A Novel Dual-CNN Architecture with Adaptive Persistence for Medium-Term Electricity Load Forecasting

Pascal Tribel[1][0009-0006-3701-1223], Gian Marco Paldino[1][0000-0002-8680-9403], Alejandro Morales-Hernández[1][0000-0003-0053-4902], and Gianluca Bontempi[1][0000-0001-8621-316X]

Machine Learning Group, Université Libre de Bruxelles, Belgium
{pascal.tribel,gian.marco.paldino,alejandro.morales.hernandez, gianluca.bontempi}@ulb.be

Abstract. Accurate electricity load forecasting is essential for ensuring the reliable and efficient operation of power systems. However, modeling complex temporal patterns across multiple timescales using heterogeneous data sources remains a complex and challenging task. Overcoming this challenge enables more robust forecasting models that support better grid management and decision-making in real-world operational settings. This paper presents a novel dual-CNN architecture for medium-term electricity load forecasting that leverages temporal patterns across multiple timescales. Our approach combines two parallel convolutional neural networks: one processes historical load data, and the other processes meteorological information. These are merged into a final linear layer that adaptively adjusts an optimal persistence forecast. The architecture exploits short-term patterns and long-term seasonal effects by incorporating historical data from the previous month and corresponding periods of the prior year. We demonstrate the generalizability of our approach by forecasting the total load at a 30-hour horizon in the Belgian grid system, using a dataset comprising load measurements, meteorological data, and system imbalance information. Through rigorous validation, the results evidence that our model consistently outperforms the existing operational forecasts provided by the Belgian transmission system operator ELIA and several forecasting algorithms. This research contributes to the growing body of work on electricity load forecasting by introducing a generalizable architecture that efficiently captures complex temporal dynamics while remaining computationally tractable for operational deployment.

Keywords: Electricity load forecasting · Neural networks · Time series forecasting · Feature extraction · Convolutional neural networks

1 Introduction

Accurate electricity load forecasting is essential for power system planning, operational security, and market efficiency. System operators worldwide rely on these forecasts for day-ahead scheduling, reserve allocation, and congestion management. Medium-term forecasts, particularly in the 24-48 hour horizon, are especially valuable for operational planning and market participation, but remain challenging due to the complex temporal dynamics involved. Load forecasting has been extensively studied, with approaches ranging from statistical methods to increasingly sophisticated machine learning techniques. However, the complexity of electricity consumption patterns, driven by economic activities, weather conditions, calendar effects, and behavioral factors, continues to challenge forecasting precision. Recent advances in deep learning, particularly convolutional neural networks (CNNs), offer promising approaches for capturing these complex temporal patterns without requiring extensive feature engineering.

Despite significant progress, several challenges persist, particularly relevant to operational and viable medium-term forecasting. First, while methods for parallel processing of inputs have been proposed [27,5], there is still a need for developing specialized CNN architectures explicitly designed for the distinct characteristics of load versus meteorological time series over long historical windows relevant for medium-term load forecasting (MTLF). Second, the integration of domain knowledge, specifically optimally identified persistence forecasts, with advanced DL feature extractors like CNNs remains underexplored. Third, many complex hybrid models involving multi-stage decomposition or large ensembles [34,35] may face challenges regarding computational tractability for operational deployment, where timely retraining and inference are crucial. Finally, there is a continuous need for generalizable frameworks that can be readily adapted to different power systems and forecasting horizons without requiring extensive system-specific tuning.

In this paper, we propose a novel dual-CNN architecture for medium-term electricity load forecasting, and the main contributions of this paper are:

- our architecture employs two parallel CNNs to efficiently extract relevant temporal patterns from historical load data and meteorological information,
- the proposed model captures daily, weekly, and monthly patterns while incorporating yearly seasonal effects through lagged annual features,
- our model learns to adjust an optimal persistence forecast identified through autocorrelation analysis by combining statistical insights with deep learning capabilities, and
- our model remains tractable for operational deployment despite processing large volumes of historical data.

We demonstrate the effectiveness of our approach by forecasting the total load at a 30-hour horizon in the Belgian grid system, comparing our results against operational forecasts provided by the Belgian transmission system operator ELIA and several traditional ML algorithms. This specific forecasting horizon has critical operational significance: in day-ahead electricity markets, load forecasts must typically be submitted by late afternoon (around 6 PM) for the entire following day's operations. This means a single forecast value must cover the entire 30-hour period from the submission deadline through the end of the next day. While Belgian load data serves as our case study, the architecture is designed to be generalizable to other power systems and forecasting horizons.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work in electricity load forecasting, emphasizing deep learning approaches. Section 3 outlines our model architecture and training procedure, while subsection 3.1 describes the data sources and preprocessing methodology used in our case study. Section 4 defines the numerical simulations used to validate our proposal, and Section 5 presents the experimental results and comparative analysis, and discusses implications and generalizability. Lastly, Section 6 concludes with key findings and suggestions for future research.

2 Literature Review

Electricity load forecasting (ELF) has evolved significantly over the decades, driven by the critical need for accurate predictions in power system operations and energy markets [1,15]. Methodologies have progressed from traditional statistical approaches to sophisticated machine learning (ML) and deep learning (DL) techniques, often culminating in hybrid models that leverage the strengths of multiple methods [12].

Conventional approaches often rely on statistical time series models. Autoregressive Integrated Moving Average (ARIMA) models and their seasonal variants (SARIMA) capture linear dependencies and seasonality effectively [29,4]. Exponential smoothing methods, particularly Holt-Winters, have also been widely used for short-term forecasting [30]. Regression-based models incorporating calendar effects, weather variables, and economic indicators provide interpretability but often struggle with complex non-linearities [17,1].

The limitations of purely statistical methods led to the adoption of machine learning techniques. Support Vector Regression (SVR) demonstrated improved handling of non-linear relationships [6,35]. Ensemble methods like Random Forests and Gradient Boosting offer robustness [20,28]. Stacking or blending predictions from multiple base models (e.g., ANN, XGBoost, LSTM, SVR) using a meta-learner is another popular strategy to improve robustness and accuracy [35,16]. In general, these methods typically require careful, often manual, feature engineering to incorporate domain knowledge effectively, limiting their ability to automatically discover intricate temporal patterns across diverse timescales.

The advances of deep learning methods have significantly improved ELF by enabling automatic feature extraction from raw data [15,12]. Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) [34,33] and Gated Recurrent Unit (GRU) [23,9] architectures, became popular due to their inherent ability to model sequential dependencies [19,25]. However, training RNNs on very long sequences, as often required for capturing yearly seasonality in medium-term forecasting, can be computationally challenging and prone to issues like vanishing gradients.

Convolutional Neural Networks (CNNs), originally developed for image processing, have emerged as a powerful alternative for time series analysis, including ELF [7,14]. CNNs excel at extracting local patterns through convolutional filters and are highly parallelizable, making them efficient for processing long sequences [32]. Multi-channel or parallel CNN architectures have been proposed to process different input variables (e.g., load, weather) independently before fusion [22], aligning with the dual-path concept in our work. Compared to RNNs, CNNs can be less sensitive to long-range dependencies but are adept at capturing hierarchical features across multiple timescales [31].

Recognizing that no single model is universally optimal, hybrid approaches combining different techniques have become a major research trend [1,15]. Many studies employ signal decomposition techniques like Variational Mode Decomposition (VMD) [34,23], Empirical Wavelet Transform (EWT) [14], Empirical Mode Decomposition (EMD) [26], or wavelet transforms [13,18] to separate the load signal into simpler components (e.g., trend, seasonality, noise). Different models (often LSTMs, GRUs, or SVMs) are then trained on these components before reconstructing the final forecast. While effective, these methods can introduce complexity and potential information leakage if not carefully implemented [34].

Combining different DL architectures, such as CNNs for feature extraction followed by LSTMs for sequence modeling (CNN-LSTM) [31] and, potentially, attention mechanisms [9], aims to leverage the complementary strengths of different network types. Novel architectures like Kolmogorov-Arnold Networks (KANs) adapted for recurrence (KARN) [8] or Residual Networks [13] also continue to emerge.

Lastly, integrating statistical insights, such as persistence forecasts, with ML/DL models is a promising direction. Persistence models provide strong baselines, especially for short horizons or regular patterns [21]. Li et al. [21] combined persistence with feedforward networks, but without the automatic feature extraction of CNNs or a systematic method for identifying the optimal persistence lag. Other advanced concepts gaining traction in ELF include transfer learning for data-scarce scenarios [33], spatial-temporal modeling for distributed loads [18], forecasting for integrated energy systems [26], and incorporating explainability techniques [2].

3 Dual-CNN architecture with adaptative persistence

This section details our forecasting approach for the Total Load prediction task. We start detailing the data preparation for the forecasting algorithm (Section 3.1). Next, we introduce our novel dual-CNN architecture (Section 3.2) designed to leverage both load-related and meteorological data. We describe the specialized CNN feature extractors that process these distinct data streams, followed by our adaptive integration layer that combines extracted features with persistence-based forecasts.

3.1 Data preparation

Let $x \in \mathbb{R}$ be a variable observed over a discrete time scale λ within a period $t \in \{1, 2, \ldots, T\}$ where $T \in \mathbb{N}$ is the number of observations. Hence, a univariate time series can be defined as a sequence of observations $\{x^{(t)}\}_{t=1}^{T} = \{x^{(1)}, x^{(2)}, \ldots, x^{(T)}\}$. Similarly, we can define a multivariate time series as a sequence $\{X^{(t)}\}_{t=1}^{T} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(T)}\}$ of vectors of D variables, such that $\mathbf{X}^{(t)} = [x_1^{(t)}, x_2^{(t)}, \ldots, x_D^{(t)}]$. Typically, a model $\mathcal{F}$ is used to forecast the next $\mathcal{H}$ steps. In this paper, we assume that total load measurements ($\mathcal{L}$) and meteorological data ($\mathcal{M}$) are available in the form of a multivariate time series, such as $\mathcal{D} = |\mathcal{L}| + |\mathcal{M}|$.

All independent data sources are converted to UTC and merged on their respective timestamp fields for temporal alignment. The merged dataset is then resampled to match a given dataset's granularity. Let us assume that $\mathbf{X} \in \mathbb{R}^{\mathcal{D}\times T}$ is the resulting dataset of this pre-processing step, and that $\mathbf{X}^{\mathcal{L}} \in \mathbb{R}^{\mathcal{L}\times T}$ and $\mathbf{X}^{\mathcal{M}} \in \mathbb{R}^{\mathcal{M}\times T}$ refer to the multivariate time series containing the total load measurements and meteorological variables respectively.

Firstly, we need to transform $\mathbf{X}^{\mathcal{L}}$ and $\mathbf{X}^{\mathcal{M}}$ into sets $\mathbf{Q}^{\mathcal{L}}$ and $\mathbf{Q}^{\mathcal{M}}$ of subsequences obtained by applying a rolling window of size l in $\mathbf{X}^{\mathcal{L}}$ and $\mathbf{X}^{\mathcal{M}}$ independently. Consequently, each sample in $\mathbf{Q}^{\mathcal{L}}$ and $\mathbf{Q}^{\mathcal{M}}$ is a tensor with the form $\mathcal{L} \times l$ and $\mathcal{M} \times l$ respectively. For target creation, we used the total load value h hours ahead as the prediction target for each time point. Unlike traditional forecasting problems where a complete horizon h is forecasted, we only predict a single point at time h in the future. Figure 1 shows an example of the forecasted value at time h given the observed total load until time T.

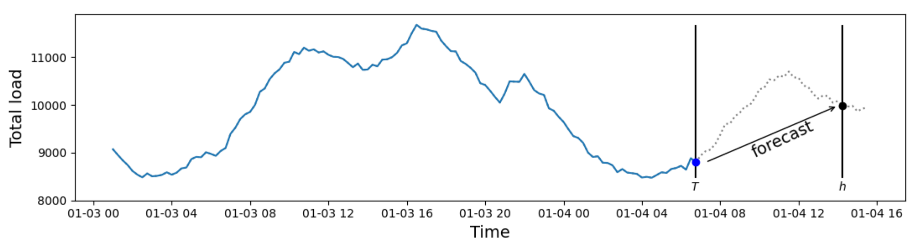

Fig. 1. Forecast of a single-point at time h in the future, given a measured variable until time T. For visualization purposes, this example only considers the total load.

A key innovation in our approach is the systematic identification and integration of optimal persistence forecasts. Therefore, an autocorrelation analysis should be conducted to determine the optimal persistence lags for the electricity load time series. This persistence analysis informed our architecture design in two fundamental ways: first, by establishing an optimal persistence baseline of length $\mathcal{B}$ for our model to adaptively adjust, and then guiding our feature selection to explicitly incorporate historical values capturing daily, weekly, and yearly patterns. The integration of this domain knowledge with deep learning techniques represents a key contribution of our approach, and it is further detailed in the next sections.

3.2 Network Architecture

Based on the persistence analysis and the multi-source nature assumed in this research, we developed a novel dual-CNN architecture with three main components. A **Load Feature extractor** block processes historical load-related time series ($\mathbf{Q}^{\mathcal{L}}$) and a **Weather Feature extractor** block processes weather-related time series ($\mathbf{Q}^{\mathcal{M}}$). Lastly, a **Adaptive Integration Layer** consists of a single linear layer that combines, through concatenation and addition operators (Figure 2), the extracted features with the optimal persistence forecast to produce the final prediction. Figure 2 provides a visual representation of this architecture.

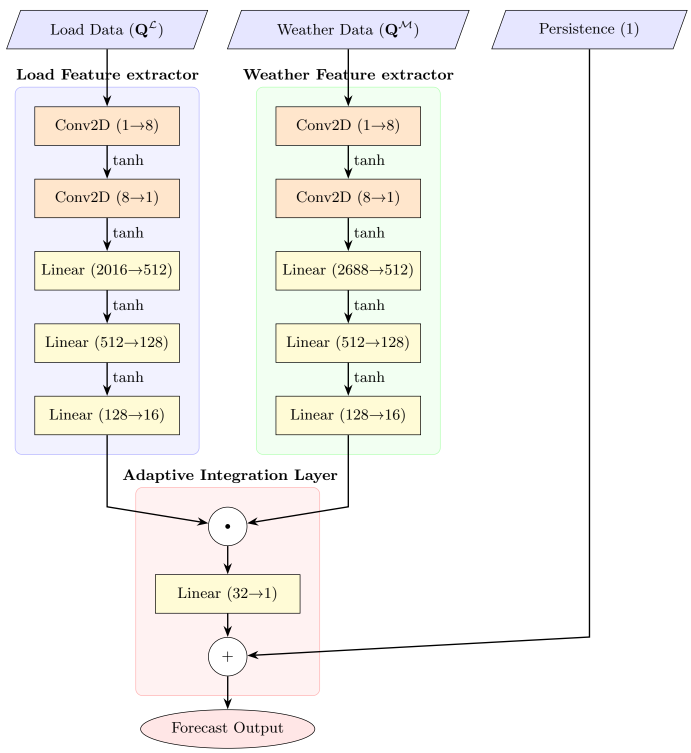

Fig. 2. Architecture of the dual-CNN architecture for total load forecasting. The model consists of two convolutional feature extractors for load and meteorological data, followed by a concatenation and a final linear layer that adjusts the baseline prediction.

Both CNN feature extractors follow identical structures but operate on different input data types. This parallel design enables the model to learn relevant patterns from independent data sources before integration. Each feature extractor begins with an input layer that accepts multi-dimensional time series data arranged as a 2D tensor, which in this case, the input layer consists of the sequence of $\mathbf{Q}^{\mathcal{L}}$ and $\mathbf{Q}^{\mathcal{M}}$ for total load measurements and meteorological variables respectively.

The structure of a feature extractor block contains a convolutional layer with 8 filters, using 3×3 kernels and "same" padding, to capture local patterns across both feature and time dimensions. Next, a second convolutional layer with a single filter (also using a 3×3 kernel and "same" padding) integrates the output of the 8 channels from the first layer. The architecture then performs feature flattening, converting 2D outputs to 1D vectors for further processing. Finally, dimensionality reduction occurs through three fully-connected layers with sizes 512, 128, and 16 nodes, progressively compressing the extracted features into a compact representation. Notice that the output of the first two CNN layers and Linear layers is passed through the hyperbolic tangent (tanh) activation function to add a non-linearity and maintain the gradient flow stable for deep networks.

The final component of our architecture is the Adaptive Integration Layer that concatenates the output of the two Feature Extractor blocks to obtain a 32-dimensional feature vector (16 from each CNN extractor). Then, a single linear layer processes this vector to produce an adjustment value that is further combined with the persistence forecast via element-wise addition to produce the model's final prediction.

This design choice leverages the already strong predictive power of optimal persistence forecasts, simplifies the learning task by focusing on predicting the correction rather than absolute values, and provides interpretability by separating the persistence forecast from the model's adjustments. It also allows for easy adaptation to different systems by identifying system-specific optimal persistence lags. This combination of CNNs for feature extraction with an adaptive persistence approach represents a novel contribution to electricity load forecasting methodology.

4 Numerical simulations

Our study uses two primary data sources: Belgian total load measurements (including system imbalance information) and weather data, collected from 26/07/2016 until 23/03/2025. The Belgian total load data were obtained from ELIA's open data platform [11], which contains measurements taken at 15-minute intervals. Each entry includes the total load, the system imbalance, and the `Most Recent Forecast` values provided by ELIA approximately 30 hours before the corresponding forecasting time point. The latter is used in this research as the baseline forecast for comparison. System imbalance data, representing deviations between generation, consumption, and commercial transactions, was obtained from ELIA's open data platform [10]. These 15-minute interval measurements provide information about real-time market and system conditions that may correlate with load forecast errors.

Hourly meteorological measurements across Belgium were retrieved from the Royal Meteorological Institute [24]. The data collected from several weather stations distributed throughout Belgium were aggregated to provide a comprehensive representation of the country's weather conditions.

Since these data sources have different data granularity, we resampled the total load and system imbalance measurements to hourly intervals to match the meteorological data's granularity. Additionally, we included the previous year's load for the same forecast horizon. Forward and backward filling were applied to handle any missing values in the data. Through this process, we created a comprehensive dataset spanning multiple years with hourly resolution, allowing for robust model training and evaluation.

In summary, three features related to the total load (actual total load, total load one year before the forecasting point, and System imbalance) and four weather variables (soil average temperature, precipitation quantity, wind speed, and air pressure) were used as input features to our proposal and all the models included in the benchmark. For target creation, we designated the total load value 30 hours ahead as demonstrated in Figure 1.

A key innovation in our approach is the identification and integration of optimal persistence forecasts. Before developing our neural network architecture, we conducted a comprehensive autocorrelation analysis to identify optimal persistence lags for electricity load time series. For our Belgian case study, we systematically evaluated lags from 1 hour up to two years (17,520 hours), calculating the root mean square error (RMSE) for each lag when used to predict values 30 hours ahead (the target moment in time).

Figure 3 shows that the persistence analysis highlighted a strong daily (24-hour), weekly (168-hour), and yearly (8,760-hour) patterns in the load data. Notably, the weekly lag demonstrated the strongest predictive power, suggesting that the load value from exactly one week before the forecast horizon provides a strong persistence forecast. This finding aligns well with established insights into electricity consumption behavior, which typically follow weekly cycles of economic and social activity. This persistence analysis informed our architecture design in two fundamental ways: first, by establishing the one-week persistence as an optimal persistence forecast for our model to adaptively adjust NN's predictions, and then guiding our feature selection to explicitly incorporate historical values capturing daily, weekly, and yearly patterns.

Based on the monthly cycle observed in the persistence analysis, we decided to use a lag value of 28 days (representing 672 observations in the hourly preprocessed dataset). Therefore, each instance in $\mathbf{Q}^{\mathcal{L}}$ and $\mathbf{Q}^{\mathcal{M}}$ is a matrix of shape 3×672 and 4×672, respectively (or 2,016 and 2,688 values, respectively, if flattened). Based on this analysis, the proposed architecture efficiently captures patterns at multiple scales across the 28-day input window, from short-term fluctuations to weekly cycles. The integration of this domain knowledge with deep learning techniques represents a key contribution of our approach.

The model was trained through a comprehensive procedure that began with the data preparation procedure explained previously. To ensure robust evaluation, a 5-fold cross-validation scheme with shuffling was adopted. The optimization of the model's parameters was conducted with the Adam optimizer with an

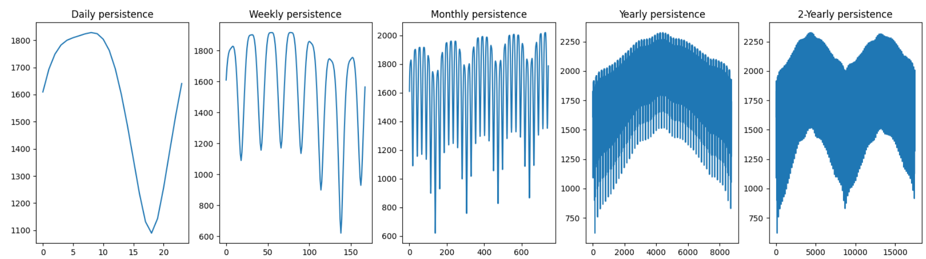

Fig. 3. Persistence analysis of the hourly-sampled total load data. Each point represents the RMSE of a persistent model using the lag given on the x axis. The five plots show the analysis on different time scales, for day, week, month, year, and two years. The analysis reveals strong daily, weekly, and yearly cycles. A weekly lag (7 days) stands out as the most effective for persistence modeling, indicating that using the load value from one week prior yields the best forecast. Monthly patterns are notably less distinct.

initial learning rate of 0.0002, combined with a *ReduceLROnPlateau* scheduler (factor=0.75, patience=5) for adaptive learning rate adjustment. Mean squared error (MSE) between predicted and actual load values served as the loss function. Each fold underwent training for 30 epochs with a batch size of 32. For evaluation purposes, both our model and ELIA's forecasts were evaluated using RMSE and Mean Absolute Error (MAE) on the same validation sets, enabling direct comparison. In addition, we report the training and test times (in seconds) of each forecasting model computed by adding the time needed to train/test the algorithm in each fold, respectively.

We contrast the proposed architecture's performance against several ML algorithms: Elastic net (ElasticNet)[1], Gradient boosting(GBR)[2], k-nearest neighbors(KNN)[3], Random Forest (RF)[4] with and without feature selection(PipeRF)[5],

[1] `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.ElasticNet.html`

[2] `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingRegressor.html`

[3] `https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html`

[4] `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html`

[5] `https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectFromModel.html`

Lasso regression (Lasso)[6], PLS[7], Ridge regression (Ridge)[8], and LightGBM[9]. The library HyperOpt [3] was considered to perform the hyperparameter optimization of these algorithms (20 iterations) using the MAE as the optimization objective. Appendix A summarizes the hyperparameters considered during the optimization. Unlike the proposed approach, which uses a set of CNN kernels to extract relevant features directly from the data, the benchmark algorithms require training on high-dimensional datasets (75,198 instances and 4,704 features). Therefore, we included an intermediate step to perform data fusion through Principal Component Analysis (PCA) to reduce the number of features before training the benchmarking algorithms. PCA revealed that 1,500 components are enough to represent the original set of features, explaining more than 98.6% of the total variance in the data. Therefore, we used this reduced dataset to train the algorithms in the benchmark to keep processing time realistic and make the evaluation process more practical and efficient.

5 Results and discussion

Table 1 presents the cross-validation results, comparing our neural network model against ELIA's operational forecasts and the algorithms included in the benchmark.

The superior performance of our dual-CNN architecture compared to operational forecasts in the Belgian case study demonstrates several key advantages of our approach. The CNN architecture captures complex temporal patterns across multiple timescales without requiring explicit feature engineering or predetermined lag structures, which is particularly valuable for electricity load forecasting, where patterns exist at daily, weekly, and seasonal levels. By using parallel CNNs for different data types, our architecture learns optimal representations for each source before integration, recognizing that load data and meteorological data have different characteristics and temporal dynamics. Despite processing a month of historical data at hourly resolution, our architecture remains computationally tractable through judicious dimensionality reduction, with the final 32-dimensional feature vector effectively compressing the relevant information from 7 features across 672 time points. Rather than predicting absolute load values, our model learns to adjust an already strong persistence baseline, simplifying the learning task while maintaining high accuracy.

While our experimental validation focused on Belgian load data, several aspects of our architecture suggest broader generalizability. The system-agnostic design of the dual-CNN architecture makes no assumptions specific to the Belgian power system and could be applied to any region where similar data sources

[6] `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html`

[7] `https://scikit-learn.org/stable/modules/generated/sklearn.cross_decomposition.PLSRegression.html`

[8] `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html`

[9] `https://lightgbm.readthedocs.io/en/stable/`

Table 1. Cross-validation results of the benchmarking of our proposal

Algorithms	MAE (mean ± std.)	RMSE (mean ± std.)	Training time (mean (sec.))	Validation time (mean (sec.))
ELIA forecast	164.6831 ± 1.48	205.8304 ± 2.03	-	-
Our proposal	**146.4913 ± 8.86**	**187.1914 ± 10.36**	628.29	2.36
ElasticNet	313.726 ± 2.02	433.6901 ± 1.67	1.06	0.03
GBR	228.3367 ± 2.64	298.4986 ± 3.71	15.61	0.03
KNN	229.8504 ± 1.93	307.2091 ± 2.68	9.62	1791.09
Lasso	313.6927 ± 1.97	433.6443 ± 1.61	2.55	0.03
PipeRF	472.2681 ± 4.62	626.8217 ± 5.20	303	0.18
PLS	314.9771 ± 1.75	435.2407 ± 1.53	46.17	0.1
Ridge	314.9765 ± 1.75	435.24 ± 1.53	1.18	0.05
RF	908.5341 ± 3.25	1130.5639 ± 3.71	68.28	0.17
LightGBM	206.4131 ± 1.41	270.9815 ± 2.10	199.71	0.33

are available. Our approach features flexible persistence integration, as the persistence analysis component can identify system-specific optimal lags, adapting the model to different consumption patterns across regions. Additionally, the CNN architecture offers a scalable input window, meaning that although we used 28 days of history for our experiments, it can accommodate different input window lengths based on specific forecasting needs. The model also provides an adaptable forecasting horizon; while we focused on 30-hour ahead forecasting, the architecture could be trained for different horizons by adjusting the target definition accordingly. This generalizability makes our approach potentially valuable for a wide range of electricity forecasting applications beyond the specific case study presented.

The medium-term forecasting capabilities of our architecture have significant practical importance. Improved forecasts in the 24-48 hour horizon enhance day-ahead operational planning, reserve allocation, and maintenance scheduling for system operators. For market participants, more accurate forecasts can reduce imbalance costs, improve bidding strategies, and optimize generation scheduling. The demonstrated improvement over operational forecasts suggests potential economic value that could justify implementation costs, especially in large power systems. Additionally, the architecture could be extended to integrate renewable generation forecasts, supporting the transition to low-carbon energy systems.

Despite the promising results, several limitations should be acknowledged. The model's reliance on historical patterns means that unprecedented events (e.g., pandemic lockdowns) may still pose challenges without specific adaptation mechanisms but the persistence baseline helps tackling this. While our approach integrates meteorological data, it does not explicitly account for forecast uncertainty in these inputs, which could affect operational reliability. The current implementation produces point forecasts rather than probabilistic distributions, limiting uncertainty quantification for risk-aware decision-making. Despite the interpretable persistence component, the CNN feature extractors remain largely black-box, limiting full explainability of predictions.

6 Conclusion and Future Work

This paper proposes a dual-CNN architecture with adaptive persistence integration for medium-term electricity load forecasting. The model extracts features from multiple data sources using parallel convolutional networks and refines a persistence forecast through an adaptive mechanism. It captures temporal patterns across multiple timescales without extensive feature engineering. Applied to forecasting a single load value 30 hours ahead for the Belgian grid, the approach consistently outperforms both the operational forecasts of the national TSO and several machine learning baselines.

The proposed architecture offers several key contributions to electricity load forecasting. It introduces a generalizable dual-CNN design capable of efficiently extracting features from diverse time series inputs, along with a systematic integration of optimal persistence forecasts informed by domain knowledge. The model balances forecasting accuracy with computational efficiency, enabling the use of long historical windows and flexible adaptation to various power systems and forecasting horizons. Beyond the Belgian case study, the framework has broader applicability in supporting reliable and efficient grid operations, including those transitioning toward renewable-dominant energy systems.

Future research will focus on extending the architecture to generate probabilistic forecasts, enabling uncertainty quantification. Enhancing multi-horizon prediction capability and integrating recurrent components (e.g., LSTM, GRU) could improve temporal consistency and long-term dependency modeling. Incorporating additional inputs such as electricity prices, renewable generation forecasts, and cross-border flows may enhance accuracy. Opportunities also exist in transfer learning, adapting models trained on data-rich systems to data-scarce regions. Finally, robustness to extreme events could be improved through synthetic data generation and adversarial training techniques.

Data and Code Availability

The datasets used in this study are publicly available. Electricity load and system imbalance data were obtained from the ELIA Open Data platform, and meteorological data were sourced from the Royal Meteorological Institute of Belgium. All sources are cited in Section 4. The source code for the proposed model and the experiments presented in this paper is publicly available on GitHub: `https://github.com/gmpal/belgian-load-forecasting`.

Acknowledgment

The Service Public de Wallonie Recherche has supported this research work under grant nr 2010235–ARIAC by DigitalWallonia4.ai., and by the Joint R&D TORRES Project (2022-RDIR-59b) funded by "Région de Bruxelles-Capitale - Innoviris". Computational resources have been provided by the Consortium des Équipements de Calcul Intensif (CÉCI), funded by the Fonds de la Recherche Scientifique de Belgique (F.R.S.-FNRS) under Grant No. 2.5020.11 and by the Walloon Region.

A Hyperparameter configuration of the algorithms included in the benchmark

ElasticNet: 'alpha': 0.7373749519, 'l1_ratio': 0.986225753, 'max_iter': 601.0, 'selection': 'cyclic'

GBR: 'l2_regularization': 0.53676543, 'learning_rate': 1.2163556646, 'loss': 'poisson', 'max_depth': 19.0, 'max_iter': 191.0, 'max_leaf_nodes': 49.0, 'min_samples_leaf': 10.0

KNN: 'algorithm': 'ball_tree', 'n_neighbors': 3, 'weights': 'distance'

Lasso: 'alpha': 0.6828204633076271, 'l1_ratio': 0.5963760494, 'max_iter': 862.0, 'selection': 'random'

PipeRF: 'criterion': 'squared_error', 'max_depth': 49.0, 'min_samples_leaf': 0.003143505644945082, 'min_samples_split': 0.01048220391, 'n_estimators': 112.0

PLS: 'max_iter': 1644.0, 'n_components': 43.0

Ridge: 'alpha': 1.924628138196723

Random Forest: 'criterion': 'friedman_mse', 'max_depth': 32.0, 'min_samples_leaf': 0.128454286644, 'min_samples_split': 0.448823903, 'n_estimators': 138.0

LightGBM: 'boosting_type': 'gbdt', 'learning_rate': 1.207566799, 'max_depth': 9.0, 'min_child_samples': 15.0, 'n_estimators': 183.0, 'num_leaves': 85.0, 'reg_alpha': 1.99499047, 'reg_lambda': 1.287815559

References

1. Alfares, H.K., Nazeeruddin, M.: Electric load forecasting: literature survey and classification of methods. International journal of systems science **33**(1), 23–34 (2002)
2. Baur, L., Ditschuneit, K., Schambach, M., Kaymakci, C., Wollmann, T., Sauer, A.: Explainability and interpretability in electric load forecasting using machine learning techniques–a review. Energy and AI p. 100358 (2024)
3. Bergstra, J., Yamins, D., Cox, D.: Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In: International conference on machine learning. pp. 115–123. PMLR (2013)
4. Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time series analysis: forecasting and control. John Wiley & Sons (2015)

5. Buroni, G., Lebichot, B., Bontempi, G.: Ast-mtl: An attention-based multi-task learning strategy for traffic forecasting. IEEE Access **PP**, 1–1 (05 2021). https://doi.org/10.1109/ACCESS.2021.3083412
6. Chen, B.J., Chang, M.W., et al.: Load forecasting using support vector machines: A study on eunite competition 2001. IEEE transactions on power systems **19**(4), 1821–1830 (2004)
7. Chen, K., Chen, K., Wang, Q., He, Z., Hu, J., He, J.: Short-term load forecasting with deep residual networks. IEEE Transactions on Smart Grid **10**(4), 3943–3952 (2018)
8. Danish, M.U., Grolinger, K.: Kolmogorov–arnold recurrent network for short term load forecasting across diverse consumers. Energy Reports **13**, 713–727 (2025)
9. Dong, J., Jiang, Y., Chen, P., Li, J., Wang, Z., Han, S.: Short-term power load forecasting using bidirectional gated recurrent units-based adaptive stacked autoencoder. International Journal of Electrical Power & Energy Systems **165**, 110459 (2025)
10. ELIA: Current system imbalance (near real-time). https://opendata.elia.be/explore/dataset/ods169/ (2024), accessed: 2025-04-22
11. Elia Transmission Belgium SA: Measured and forecasted total load on the belgian grid (historical data). https://opendata.elia.be/explore/dataset/ods001/ (2024), accessed: 2025-04-22
12. Eren, Y., Küçükdemiral, İ.: A comprehensive review on deep learning approaches for short-term load forecasting. Renewable and Sustainable Energy Reviews **189**, 114031 (2024)
13. Eskandari, H., Imani, M., Moghaddam, M.P.: A deep residual network integrating entropy-based wavelet packet ensemble model for short-term electrical load forecasting. Energy **314**, 134168 (2025)
14. Fan, G.F., Han, Y.Y., Li, J.W., Peng, L.L., Yeh, Y.H., Hong, W.C.: A hybrid model for deep learning short-term power load forecasting based on feature extraction statistics techniques. Expert Systems with Applications **238**, 122012 (2024)
15. Fida, K., Abbasi, U., Adnan, M., Iqbal, S., Mohamed, S.E.G.: A comprehensive survey on load forecasting hybrid models: Navigating the futuristic demand response patterns through experts and intelligent systems. Results in Engineering p. 102773 (2024)
16. Guo, F., Mo, H., Wu, J., Pan, L., Zhou, H., Zhang, Z., Li, L., Huang, F.: A hybrid stacking model for enhanced short-term load forecasting. Electronics **13**(14), 2719 (2024)
17. Hong, T., Pinson, P., Fan, S., Zareipour, H., Troccoli, A., Hyndman, R.J.: Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond (2016)
18. Jalalifar, R., Delavar, M.R., Ghaderi, S.F.: Sac-convlstm: A novel spatio-temporal deep learning-based approach for a short term power load forecasting. Expert Systems with Applications **237**, 121487 (2024)
19. Kong, W., Dong, Z.Y., Jia, Y., Hill, D.J., Xu, Y., Zhang, Y.: Short-term residential load forecasting based on lstm recurrent neural network. IEEE transactions on smart grid **10**(1), 841–851 (2017)
20. Lahouar, A., Slama, J.B.H.: Day-ahead load forecast using random forest and expert input selection. Energy Conversion and Management **103**, 1040–1051 (2015)
21. Li, P., Zhou, K., Lu, X., Yang, S.: A hybrid deep learning model for short-term pv power forecasting. Applied Energy **259**, 114216 (2020)

22. Liang, R., Yang, L., Wu, S., Li, H., Jiang, C.: A three-stream cnn-lstm network for automatic modulation classification. In: 2021 13th International Conference on Wireless Communications and Signal Processing (WCSP). pp. 1–5. IEEE (2021)
23. Ma, K., Nie, X., Yang, J., Zha, L., Li, G., Li, H.: A power load forecasting method in port based on vmd-icss-hybrid neural network. Applied Energy **377**, 124246 (2025)
24. Royal Meteorological Institute of Belgium: Hourly meteorological measurements. `https://www.meteo.be/en` (2024), accessed: 2025-04-22
25. Shi, H., Xu, M., Li, R.: Deep learning for household load forecasting—a novel pooling deep rnn. IEEE transactions on smart grid **9**(5), 5271–5280 (2017)
26. Shi, J., Teh, J.: Load forecasting for regional integrated energy system based on complementary ensemble empirical mode decomposition and multi-model fusion. Applied Energy **353**, 122146 (2024)
27. Sun, Y., Zhu, L., Wang, G., Zhao, F.: Multi-input convolutional neural network for flower grading. Journal of Electrical and Computer Engineering **2017** (2017)
28. Taieb, S.B., Hyndman, R.J.: A gradient boosting approach to the kaggle load forecasting competition. International journal of forecasting **30**(2), 382–394 (2014)
29. Taylor, J.W.: Short-term electricity demand forecasting using double seasonal exponential smoothing. Journal of the Operational Research Society **54**(8), 799–805 (2003)
30. Taylor, J.W., De Menezes, L.M., McSharry, P.E.: A comparison of univariate methods for forecasting electricity demand up to a day ahead. International journal of forecasting **22**(1), 1–16 (2006)
31. Ullah, K., Ahsan, M., Hasanat, S.M., Haris, M., Yousaf, H., Raza, S.F., Tandon, R., Abid, S., Ullah, Z.: Short-term load forecasting: A comprehensive review and simulation study with cnn-lstm hybrids approach. IEEE Access (2024)
32. Wang, X., Zhao, Y., Pourpanah, F.: Recent advances in deep learning. International Journal of Machine Learning and Cybernetics **11**, 747–750 (2020)
33. Wei, N., Yin, C., Yin, L., Tan, J., Liu, J., Wang, S., Qiao, W., Zeng, F.: Short-term load forecasting based on wm algorithm and transfer learning model. Applied Energy **353**, 122087 (2024)
34. Xu, A., Chen, J., Li, J., Chen, Z., Xu, S., Nie, Y.: Multivariate rolling decomposition hybrid learning paradigm for power load forecasting. Renewable and Sustainable Energy Reviews **212**, 115375 (2025)
35. Yang, Q., Tian, Z.: A hybrid load forecasting system based on data augmentation and ensemble learning under limited feature availability. Expert Systems with Applications **261**, 125567 (2025)