value is selected with this approach is presented in Fig. 3(b).

3.2.3. Weight-based prediction

The weighted prediction method is based on the Roth–Erev algorithm, presented in Eq. (8), which considers two features: the previous coefficient value and the error made in the last prediction [47]. The algorithm uses a weight factor W that acts as an inertia constant, with a value between 0 and 1. It is regulated based on the dependence of each appliance on the past values to predict the future load accurately. The closer the weight value is to 1, the larger the dependency on the past values.

$$C_{t+1}^{n} = C_{t}^{n} \times W - |(R_t - P_t^n)| \times (1 - W) \tag{8}$$

In Eq. (8), coefficients C_{t+1}^{n} and C_t^n are one dimensional vectors, with C_t^n representing the mean errors between each prediction position n for the historical values until t and test values. This way, C_{t+1}^{n} is also a 1-D vector, where its positions n are determined individually with the historical and actual error values. Each position represents the prediction number of the respective time instant: if position n yields the lowest error for t, it means that the n_{th} prediction made for t is the most correct one. Then, this index is saved and used to set the n_{th} prediction as the most accurate value for $t+1$. This implies that the final prediction for time instant t can never present an error lower than the lowest error prediction for the same instant. The error values for $|(R_t - P_t^n)|$ are then added to the historical component. This process is carried out until the end of the prediction dataset is reached. An illustration of this method is provided in Fig. 3(c).

Since the final prediction is a function of two components, the sum of the weight values that multiply by these components must be enforced to 1. However, the appliances are expected to have different dependencies of the past and actual error values. This way, a sensitivity analysis was performed about the weight parameter to understand which values better suit each appliance. The values chosen for the sensitivity analysis were 0.05, 0.1, a.25, 0.5, 0.75, 0.9, and 0.95.

3.2.4. Baseline: Average prediction

The baseline method works by assigning to each instant t the average of the previous predictions for that instant. An example of how the predicted value is calculated with this approach is presented in Fig. 3(d). As can be seen, the number of samples to calculate the average increases with time until W_{pred} is reached. An illustration of this method is provided in Fig. 3(d).

3.3. Performance evaluation

The forecasts were evaluated in three different stages: (1) during the model optimization stage to obtain the best hyper-parameters and historic window size; (2) at the output of the forecast models to provide an overview of the forecast quality; and (3) after the post-processing techniques, to provide a more conclusive result about the methodology. In each case, the performance metrics are reported considering the average of all the prediction windows.

To quantify the performance, the RMSE was used (Eq. (9)). The RMSE metric penalizes variance as it gives more weight to larger errors [48].

Moreover, since the used dataset consists of a diverse set of appliances, some of which have very distinct working patterns and levels of consumption, two normalized versions of the RMSE were also used. The MXNRMSE metric is used in stage one and is normalized by the maximum consumption value observed in each device (Eq. (10)). The SDNRMSE metric is used in stages two and three, and it is normalized by the standard deviation observed for each device to enable a fairer comparison across devices (Eq. (11)).

$$RMSE(y,\hat{y}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y(t)-\hat{y}(t))^2} \tag{9}$$

$$MXNRMSE(y,\hat{y}) = \frac{1}{max(Y)} \times RMSE(y,\hat{y}) \tag{10}$$

$$SDNRMSE(y,\hat{y}) = \frac{1}{std(Y)} \times RMSE(y,\hat{y}) \tag{11}$$

Where $y(t)$ stands for the truth value of power at the instant t, while $\hat{y}(t)$ is the predicted value for the same time point, $max(y)$ is the maximum value in the dataset, $std(Y)$ is the standard deviation value in the entire dataset, and N represents the number of samples. The $max(Y)$ and $std(Y)$ are determined after the data pre-processing step.

4. Case study specification

The proposed methodology was evaluated using consumption data from 15 appliances that were monitored from one IKs during its daily operation. The present section describes the appliance consumption data, the necessary data preparation steps, and the hardware/software configurations that were used to produce the experiments.

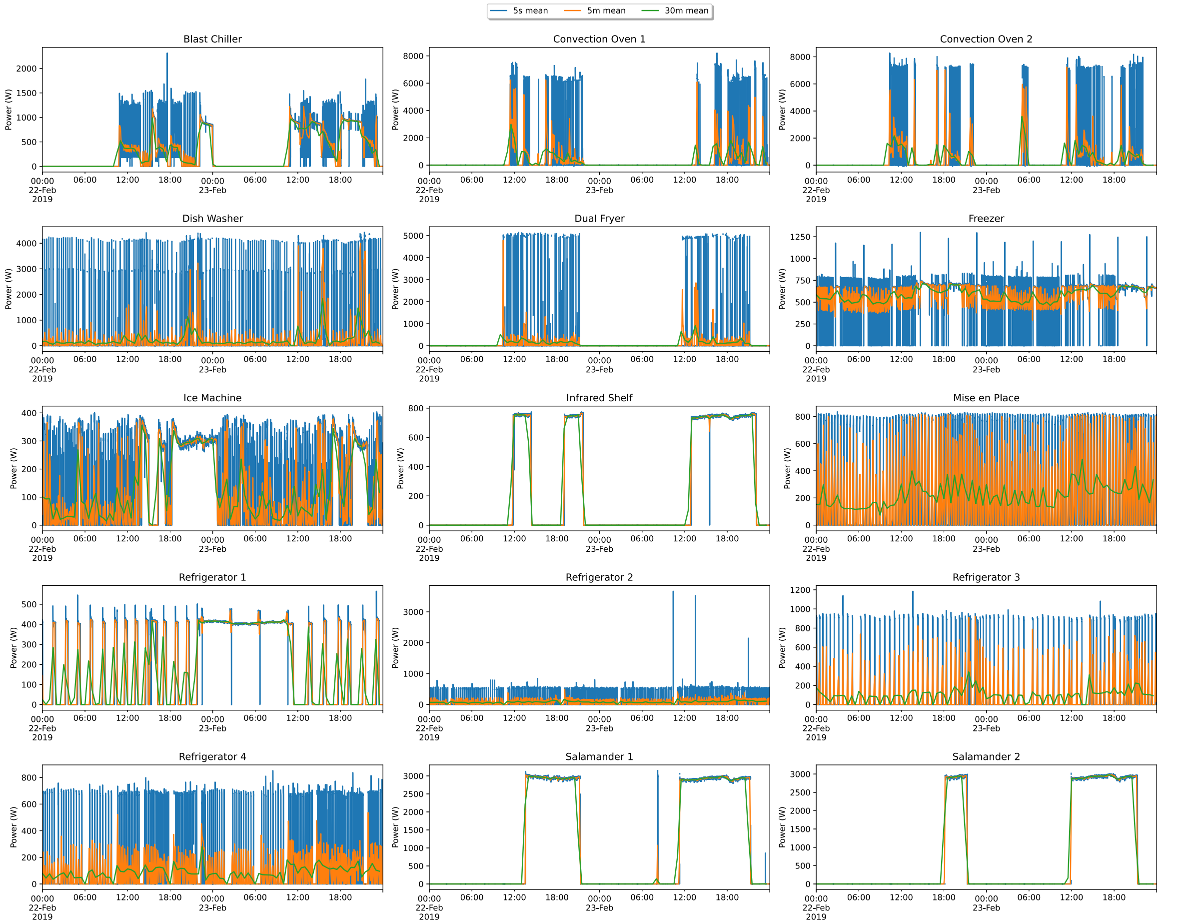


Fig. 4. Two days of active power for each appliance considered. The data is shown using three different sampling rates: 0.5 Hz (original), 1/5 Hz, and 1/30 Hz.

4.1. Appliance consumption data

The consumption data for the 15 appliances was taken from the FIKElectricity dataset [49], more precisely from IK 1. This dataset consists of aggregated and individual appliance electricity consumption data collected from three restaurant kitchens in Funchal, Portugal. The data contains information about electric measurements such as voltage, current, reactive power, active power, and apparent power. In this work, only active power is considered since this is what measures the actual energy consumed.

The electricity consumption was measured using the e-Gauge circuit-level meter[1] at a sampling rate of 1/5 Hz. The period of measurement for IK 1 ranges from the sixth of February of 2019 (6th of February 2019 00:00:00) to the fourth of March at 03:00 (4th of March 2019 03:00:00). An overview of the appliance consumption trace is provided in Fig. 4. Each plot shows two days of data using three different sampling rates: 0.5 Hz (original), 1/5 Hz, and 1/30 Hz.

4.1.1. Data pre-processing

In this work, the input data has a very high sampling rate. This adds a challenge since it implies the forecast of a large number of samples, especially for day-ahead forecasts, representing a challenge for both forecasting accuracy and computational resources. To minimize this problem, data is re-sampled to a higher period according to the type of prediction, namely, thirty minutes for day-ahead forecasts and five minutes for hour-ahead forecasts. This re-sampling is done by computing the mean value of data in the new sampling period.

Missing data and filtering. The data used in this work was already treated before being published. However, no methods were used to fill in the missing data. Therefore, resampling to a constant frequency caused some missing data to appear. More precisely, the missing data represent only around 2.5% of the total data. More importantly, the missing data consists of the majority of one specific

[1] eGauge, https://www.egauge.net/

Features

	1	2	3	4				95	96
Sequence x	00:00 15/02/2019	00:30 15/02/2019	01:00 15/02/2019	01:30 15/02/2019	...	...	...	23:00 16/02/2019	23:30 16/02/2019
Sequence x+1	00:30 15/02/2019	01:00 15/02/2019	01:30 15/02/2019	02:00 15/02/2019	...	...	...	23:30 16/02/2019	00:00 17/02/2019
Sequence x+2	01:00 15/02/2019	01:30 15/02/2019	02:00 15/02/2019	02:30 15/02/2019	...	...	...	00:00 17/02/2019	00:30 17/02/2019

Targets

	1	2	3	4				47	48
Sequence x	00:00 17/02/2019	00:30 17/02/2019	01:00 17/02/2019	01:30 17/02/2019	...	...	...	23:00 17/02/2019	23:30 17/02/2019
Sequence x+1	00:30 17/02/2019	01:00 17/02/2019	01:30 17/02/2019	02:00 17/02/2019	...	...	...	23:30 17/02/2019	00:00 18/02/2019
Sequence x+2	01:00 17/02/2019	01:30 17/02/2019	02:00 17/02/2019	02:30 17/02/2019	...	...	...	00:00 18/02/2019	00:30 18/02/2019

Fig. 5. Example of feature and target sequences — day-ahead.

Table 3
Historic window size analysis — day-ahead (MXNRMSE).

Window size	XGBoost	CatBoost	LSTM
24h	16.05%	15.45%	**25.94%**
48h	**15.52%**	**15.27%**	26.26%
72h	16.07%	15.56%	27.88%

day (12-02), so removing approximately one day of data is not expected to affect the overall data quality. Also, by this analysis, it is possible to notice that the NaN values originated by re-sampling are just a small fraction of the dataset. This way, the missing data points caused by re-sampling were dropped.

A median filter was used to remove some noise created by changing the data sampling rate and to smooth the signal. Median filters have excellent noise-reduction capabilities, particularly for impulse noise reduction [50]. In this regard, a filter window with a size of seven was used, indicating that the median is computed with three points backward, the point itself, and three points forward. This corresponds to 1.5 h in each direction for day-ahead scenarios and fifteen minutes in each direction for the hour-ahead scenario. A simple filter of negative loads was applied to replace negative load values with zero values to deal with misreadings of the data acquisition device.

4.2. Training and testing data partitioning

The models require the data to be separated into features and targets to perform the forecasts. Authors in [51] use the method of sliding windows to reconstruct the data into a multidimensional matrix containing features and targets to smooth the non-linearity and non-stationary behavior of load time series. Here, the features considered are the historical values of power and the targets, the load values to be predicted, but separated into two different matrices. It results in a feature dataset comprising sequences of the previous W_{hist} load values, while the targets comprise sequences of the next W_{pred} values. An example of how these datasets were generated for day-ahead forecasting with two days of historical measurements can be seen in Fig. 5, with $W_{hist} = 96$ and $W_{pred} = 48$.

To avoid train targets being included in the test set as features, a buffer zone was created between the end of the training set and the beginning of the test set, with the size of one historic window. Finally, the active power values were re-scaled to the interval [0,1] for the LSTM network (using min–max normalization), allowing a faster convergence and avoiding numeric instability. This was not necessary for the boosting algorithms since these are not sensitive to the variance in the data [34]. When calculating the performance metrics, the active power values were scaled back to the original range.

4.3. Hardware and software configurations

All the models were trained and evaluated in a 2018 MacbooPro. CPU: 2.7 GHz Quad-Core Intel Core i7; RAM: 16 GB 2133 MHz LPDDR3; OS: Ventura 13.4.1(c).

Table 4
Historic window size analysis — hour-ahead (MXNRMSE).

Window size	XGBoost	CatBoost	LSTM
2h	10.15%	9.80%	13.75%
4h	10.10%	9.77%	14.35%
6h	10.19%	9.82%	14.06%
8h	10.32%	9.91%	12.56%
12h	10.15%	9.74%	**11.42%**
24h	**9.65%**	**9.19%**	14.97%

All the code was developed in Python3 (version 3.10.22) using open-source implementations of the forecasting algorithms and supporting libraries: XGBoost,[2] CatBoost,[3] TensorFlow,[4] scikit-learn,[5] and Optuna.[6]

5. Results and discussion

The results are presented and discussed individually for the model optimization (Section 5.1), device consumption forecasts (Section 5.2), and post-processing methods (Section 5.3).

5.1. Model optimization

5.1.1. Window size results

For this analysis, the models' default hyper-parameters were kept constant while the training size was set to the first 19 days. The remaining six days were used for testing.

Tables 3 and 4 present the results obtained for the day-ahead and hour-ahead scenarios, respectively. The scores represent the average MXNRMSE across the 15 devices.

In the day-ahead scenario, it can be observed that the differences in performance across the three window sizes are very small. This suggests that, on average, very similar performances can be expected independently of the context window size. The results also show that in all the presented cases, the LSTM models are the least performing, whereas the boosting models have comparable performances. Based on this, it was decided to set the historical window size to 48 h for the day-ahead forecasts, i.e., 96 samples of context and 48 samples of the forecasting horizon.

For the hour-ahead scenario, it was possible to test a larger number of context window sizes, ranging from two to 24 h. In this scenario, and in line with the day-ahead case, the boosting algorithms continue to outperform the LSTM. However, in this situation, the difference in performance is smaller. A potential explanation for this effect is that in the hour-ahead analysis, more data is available for training the LSTM models, which increases overall performance. As for the best size for the context windows, the boosting models perform better when seeing 24 h of historical consumption, whereas for the LSTM, the best performance is achieved for 12 h. However, since the results for the LSTM show higher variability across the different window sizes (3.55 percentage points), it was decided to set the historical window size to 24 h for the day-ahead forecasts in all three algorithms, i.e., 288 samples of context and 12 samples of the forecasting horizon.

5.1.2. Hyper-parameter optimization

The results of the HPO analysis were aggregated in six tables, one for each algorithm and scenario. Due to the large size of each table, it was decided to omit them from the main body of the paper. Instead, the interested reader can refer to Appendix A.

5.2. Forecast results

This section presents the forecasting results obtained with the optimal hyper-parameters identified through HPO.

5.2.1. Day-ahead

Table 5 presents the average RMSE and NRMSE values between every prediction and target window for the different methods and appliances. Looking at the RMSE values, it is possible to conclude that the boosting algorithms present similar results, with neither model presenting consistent and better results than the other. The biggest improvement of XGBoost on accuracy was on the

[2] XGBoost, https://github.com/dmlc/xgboost (version 1.7.6).
[3] Catboost, https://catboost.ai (version 1.2.1).
[4] TensorFlow, https://www.tensorflow.org/ (version (2.11.0).
[5] scikit-learn, https://scikit-learn.org/ (version 1.3.0).
[6] Optuna, https://optuna.org/ (version 3.3.0).

Table 5
Forecasting results in day-ahead RMSE, and SDNRMSE.

Device	Standard Deviation (W)	RMSE (W)			SDNRMSE		
		XGBoost	CatBoost	LSTM	XGBoost	CatBoost	LSTM
Blast Chiller	265.4	193.6	186.3	207.5	0.73	0.70	0.78
Convection Oven 1	334.3	247.1	248.5	281	0.74	0.74	0.84
Convection Oven 2	343.8	288.9	283	302.4	0.84	0.82	0.88
Dish Washer	257.4	160.6	162.9	203	0.62	0.63	0.79
Dual Fryer	79.7	31.3	35	44	0.39	0.44	0.55
Freezer	97.7	54.1	48.6	203.6	0.55	0.50	2.08
Ice Machine	103.9	78.3	76.4	89.7	0.75	0.74	0.86
Infrared Shelf	318.5	191.9	202.4	216.1	0.60	0.64	0.68
Mise en Place	54.8	42.1	42.4	134.9	0.77	0.77	2.46
Refrigerator 1	96.7	62.0	61.4	65.6	0.64	0.63	0.68
Refrigerator 2	32.4	26.2	25.7	52.5	0.81	0.79	1.62
Refrigerator 3	26.6	26.3	25.3	52.3	0.99	0.95	1.96
Refrigerator 4	22.9	19.4	19.5	41.4	0.85	0.85	1.81
Salamander 1	1322.8	932.7	908.4	919.2	0.71	0.69	0.69
Salamander 2	1038.5	606.0	645.4	672	0.58	0.62	0.65
Total RMSE	-	**2960.4**	**2971.2**	**3485.2**	-	-	-
Average SDNRMSE	-	-	-	-	**0.71**	**0.70**	**1.16**

Dual Fryer, with a reduction of around 10% in error in relation to CatBoost. On the other hand, Catboost presented the biggest improvement in performance for the Freezer, with a reduction also of around 10% of the error in relation to XGBoost. Finally, the LSTM model presents a poor performance for the majority of the appliances when compared with the two previous methods. The only exception here is the Salamander 1, whose performance is comparable with that of the XGBoost. These conclusions are supported by the Total RMSE row, which shows a difference of less than 11 W between the boosting algorithms and over 500 W for the LSTM.

The NRMSE enables a direct comparison between all the different appliances and models. As expected, the best-performing appliances are the dual fryer, freezer, and salamander 2, all performing fairly well in the three models. On the other hand, the least performing appliances are three out of four refrigerators (2, 3, and 4) and the mise en place, which has a similar working principle to a refrigerator and is used to keep a workbench at a low temperature. Noticeably, all these appliances are among the worst in the LSTM models.

5.2.2. Hour-ahead

The results for the hour-ahead forecasts are presented in Table 6, showing a similar pattern to the day-ahead results. In other words, there is no significant difference in the performance of the boosting models, and the LSTM still performs considerably poorer than the alternatives, even though the difference in performance is now slightly lower. Again, these conclusions are supported by the Total RMSE row, which shows a difference of 20 W between the boosting algorithms and around 500 W for the LSTM.

When comparing the performance of the different appliances through the NRMSE scores, it can be observed that the best-performing are the same as before (salamander 2, dual fryer, freezer, and the infrared shelf) with the addition of salamander 1 and the blast chiller. In contrast, the refrigerators (all four) and the mise en place are among the worst-performing appliances. Finally, looking at the NRMSE, it is also possible to observe that the overall error in the hour-ahead models is considerably lower than that of the day-ahead models.

5.2.3. Training times

When forecasting models are used, it is important to consider the respective computation times and find a balance between low error and reduced training time. This is especially relevant for short-term forecasts where the training time is relevant for the real-world applicability of the forecasting method.

Thus, to select which model to use for assessing the post-processing methodology, the average training times for each model in both day-ahead and hour-ahead were calculated and compared. The obtained results, on average, are shown in Table 7, where it is possible to conclude that the XGBoost model averages a lower training time in every situation, hence providing the best trade-off between forecasting accuracy and training time. This way, the post-processing methodologies were analyzed for the results obtained with XGBoost. As a final note, it is important to clarify that these results were obtained by training the models using only the Central Processing Unit (CPU). This can be especially relevant for the case of the LSTM, which can potentially achieve lower training times using Graphics Processing Units (GPUs).

5.3. Post-processing results

This section presents the results of the post-processing techniques. The best weights obtained for each device in the WP technique are provided in Appendix B.

Table 6
Forecasting results in hour-ahead RMSE, and SDNRMSE.

Device	Standard Deviation (W)	RMSE (W)			SDNRMSE		
		XGBoost	CatBoost	LSTM	XGBoost	CatBoost	LSTM
Blast Chiller	300.9	116.0	118.9	148	0.39	0.40	0.49
Convection Oven 1	497.0	309.5	311.6	350	0.62	0.63	0.70
Convection Oven 2	587.9	308.7	313.8	326.6	0.53	0.53	0.56
Dish Washer	478.1	253.4	259.6	272.8	0.53	0.54	0.57
Dual Fryer	167.7	65.7	69	76.7	0.39	0.41	0.46
Freezer	119.9	44.5	42.8	340.1	0.37	0.36	2.84
Ice Machine	130.4	78.0	77.5	90.5	0.60	0.59	0.69
Infrared Shelf	332.1	107.5	104.9	120.9	0.32	0.32	0.36
Mise en Place	125.7	98.2	97.5	105.1	0.78	0.78	0.84
Refrigerator 1	159.9	127.9	127.3	143.6	0.80	0.80	0.90
Refrigerator 2	70.7	44.5	44.7	49.8	0.63	0.63	0.70
Refrigerator 3	38.1	18.5	18.2	19.1	0.49	0.48	0.50
Refrigerator 4	60.6	41.6	42.1	43.7	0.69	0.69	0.72
Salamander 1	1355.8	308.5	299.4	338.3	0.23	0.22	0.25
Salamander 2	1108.3	215.3	230.6	249.2	0.19	0.21	0.22
Total RMSE	-	**2137.8**	**2157.9**	**2674.4**	-	-	-
Average SDNRMSE	-	-	-	-	**0.50**	**0.51**	**0.72**

Table 7
Average training times [s].

Analysis	XGBoost (StdDev)	CatBoost (StdDev)	LSTM (StdDev)
Day-ahead	**22.36 (16.43)**	49.80 (35.54))	49.94 (25.64)
Hour-ahead	**44.36 (33.57)**	52.94 (28.80)	288.98 (227.18))

Table 8
Forecasting error (RMSE) and Improvements to the original forecasts (%) - day-ahead analysis.

Device	Forecast Error (RMSE)					Improvement to Original Forecast			
	Original	Average	Persistence	Most Recent	Weighted	Average	Persistence	Most Recent	Weighted
Blast Chiller	193.6	190.7	150.5	146.4	131.6	1.5%	22.2%	24.4%	32.0%
Convection Oven 1	247.1	245.5	179.3	167.5	163.8	0.6%	27.5%	32.2%	33.7%
Convection Oven 2	288.9	284.4	223.4	193.8	190.7	1.6%	22.7%	32.9%	34.0%
Dish Washer	160.6	157.6	120.8	112.7	111.9	1.8%	24.7%	29.8%	30.3%
Dual Fryer	31.3	30.6	27.9	24.5	22.5	2.1%	11.0%	21.7%	28.0%
Freezer	54.1	53.2	36.8	34.7	32.0	1.6%	31.9%	35.7%	40.7%
Ice Machine	78.3	76.2	57.2	51.7	44.4	2.6%	27.0%	33.9%	43.3%
Infrared Shelf	191.9	187.2	133.3	119.5	97.5	2.5%	30.5%	37.7%	49.2%
Mise en Place	42.1	41.2	33.7	30.4	27.7	2.2%	20.1%	27.8%	34.1%
Refrigerator 1	62.0	61.0	55.9	47.7	41.0	1.7%	9.8%	23.2%	34.0%
Refrigerator 2	26.2	25.6	20.2	16.3	15.3	2.5%	22.9%	37.8%	41.5%
Refrigerator 3	26.3	25.8	22.8	19.6	18.5	1.9%	13.5%	25.5%	29.7%
Refrigerator 4	19.4	19.1	18.4	16.6	14.8	1.6%	5.4%	14.7%	23.7%
Salamander 1	932.7	920.4	634.2	611.0	562.4	1.3%	32.0%	34.5%	39.7%
Salamander 2	606.0	601.6	501.8	484.5	466.0	0.7%	17.2%	20.1%	23.1%
Total / Average	**2960.4**	**2920.0**	**2216.0**	**2076.8**	1940.3	**1.4%**	**25.1%**	**29.8%**	**34.5%**

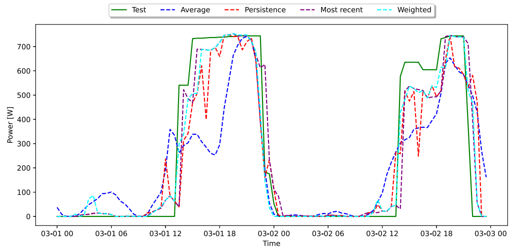
5.3.1. *Day-ahead*

The results for the day ahead forecasting are presented in Table 8, indicating clear improvements in performance in any of the three post-processing methodologies. Overall, the weighted methodology provides the best results, with an average improvement of 34.5% compared to the original prediction. This is followed by the methodology based on the most recent prediction (29.8% improvement), and finally, the persistence method with an improvement of 25.1%.

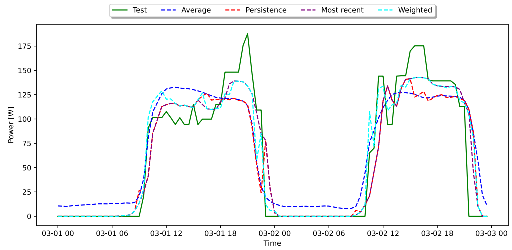
Interestingly, all three methods substantially increased the performance of those appliances that already had the best results in the original forecasts. This is the case of the infrared shelf, which saw an increase in performance of 49.2%, and the freezer, with an improvement of 40.7%, using the weighted methodology. Moreover, all the proposed methods also improved the accuracy of appliances that were originally underperforming, which was the case of the refrigerators and the ice machine.

To further illustrate the effects of the proposed post-processing methodologies, Fig. 6 depicts the results obtained for the following six appliances between 2019-03-01 and 2019-03-02: infrared shelf (Fig. 6(a)), dual fryer (Fig. 6(b)), ice machine (Fig. 6(c)), salamander 1 (Fig. 6(d)), refrigerator 2 (Fig. 6(e)), and refrigerator 3 (Fig. 6(f)). These graphs present the target variable (Test) and the update of the prediction window at each time instant of the respective methods (persistence, most recent, and weighted). Finally, the average value of the original predictions is also provided to establish a baseline (average).

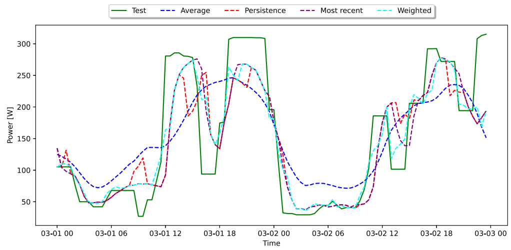
As can be observed in all six devices, and especially in the ice machine and salamander 1, the main limitation of the baseline forecasts (average) is the inability to predict peak demand. This effect is especially visible in the case of the ice machine and the salamander 1, which partly explains the inferior performance reported in Table 5 for these two devices. Overall, this limitation is



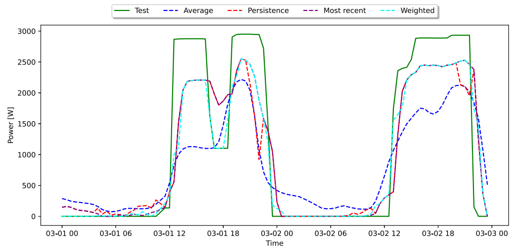
(a) Infrared Self



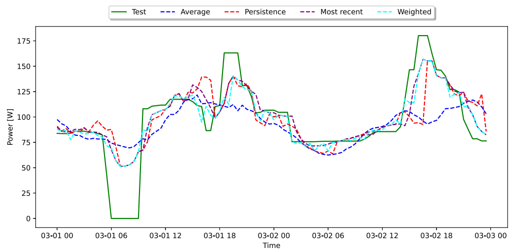
(b) Dual Fryer



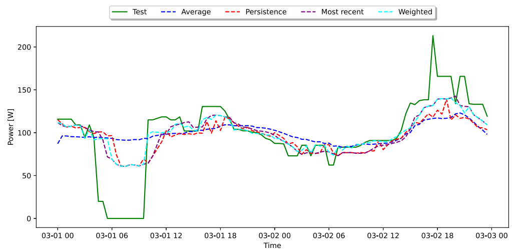
(c) Ice Machine



(d) Salamander 1



(e) Refrigerator 2



(f) Refrigerator 3

Fig. 6. Results for day-ahead analysis between 2019-03-01 and 2019-03-02.

mitigated by any of the post-processing strategies, which explains the considerable increases in performance obtained independently of the method. Nevertheless, a closer observation reveals that the weighted method can follow the start and end of the consumption cycles more closely. This is particularly visible in Salamander 1, where the weighted strategy is the only one that predicts the drop in consumption around 6 PM (18:00 on the x-axis) on day 1.

From Fig. 6(b), it is also possible to observe why the dual fryer performs well even in the baseline case. However, it can also be seen that the baseline fails to predict the periods with lower or no consumption; a pattern can be easily observable in the cases of the ice machine (Fig. 6(c)) and Salamander 1 (Fig. 6(d)) as well. In this regard, it is also clear from the graphs that the three post-processing strategies can mitigate this issue with very close results.

Regarding the two refrigerators, the first observation concerns the period between 6 AM and 9 AM on day 1, where the readings from refrigerators 2 and 3 are zero (we do not have an explanation for this zero consumption, but it is likely an issue with the breaker box in a specific area of the kitchen since this also happened with refrigerator four and the mise en place). Nevertheless, as expected, the forecasting accuracy of this period is very low, which ultimately explains the lower performance reported in Table 5 for these devices when compared to refrigerator 1. Furthermore, in line with the other devices, it is possible to see that the baseline cannot forecast the peak in consumption that occurred around 6 PM in refrigerator 2 (both days) and refrigerator 3 (second day).

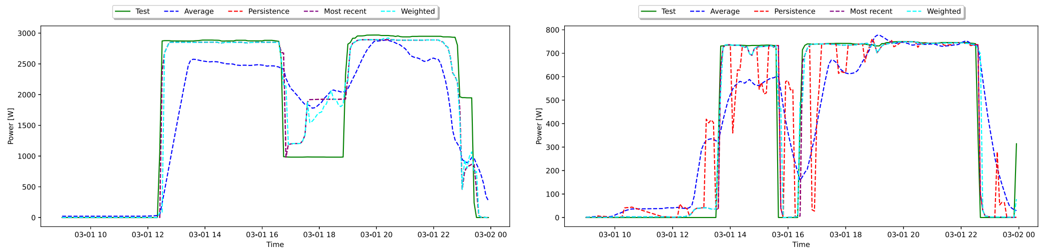
Finally, when compared to the other devices, in the case of the refrigerators, the differences between the three post-processing methods are not so evident, even though Table 8 shows better results for the weighted methodology. However, based on the other appliances, it is very likely that the differences occur due to the weighted methodology's better ability to follow the changes in power consumption more closely.

5.3.2. Hour-ahead

The results obtained applying the post-processing techniques to the hour-ahead forecasts are presented in Table 9, indicating an even higher improvement in the forecasting accuracy using any of the three methods. However, unlike the day-ahead case, there

Table 9
Forecasting error (RMSE) and Improvements to the original forecasts (%) - hour-ahead analysis.

Device	Forecast Error					Improvement to Original Forecast			
	Original	Average	Persistence	Most Recent	Weighted	Average	Persistence	Most Recent	Weighted
Blast Chiller	116.0	111.5	106.2	42.5	43.4	3.9%	8.5%	63.3%	62.6%
Convection Oven 1	309.5	300.3	212.2	160.8	165.4	3.0%	31.5%	48.1%	46.6%
Convection Oven 2	308.7	300.1	172.8	168.0	166.8	2.8%	44.0%	45.6%	46.0%
Dish Washer	253.4	247.5	174.0	174.0	171.9	2.3%	31.3%	31.3%	32.2%
Dual Fryer	65.7	63.4	44.1	44.1	42.7	3.5%	32.9%	32.9%	34.9%
Freezer	44.5	41.6	41.3	31.7	28.7	6.5%	7.3%	28.8%	35.4%
Ice Machine	78.0	73.5	54.9	32.5	30.9	5.8%	29.6%	58.4%	60.4%
Infrared Shelf	107.5	102.4	80.5	33.4	34.3	4.8%	25.1%	69.0%	68.1%
Mise en Place	98.2	94.7	93.4	82.3	86.0	3.5%	4.9%	16.2%	12.4%
Refrigerator 1	127.9	124.1	65.4	65.4	70.3	3.0%	48.9%	48.9%	45.0%
Refrigerator 2	44.5	43.1	34.0	33.4	30.9	3.1%	23.5%	25.0%	30.6%
Refrigerator 3	18.5	18.1	13.4	13.4	13.4	2.4%	27.3%	27.3%	27.4%
Refrigerator 4	41.6	40.5	36.7	30.0	28.7	2.7%	11.8%	28.0%	31.1%
Salamander 1	308.5	295.5	91.5	91.5	104.0	4.2%	70.3%	70.3%	66.3%
Salamander 2	215.3	208.6	51.0	51.0	60.8	3.1%	76.3%	76.3%	71.8%
Total / Average	**2137.8**	**2064.9**	**1271.3**	**1053.8**	1078.2	**3.4%**	**40.5%**	**50.7%**	**49.6%**



(a) Salamander (2019-03-02 09:00 - 23:59.) (b) Infrared Shelf (2019-03-02 09:00 - 23:59.)

Fig. 7. Results of the hour-ahead analysis for two resistive devices (Salamander and Infrared Shelf) between 09:00 and 23:59 of 2019-03-01.

is no clear winner between the last-prediction and weighted-prediction methodologies for the hour-ahead case. More precisely, the last prediction methodology shows an average improvement of 50.7%, against 49.6% in the weighted methodology. As for the persistence method, the average performance improvement is 40.5%.

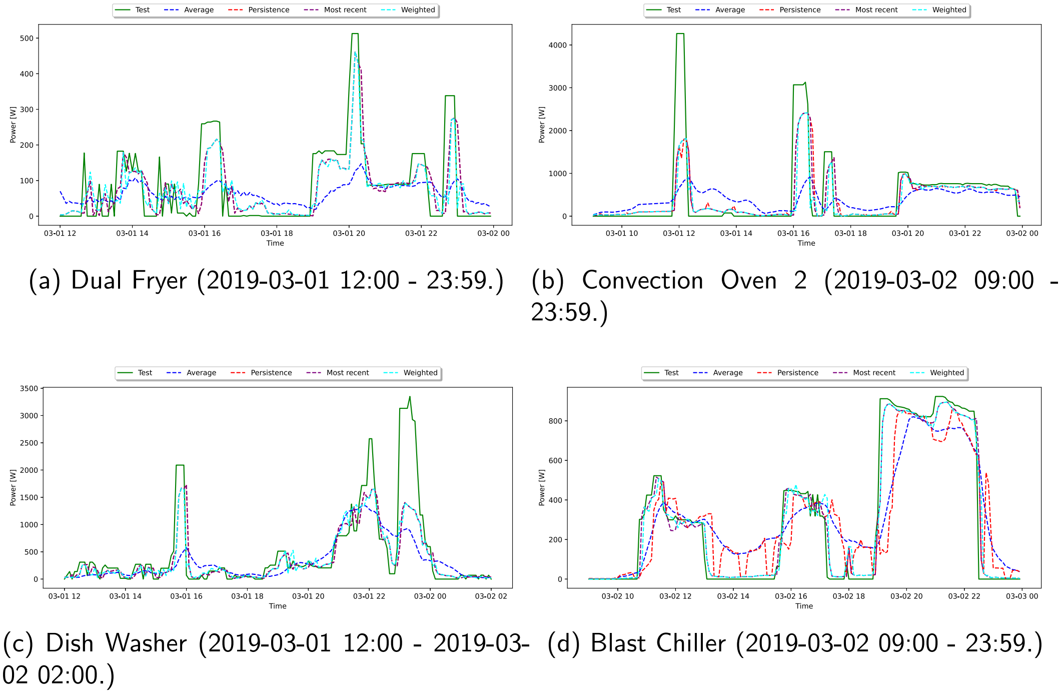
Like with the day-ahead forecasts, the proposed methodologies significantly improve the performance of certain appliances (≥63%), including the blast chiller, the infrared shelf, and the two salamanders. All these appliances have already been shown to perform best in the original methodology. To further understand the effects of the post-processing methodologies in the hour-ahead forecasting results, Figs. 7 to 9 depicts the results obtained for the following ten devices: salamander (Fig. 7(a)), infrared shelf (Fig. 7(b)), dual fryer (Fig. 8(a)), convection oven 2 (Fig. 8(b)), dishwasher (Fig. 8(c)), blast chiller (Fig. 8(d)), refrigerator 1 (Fig. 9(a)), mise en place (Fig. 9(b)), ice machine (Fig. 9(c)), and freezer (Fig. 9(d)).

Regarding the Salamander 1 and the infrared shelf, the first observation relates to the very good ability to forecast maximum power demand, which greatly contrasts the results obtained for the day-ahead forecasting in these two appliances. It is also interesting to observe the very good ability of the post-processing approaches to predict the periods with sharper increases and decreases in power demand for Salamander 1. Nevertheless, in the infrared shelf, the persistence method is not so effective as it struggles to stabilize the forecasted values. This significantly affects the effectiveness of this method for this particular appliance, with an improvement of only 25.1% when compared to the original forecast, as shown in Table 9.

Concerning the dual fryer, the convection oven 2, and the dishwasher, it is possible to see that there is virtually no difference between the three post-processing methods (as expected from the results presented in Table 9). Overall, the three approaches manage to approximate the power consumption profile despite struggling to forecast the periods with higher demand. In contrast, in the blast chiller, the persistence is significantly outperformed by the two other methods, particularly when forecasting low or no demand.

Concerning refrigerator 1, it can be seen that all the approaches struggle to forecast longer activations, which are less frequent in the data. These activations are seen in the figure between 3 PM and 4 PM and 6 PM to 7 PM. In the remaining periods, the three post-processing methods have very similar performances. Still, the most interesting observation is the fact that the persistence and most recent methods achieved the same performance, indicating that for this appliance, the best prediction for a given timestamp is always the most recent. This may be expected given the working mode of this device, which is somehow stable throughout the day.

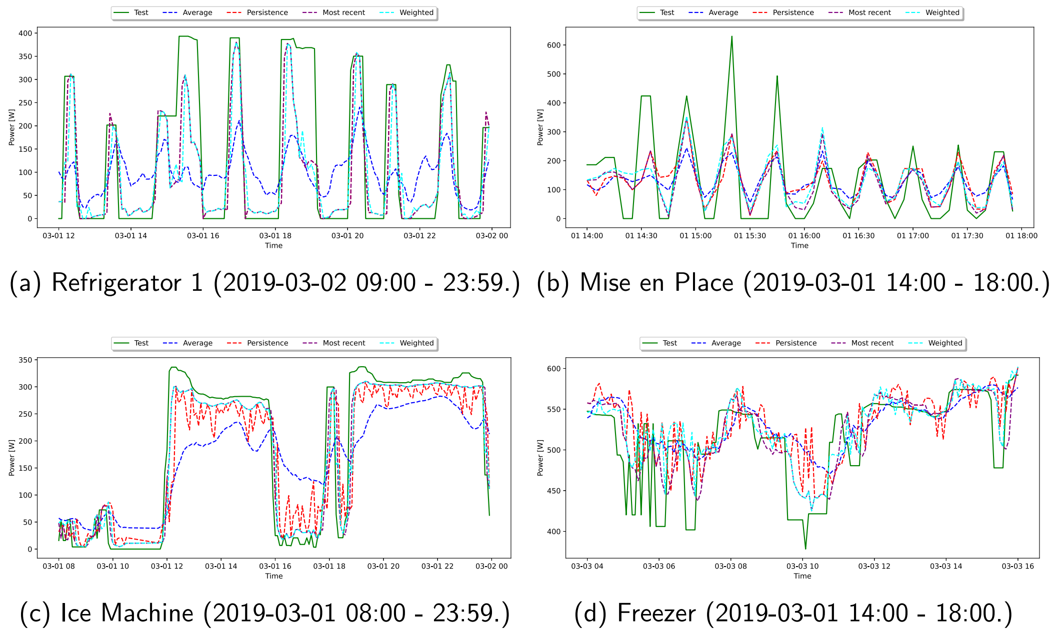
Concerning the mise en place, it is evident from the graphs that all the methods have difficulties predicting the peak consumption despite being able to keep track of the power demand profile. One important consideration to make about this device is the fact that it is the one that benefits the least from the post-processing methodologies, with only a 16% improvement in the best case. A possible



(a) Dual Fryer (2019-03-01 12:00 - 23:59.) (b) Convection Oven 2 (2019-03-02 09:00 - 23:59.)

(c) Dish Washer (2019-03-01 12:00 - 2019-03-02 02:00.) (d) Blast Chiller (2019-03-02 09:00 - 23:59.)

Fig. 8. Results of the hour-ahead analysis for four cyclic devices (dual fryer, convection oven, dishwasher, and blast chiller).



(a) Refrigerator 1 (2019-03-02 09:00 - 23:59.) (b) Mise en Place (2019-03-01 14:00 - 18:00.)

(c) Ice Machine (2019-03-01 08:00 - 23:59.) (d) Freezer (2019-03-01 14:00 - 18:00.)

Fig. 9. Results of the hour-ahead analysis for four cooling devices (refrigerator 1, mise en place, ice machine, and freezer).

explanation for this is the short duration of the activations, often lasting less than the 5-minute period selected for downsampling the data, resulting in considerable spikes when in the downsampled signal.

Regarding the ice machine (Fig. 9(c)), it is possible to observe significant improvements when using the last-prediction and weighted prediction strategies, particularly when forecasting peak demand. Concerning the persistence method, it can still predict the consumption profile with a good performance. However, it cannot return stabilized predictions, much in line with what happens with the infrared shelf.

Finally, with respect to the freezer (Fig. 9(d)), the first observation is the fact that during the period of six hours, the consumption never reached zero, indicating that the desired temperature was never reached or if it was reached, it was just for a few seconds. This greatly contrasts the refrigerator, which has a very similar working principle. This behavior can be explained in different ways. Firstly, it could be due to the equipment needing maintenance, for example, a malfunctioning thermostat. Alternatively, it is possible that the freezer is being used beyond its capacity, resulting in higher thermal inertia. This creates a situation where the freezer never reaches the target temperature, even if it operates continuously.

As for the forecasting results, there is very little difference between the last and the weighted predictions, with both methods struggling to forecast the periods when the consumption is lower. Concerning the persistence-based method, the improvements are marginal (7.3% compared to 28.8% and 35.4% in the most recent and weighted methods, respectively.)

6. Conclusion

This work aimed to develop and assess methodologies for forecasting device-level active power consumption within IKs. The analysis encompasses both hour-ahead and day-ahead scenarios to capture diverse consumption patterns. The day-ahead analysis offers insights into daily load patterns, while the hour-ahead analysis delves into individual load behavior. A two-step approach was proposed: initially benchmarking three forecasting algorithms (XGBoost, CatBoost, and LSTM), followed by post-processing the results of the best-performing algorithm using three distinct methods (PbP, MRP, and WP).

The proposed methods were implemented on power consumption data from 15 industrial kitchen devices sourced from the FIKElectricity dataset. The benchmarking results revealed that both boosting algorithms performed similarly well, outperforming LSTM models significantly. Due to the superior trade-off between performance and training time, the results from the XGBoost algorithm were selected for subsequent post-processing steps.

Comparing the results obtained with post-processing methods in both day-ahead and real-time scenarios, it is evident that the weight-based method exhibits the best performance. This can be attributed to the efficiency of the Roth–Erev method in selecting the function with the best performance for each time horizon and each appliance. It is worth mentioning that each appliance operates differently, which creates challenges in using a single method, as has traditionally been the case. Upon closer examination of the values, it is also apparent that the improvements are more significant in day-ahead forecasting. This is primarily due to the higher correlation between consecutive hours, which is normally captured by the latest forecast.

6.1. Research implications

Providing accurate appliance consumption forecasts is an important step toward the energy transition. However, most of the existing research focuses on aggregated load forecasts, especially in the residential sector. Instead, this work positions itself in the IK sector, where, to the best of our knowledge, there is only one published research on forecasting electric device consumption [18].

To the best of our knowledge, this is also one of the first works that considers the possibility of applying post-processing methods to improve the performance of forecasting algorithms applied to individual device consumption data. In this regard, the obtained results clearly show the importance of post-processing by re-using the outcomes of previous forecasts. The benefits of the post-processing methods are particularly visible when it comes to improving the performance of the periods with higher consumption, which is where most methods tend to produce higher errors, as stated by other authors, e.g., [15]. For example, in the hour-ahead analysis, it was possible to increase the performance of the two salamanders by at least 66%, and the infrared shelf by 69%.

Moreover, by considering several appliances of the same kind (e.g., four refrigerators), this work also reveals that although some appliances share the same working principles, the forecasting algorithms' performance (and post-processing methods) highly depends on how the appliance is used. Still, in this regard, the obtained results also show that selecting the best forecasting algorithms and post-processing does not depend only on the forecasting horizon and the amount of historical data. Instead, the working principles and type of usage should be considered. An example of this aspect is the persistence-based method, which, despite being a widely used baseline in aggregated load forecasting and having shown a good performance when used as a post-processing method, can significantly under-perform when applied to the wrong appliances. In this regard, a key advantage of the proposed method is that it can be applied directly to any forecasting model and prediction horizon since it depends only on the historical prediction and the actual values. Furthermore, the proposed post-processing methods do not require training additional ML models, making them dataset-independent.

6.2. Limitations and future work

While the original objectives were successfully met, certain limitations should be addressed in future iterations of this work. First and foremost, the main limitation of this work is the short duration of the dataset, which, despite comprising 15 IK devices, contains only about four weeks of data. Hence, future work should attempt to replicate the proposed methodology in a longer dataset. Based on the results obtained in this paper, it is likely that the performance of the forecasting algorithms would increase (particularly in the case of the LSTM models), which could also affect the gains obtained in the post-processing step. Hence, future work should also strive to explore more advanced LSTM- and CNN-based forecasting models, e.g., [52,53]

Besides the duration of the data, the obtained results strongly suggest that the accuracy of the proposed methods depends not only on the working principles of devices but also on how they are used in daily operations. Consequently, evaluating the proposed methodologies using data from other IKs would be interesting. Moreover, given the recent focus on forecasting the consumption of household appliances, it would also be interesting to replicate the proposed methods in the widely available residential datasets [54,55].

Moreover, even though this paper already looked at two different forecasting horizons, assessing the generalizability of the proposed methods in other forecasting horizons and sampling rates would be relevant. For example, it would be interesting to forecast the demand for periods under one hour using higher sampling rates (e.g., forecast for the next 15 min at 1/60 Hz are very relevant in the context of small consumer electricity markets [7]).

Concerning the post-processing methodologies, future work should also address the development of more advanced methodologies. For example, based on the hour-ahead results, it is very likely that an extended version of the WP that considers the order of the predictions to establish the weights, i.e., giving more weight to the most recent ones, would have superior performance. In this regard, despite exhibiting superior performance, a limitation of the weight-based method is the fact that it requires fine-tuning the weighting parameter. Therefore, future work must also explore mechanisms to assist the user in setting the optimal eighth factor.

A potential drawback of post-processing approaches purely based on the prediction values is the fact that they are contingent upon the initial quality of the models; if the models exhibit poor performance by default, post-processing efficacy may be limited. Additionally, as models degrade over time due to unforeseen variations in data distribution, the effectiveness of post-processing techniques may diminish accordingly. Hence, future research should delve into the effects of post-processing across a spectrum of scenarios, encompassing very poor models, highly proficient models, and models with fluctuating performance.

Finally, a broader vision entails integrating post-processing elements into forecasting algorithms. This could involve modifying the objective function to enable the model to discern and penalize errors based on their temporal position within the forecasting horizon.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data is publicly available. A link to the dataset is provided in the paper.

Acknowledgments

Lucas Pereira received funding from the Portuguese Foundation for Science and Technology (FCT) under grants CEECIND/01179/2017 and UIDB/50009/2020. Hugo Morais received funding from FCT, Portugal under grant UIDB/50021/2020. This research was funded by FCT, Portugal under the Exploring the Human-Water-Energy Nexus in Industrial Kitchens (nexIK) project (EXPL/CCI-COM/1234/2021) through national funds.

Appendix A. Hyperparameter optimization results

This appendix contains the results obtained in the hyperparameter optimization, using the *Optuna* environment (see Tables A.10–A.15).

Table A.10
Optimal hyperparameter values for XGBoost — hour-ahead analysis.

Device	Subsample	Column sample by Tree	Tree depth	Learning rate	λ	α
Blast Chiller	0.8	0.2	2	0.005	1	1
Convection Oven 1	0.8	0.7	3	0.01	1	10
Convection Oven 2	0.8	0.8	4	0.02	50	5
Dish Washer	0.4	0.5	2	0.01	10	5
Dual Fryer	0.4	0.5	2	0.01	50	10
Freezer	0.8	0.2	2	0.005	10	50
Ice Machine	0.2	0.2	2	0.005	50	5
Infrared Shelf	0.8	0.7	3	0.005	50	10
Mise en Place	0.2	0.6	2	0.005	1	1
Refrigerator 1	0.2	0.2	4	0.01	50	5
Refrigerator 2	0.8	0.5	4	0.02	1	5
Refrigerator 3	0.8	0.2	3	0.01	1	5
Refrigerator 4	0.4	0.7	3	0.02	1	10
Salamander 1	0.2	0.4	2	0.02	50	5
Salamander 2	0.2	0.2	4	0.01	10	10

Table A.11
Optimal hyperparameter values for XGBoost — hour-ahead analysis.

Device	Subsample	Column sample by Tree	Tree depth	Learning rate	λ	α
Blast Chiller	0.6	0.7	4	0.01	10	10
Convection Oven 1	0.8	0.8	3	0.02	50	10
Convection Oven 2	0.2	0.6	3	0.005	50	5
Dish Washer	0.2	0.7	3	0.01	50	1
Dual Fryer	0.2	0.8	4	0.02	5	10
Freezer	0.7	0.4	4	0.005	5	5
Ice Machine	0.2	0.8	4	0.01	50	1
Infrared Shelf	0.5	0.8	3	0.01	10	10
Mise en Place	0.6	0.7	4	0.02	50	1
Refrigerator 1	0.8	0.7	4	0.01	50	50
Refrigerator 2	0.4	0.5	3	0.005	50	5
Refrigerator 3	0.7	0.6	3	0.01	1	10
Refrigerator 4	0.6	0.6	4	0.01	50	10
Salamander 1	0.5	0.8	3	0.01	5	50
Salamander 2	0.2	0.8	3	0.005	1	5

Table A.12
Optimal hyperparameter values for CatBoost — day-ahead analysis.

Device	Subsample	Tree depth	Learning rate	λ
Blast Chiller	0.7	5	0.02	1
Convection Oven 1	0.7	5	0.01	50
Convection Oven 2	0.5	3	0.01	10
Dish Washer	0.6	3	0.02	50
Dual Fryer	0.2	3	0.02	50
Freezer	0.6	3	0.02	5
Ice Machine	0.5	5	0.02	5
Infrared Shelf	0.5	5	0.02	5
Mise en Place	0.5	3	0.005	1
Refrigerator 1	0.6	5	0.01	5
Refrigerator 2	0.2	5	0.005	5
Refrigerator 3	0.7	5	0.02	5
Refrigerator 4	0.4	3	0.01	1
Salamander 1	0.4	5	0.02	10
Salamander 2	0.8	3	0.005	50

Appendix B. Best weights for the weighted based prediction

This appendix contains the values of the weighting parameter used in the weighted-based prediction post-processing methodology (see Table B.16).

Table A.13
Optimal Hyperparameter values for CatBoost — hour-ahead analysis.

Device	Subsample	Tree depth	Learning rate	λ
Blast Chiller	0.6	5	0.02	10
Convection Oven 1	0.6	4	0.01	50
Convection Oven 2	0.6	4	0.01	5
Dish Washer	0.8	5	0.02	5
Dual Fryer	0.2	4	0.02	50
Freezer	0.8	4	0.02	10
Ice Machine	0.4	3	0.02	5
Infrared Shelf	0.8	4	0.02	1
Mise en Place	0.5	5	0.02	50
Refrigerator 1	0.6	4	0.02	1
Refrigerator 2	0.6	4	0.01	1
Refrigerator 3	0.7	4	0.01	50
Refrigerator 4	0.8	4	0.005	5
Salamander 1	0.6	4	0.01	1
Salamander 2	0.8	4	0.02	10

Table A.14
Optimal hyperparameter values for LSTM — day-ahead analysis.

Device	Layers	Units	Activation function	Dropout rate
Blast Chiller	1	64	linear	0.2
Convection Oven 1	1	32	linear	0
Convection Oven 2	1	16	linear	0.2
Dish Washer	2	64	tanh	0.2
Dual Fryer	1	16	tanh	0.1
Freezer	2	64	linear	0.1
Ice Machine	2	16	tanh	0.1
Infrared Shelf	3	64	tanh	0
Mise en Place	1	64	tanh	0.2
Refrigerator 1	3	16	tanh	0.2
Refrigerator 2	2	64	linear	0.2
Refrigerator 3	1	16	tanh	0
Refrigerator 4	1	16	linear	0.1
Salamander 1	2	64	linear	0.2
Salamander 2	3	32	tanh	0.2

Table A.15
Optimal hyperparameter values for LSTM — hour-ahead analysis.

Device	Layers	Units	Activation function	Dropout rate
Blast Chiller	1	32	tanh	0.1
Convection Oven 1	2	32	linear	0
Convection Oven 2	1	32	linear	0
Dish Washer	1	16	tanh	0.2
Dual Fryer	3	16	tanh	0.1
Freezer	2	64	tanh	0.1
Ice Machine	2	16	linear	0.1
Infrared Shelf	3	64	tanh	0.2
Mise en Place	1	16	tanh	0.2
Refrigerator 1	2	16	tanh	0.2
Refrigerator 2	2	16	linear	0.2
Refrigerator 3	2	16	tanh	0.1
Refrigerator 4	1	16	tanh	0.1
Salamander 1	3	32	linear	0.1
Salamander 2	3	16	linear	0.2

Table B.16
Optimal Weight factor for the weighted based prediction — day and hour-ahead scenarios.

Device	Weight factor (W)	
	Day-ahead	Hour-ahead
Blast Chiller	0.95	0.75
Convection Oven 1	0.90	0.95
Convection Oven 2	0.95	0.90
Dish Washer	0.95	0.75
Dual Fryer	0.95	0.95
Freezer	0.95	0.95
Ice Machine	0.95	0.95
Infrared Shelf	0.95	0.05
Mise en Place	0.90	0.05
Refrigerator 1	0.90	0.95
Refrigerator 2	0.95	0.25
Refrigerator 3	0.95	0.50
Refrigerator 4	0.90	0.10
Salamander 1	0.95	0.95
Salamander 2	0.95	0.95

References

[1] Economidou M, Todeschi V, Bertoldi P, D'Agostino D, Zangheri P, Castellazzi L. Review of 50 years of EU energy efficiency policies for buildings. Energy Build 2020;225:110322. http://dx.doi.org/10.1016/j.enbuild.2020.110322.
[2] Mudie S, Essah E, Grandison A, Felgate R. Electricity use in the commercial kitchen. Int J Low-Carbon Technol 2013;11(1):66–74. http://dx.doi.org/10.1093/ijlct/ctt068.
[3] Griffin M, Ramsson T, Gibson G. International energy agency - technology brief R06 – june 2012 - www.etsap. 2023, URL https://iea-etsap.org/E-TechDS/PDF/R06_Cooking_FINAL_GSOK.pdf. [Accessed 05 June 2023].
[4] Ana Oliveira, Miriam Ribeiro, Ricardo Rodrigues, Gonçalo de Morais, Hugo Morais, Lucas Pereira. On the role of industrial kitchens in sustainable energy systems: The nexIK Vision. In: CIRED 2023 - The 27th international conference and exhibition on electricity distribution. 2023.
[5] Higgins-Desbiolles F, Moskwa E, Wijesinghe G. How sustainable is sustainable hospitality research? A review of sustainable restaurant literature from 1991 to 2015. Curr Issues Tour 2019;22(13):1551–80. http://dx.doi.org/10.1080/13683500.2017.1383368.
[6] Legrand W, Sloan P, Simons-Kaufmann C, Fleischer S. A review of restaurant sustainable indicators. In: S. Chen J, editor. In: Advances in hospitality and leisure, vol. 6, Emerald Group Publishing Limited; 2010, p. 167–83. http://dx.doi.org/10.1108/S1745-3542(2010)0000006013.
[7] Pedro A, Krutnik M, Yadack VM, Pereira L, Morais H. Opportunities and challenges for small-scale flexibility in European electricity markets. Util Policy 2023;80:101477. http://dx.doi.org/10.1016/j.jup.2022.101477, URL https://www.sciencedirect.com/science/article/pii/S0957178722001412.
[8] Nonaka T, Shimmura T, Fujii N, Mizuyama H. Energy consumption in the food service industry: A conceptual model of energy management considering service properties. In: Umeda S, Nakano M, Mizuyama H, Hibino H, Kiritsis D, von Cieminski G, editors. Advances in production management systems: Innovative production management towards sustainable growth. Cham: Springer International Publishing; 2015, p. 605–11.
[9] Khuntia SR, Rueda JL, Meijden MA. Forecasting the load of electrical power systems in mid- and long-term horizons: A review. IET Gener Transm Distrib 2016;10(16):3971–7. http://dx.doi.org/10.1049/iet-gtd.2016.0340.
[10] Kuster C, Rezgui Y, Mourshed M. Electrical load forecasting models: A critical systematic review. Sustainable Cities Soc 2017;35:257–70. http://dx.doi.org/10.1016/j.scs.2017.08.009.
[11] Yildiz B, Bilbao JI, Dore J, Sproul AB. Recent advances in the analysis of residential electricity consumption and applications of smart meter data. Appl Energy 2017;208:402–27. http://dx.doi.org/10.1016/j.apenergy.2017.10.014.
[12] Haben S, Arora S, Giasemidis G, Voss M, Vukadinović Greetham D. Review of low voltage load forecasting: Methods, applications, and recommendations. Appl Energy 2021;304:117798. http://dx.doi.org/10.1016/j.apenergy.2021.117798.
[13] Hong T, Fan S. Probabilistic electric load forecasting: A tutorial review. Int J Forecast 2016;32(3):914–38. http://dx.doi.org/10.1016/j.ijforecast.2015.11.011.
[14] Faustine A, Pereira L. FPSeq2Q: Fully parameterized sequence to quantile regression for net-load forecasting with uncertainty estimates. IEEE Trans Smart Grid 2022;1. http://dx.doi.org/10.1109/TSG.2022.3148699.
[15] Amin MAA, Hoque MA. Comparison of ARIMA and SVM for short-term load forecasting. In: 2019 9th annual information technology, electromechanical engineering and microelectronics conference. 2019, p. 1–6. http://dx.doi.org/10.1109/IEMECONX.2019.8877077.
[16] Faria P, Jozi A, Vale Z. Forecasting refrigerators consumption to support their aggregated participation in demand response. In: 2020 IEEE international conference on environment and electrical engineering and 2020 IEEE industrial and commercial power systems Europe. 2020, p. 1–6. http://dx.doi.org/10.1109/EEEIC/ICPSEurope49358.2020.9160527.
[17] Ji Y, Buechler E, Rajagopal R. Data-driven load modeling and forecasting of residential appliances. IEEE Trans Smart Grid 2020;11(3):2652–61. http://dx.doi.org/10.1109/TSG.2019.2959770.
[18] Amantegui J, Morais H, Pereira L. Benchmark of electricity consumption forecasting methodologies applied to industrial kitchens. Buildings 2022;12(12). http://dx.doi.org/10.3390/buildings12122231, URL https://www.mdpi.com/2075-5309/12/12/2231.
[19] Nepal B, Yamaha M, Yokoe A, Yamaji T. Electricity load forecasting using clustering and ARIMA model for energy management in buildings. Jpn Architectural Rev. 2020;3(1):62–76. http://dx.doi.org/10.1002/2475-8876.12135, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2475-8876.12135.
[20] Barchi G, Pierro M, Moser D. Predictive energy control strategy for peak shaving and shifting using BESS and PV generation applied to the retail sector. Electronics 2019;8(5):526. http://dx.doi.org/10.3390/electronics8050526.
[21] Vinagre E, Pinto T, Ramos S, Vale Z, Corchado JM. Electrical energy consumption forecast using support vector machines. In: 2016 27th international workshop on database and expert systems applications. 2016, p. 171–5. http://dx.doi.org/10.1109/DEXA.2016.046.
[22] Luo J, Hong T, Gao Z, Fang S-C. A robust support vector regression model for electric load forecasting. Int J Forecast 2023;39(2):1005–20.
[23] Liao X, Cao N, Li M, Kang X. Research on short-term load forecasting using XGBoost based on similar days. In: 2019 International conference on intelligent transportation, big data & smart city. 2019, p. 675–8. http://dx.doi.org/10.1109/ICITBS.2019.00167.

[24] Zheng H, Yuan J, Chen L. Short-term load forecasting using EMD-LSTM neural networks with a Xgboost algorithm for feature importance evaluation. Energies 2017;10(8). http://dx.doi.org/10.3390/en10081168, URL https://www.mdpi.com/1996-1073/10/8/1168.

[25] Neupane B, Pedersen TB, Thiesson B. Utilizing device-level demand forecasting for flexibility markets - Full version. 2018, http://dx.doi.org/10.48550/ARXIV.1805.00702, URL https://arxiv.org/abs/1805.00702.

[26] Razghandi M, Turgut D. Residential appliance-level load forecasting with deep learning. In: GLOBECoM 2020 - 2020 IEEE global communications conference. 2020, p. 1–6. http://dx.doi.org/10.1109/GLOBECOM42002.2020.9348197.

[27] Razghandi M, Zhou H, Erol-Kantarci M, Turgut D. Short-Term load forecasting for smart home appliances with sequence to sequence learning. In: ICC 2021 - IEEE international conference on communications. 2021, p. 1–6. http://dx.doi.org/10.1109/ICC42927.2021.9500767.

[28] He M, Khorsand M. Residential appliance-level consumption modeling and forecasting via conditional hidden semi-Markov model. In: 2022 North American power symposium. 2022, p. 1–6. http://dx.doi.org/10.1109/NAPS56150.2022.10012191.

[29] Yang D, van der Meer D. Post-processing in solar forecasting: Ten overarching thinking tools. Renew Sustain Energy Rev 2021;140:110735. http://dx.doi.org/10.1016/j.rser.2021.110735.

[30] Hallgren C, Ivanell S, Körnich H, Vakkari V, Sahlée E. The smoother the better? A comparison of six post-processing methods to improve short-term offshore wind power forecasts in the Baltic Sea. Wind Energy Sci 2021;6(5):1205–26. http://dx.doi.org/10.5194/wes-6-1205-2021.

[31] Theocharides S, Makrides G, Livera A, Theristis M, Kaimakis P, Georghiou GE. Day-ahead photovoltaic power production forecasting methodology based on machine learning and statistical post-processing. Appl Energy 2020;268:115023. http://dx.doi.org/10.1016/j.apenergy.2020.115023.

[32] Ramos D, Faria P, Morais A, Vale Z. Using decision tree to select forecasting algorithms in distinct electricity consumption context of an office building. Energy Rep 2022;8:417–22. http://dx.doi.org/10.1016/j.egyr.2022.01.046.

[33] Zhu Z, Chen W, Xia R, Zhou T, Niu P, Peng B, et al. eForecaster: Unifying electricity forecasting with robust, flexible, and explainable machine learning algorithms. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, (13):2023, p. 15630–8. http://dx.doi.org/10.1609/aaai.v37i13.26853.

[34] Chen Tianqi GC. XGBoost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016.

[35] Yao X, Fu X, Zong C. Short-term load forecasting method based on feature preference strategy and LightGBM-XGboost. IEEE Access 2022;10:75257–68. http://dx.doi.org/10.1109/ACCESS.2022.3192011.

[36] Dorogush AV, Ershov V, Gulin A. CatBoost: gradient boosting with categorical features support. 2018, http://dx.doi.org/10.48550/ARXIV.1810.11363, URL https://arxiv.org/abs/1810.11363.

[37] Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A. CatBoost: unbiased boosting with categorical features. In: Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R, editors. In: Advances in neural information processing systems, vol. 31, Curran Associates, Inc.; 2018, URL https://proceedings.neurips.cc/paper/2018/file/14491b756b3a51daac41c24863285549-Paper.pdf.

[38] Hancock JT, Khoshgoftaar TM. CatBoost for big data: An interdisciplinary review. J Big Data 2020;7(1):94. http://dx.doi.org/10.1186/s40537-020-00369-8.

[39] Li P, Zhou K, Lu X, Yang S. A hybrid deep learning model for short-term PV power forecasting. Appl Energy 2020;259:114216. http://dx.doi.org/10.1016/j.apenergy.2019.114216.

[40] Wu K, Wu J, Feng L, Yang B, Liang R, Yang S, et al. An attention-based CNN-LSTM-BiLSTM model for short-term electric load forecasting in integrated energy system. Int Trans Electr Energy Syst 2021;31(1):e12637. http://dx.doi.org/10.1002/2050-7038.12637, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2050-7038.12637.

[41] Foster J, Liu X, McLoone S. Adaptive sliding window load forecasting. In: 2017 28th Irish signals and systems conference. 2017, p. 1–6. http://dx.doi.org/10.1109/ISSC.2017.7983638.

[42] Khan IA, Akber A, Xu Y. Sliding window regression based short-term load forecasting of a multi-area power system. In: 2019 IEEE Canadian conference of electrical and computer engineering. 2019, p. 1–5. http://dx.doi.org/10.1109/CCECE.2019.8861915.

[43] Khalid R, Javaid N. A survey on hyperparameters optimization algorithms of forecasting models in smart grid. Sustainable Cities Soc 2020;61:102275. http://dx.doi.org/10.1016/j.scs.2020.102275, URL https://www.sciencedirect.com/science/article/pii/S2210670720304960.

[44] Zahid M, Ahmed F, Javaid N, Abbasi RA, Zainab Kazmi HS, Javaid A, et al. Electricity price and load forecasting using enhanced convolutional neural network and enhanced support vector regression in smart grids. Electronics 2019;8(2). http://dx.doi.org/10.3390/electronics8020122, URL https://www.mdpi.com/2079-9292/8/2/122.

[45] Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A next-generation hyperparameter optimization framework. New York, NY, USA: Association for Computing Machinery; 2019, http://dx.doi.org/10.1145/3292500.3330701.

[46] Frazier PI. A tutorial on Bayesian optimization. 2018, arXiv:1807.02811.

[47] Jing Z, Ngan H, Wang Y, Zhang Y, Wang J. Study on the convergence property of re learning model in electricity market simulation. In: 8th international conference on advances in power system control, operation and management. 2009, p. 1–5. http://dx.doi.org/10.1049/cp.2009.1813.

[48] Nti IK, Teimeh M, Nyarko-Boateng O, Adekoya AF. Electricity load forecasting: a systematic review. J Electr Syst Inf Technol 2020;7(1):13. http://dx.doi.org/10.1186/s43067-020-00021-8.

[49] Pereira L, Aguiar V, Vasconcelos F, Martins R, Garcês T, Morais H. FIKElectricity: A electricity consumption dataset from three restaurant kitchens in Portugal. Scientific Data 2023;10(1):779. http://dx.doi.org/10.1038/s41597-023-02698-8.

[50] Park K-J, Son S-Y. A novel load image profile-based electricity load clustering methodology. IEEE Access 2019;7:59048–58. http://dx.doi.org/10.1109/ACCESS.2019.2914216.

[51] Jiang Q, Cheng Y, Le H, Li C, Liu PX. A stacking learning model based on multiple similar days for short-term load forecasting. Mathematics 2022;10(14). http://dx.doi.org/10.3390/math10142446, URL https://www.mdpi.com/2227-7390/10/14/2446.

[52] Peng L, Wang L, Xia D, Gao Q. Effective energy consumption forecasting using empirical wavelet transform and long short-term memory. Energy 2022;238:121756. http://dx.doi.org/10.1016/j.energy.2021.121756.

[53] Shaikh AK, Nazir A, Khalique N, Shah AS, Adhikari N. A new approach to seasonal energy consumption forecasting using temporal convolutional networks. Results Eng 2023;19:101296. http://dx.doi.org/10.1016/j.rineng.2023.101296.

[54] Pereira L, Nunes N. Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools—A review. Wiley Interdiscip Rev Data Min Knowl Discov 2018;8(6):e1265, URL: http://dx.doi.org/10.1002/widm.1265.

[55] Himeur Y, Alsalemi A, Bensaali F, Amira A. Building power consumption datasets: Survey, taxonomy and future directions. Energy Build 2020;227:110404. http://dx.doi.org/10.1016/j.enbuild.2020.110404.