Internet of Things 27 (2024) 101322

Contents lists available at ScienceDirect

Internet of Things

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

Research article

Electricity consumption forecasting for sustainable smart cities using machine learning methods

Darius Peteleaza [a], Alexandru Matei [a], Radu Sorostinean [a], Arpad Gellert [a,*], Ugo Fiore [b], Bala-Constantin Zamfirescu [a], Francesco Palmieri [b]

[a] *Computer Science and Electrical Engineering Department, Lucian Blaga University of Sibiu, Emil Cioran 4, 550025 Sibiu, Romania*
[b] *Department of Computer Science, University of Salerno, Via Giovanni Paolo II, 132, I-84084, Fisciano (SA), Italy*

ARTICLE INFO

Keywords:
Electricity consumption forecasting
Smart city
Time-series dense encoder
Machine learning
Sustainability

ABSTRACT

Integrating smart grids in smart cities is pivotal for enhancing urban sustainability and efficiency. Smart grids enable bidirectional communication between consumers and utilities, enabling real-time monitoring and management of electricity flows. This integration yields benefits such as improved energy efficiency, incorporation of renewable sources, and informed decision-making for city planners. At the city scale, forecasting electricity consumption is crucial for effective resource planning and infrastructure development. This study proposes using a time-series dense encoder model for short-term and long-term forecasting at the city level, showing its superior performance compared to traditional approaches like recurrent neural networks and statistical methods. Hyperparameters are optimized using the non-dominated sorting genetic algorithm. The model's efficacy is demonstrated on a six-year dataset, highlighting its potential to significantly improve electricity consumption forecasting and enhance urban energy system efficiency.

1. Introduction

The integration of smart grids within the context of smart cities represents a significant convergence of technological advancements aimed at enhancing urban sustainability, resilience, and efficiency. Smart grids, characterized by their utilization of advanced digital communication and control technologies, fundamentally modernize traditional electricity grid infrastructure. These systems facilitate bidirectional communication between consumers and utility providers, enabling real-time monitoring, analysis, and management of electricity flows.

In the context of urban development, smart cities leverage technology and data-driven solutions to address various challenges and improve residents' living standards. The integration of smart grid technologies into smart city initiatives yields several tangible benefits. Firstly, it promotes energy efficiency by optimizing energy consumption and reducing waste through real-time monitoring, energy analytics, and demand-side management strategies. Secondly, smart grids contribute to urban sustainability goals by enabling the incorporation of renewable energy sources and supporting clean transportation options, such as electric vehicles, thereby reducing carbon emissions and promoting environmental stewardship. Finally, the data generated by smart grid systems can inform data-driven decision-making processes for city planners and policymakers, aiding in strategic energy infrastructure investments, urban development strategies, and formulation of environmental policies, thereby fostering long-term sustainability and liveability within smart cities.

* Corresponding author.
E-mail addresses: darius.peteleaza@ulbsibiu.ro (D. Peteleaza), alex.matei@ulbsibiu.ro (A. Matei), radutraian.sorostinean@ulbsibiu.ro (R. Sorostinean), arpad.gellert@ulbsibiu.ro (A. Gellert), ufiore@unisa.it (U. Fiore), constantin.zamfirescu@ulbsibiu.ro (B.-C. Zamfirescu), fpalmieri@unisa.it (F. Palmieri).

https://doi.org/10.1016/j.iot.2024.101322
Received 18 June 2024; Received in revised form 2 August 2024; Accepted 5 August 2024
Available online 6 August 2024

The data generated by smart grid systems is multifaceted, generated by multiple actors, and amenable to analysis at different levels of aggregation. The focus of this article is on forecasting electricity consumption, at the city level, of residential customers and the technological consumption of electricity suppliers. At this level, having an accurate electricity consumption forecast, city planners and utility companies can plan resources more effectively, predict future energy needs and identify necessary infrastructure, such as power plants, transmission lines, and distribution networks. Another advantage is efficient resource allocation, allowing for the optimized deployment of renewable energy sources, backup power capacity, and targeted energy efficiency programs, resulting in significant cost savings. Using both short-term and long-term forecasts, grid operators can effectively manage and anticipate peak demand periods, balance supply and demand in real time, and reduce the likelihood of blackouts, ensuring continuous power supply for residents. Lastly, forecasts empower city officials to make informed decisions about energy infrastructure investments, policy priorities, and resource allocation strategies, facilitating data-driven decision-making and ensuring a sustainable and efficient energy system. Overall, accurate forecasting of electricity consumption is crucial for developing efficient, resilient, and sustainable energy systems that meet the needs of cities and their residents.

To forecast electricity consumption, we propose using a method that is suitable for both short-term and long-term forecasts. We chose to make use of a Time-series Dense Encoder (TiDE) model, which has the ability to maintain and even surpass transformer models while having a simpler multi-layer perceptron architecture [1]. Additionally, we use the Non-dominated Sorting Genetic Algorithm III (NSGA-III) [2,3] to tune the model hyperparameters. To showcase the performance of our approach, we test and validate the hyperparameter-optimized TiDE model on a real-world dataset that spans six years on both short-term and long-term forecasting. We also show that the optimized TiDE model outperforms traditional approaches by comparing the results to other methods like Long Short-Term Memory (LSTM), Markov chains, or statistical models based on Trigonometric seasonality, Box–Cox transformation, Auto-Regressive Moving Average errors, Trend and Seasonal components (TBATS).

The rest of the paper is organized as follows. Section 2 describes the state-of-the-art in terms of electricity consumption forecasting. Section 3 details the methodology used, describing the general approach. Section 4 describes the specific dataset used, together with the model results, comparison, and discussion. Finally, Section 5 concludes the paper and presents further work directions.

2. Related work

Time series forecasting is an area of research whose aim is to predict future values based on historical data patterns. It involves analyzing sequential data points collected over time to identify trends, seasonal variations, and other underlying patterns that can help to anticipate future outcomes. This approach is extensively applied across multiple fields, including economics and finance [4], meteorology [5], engineering [6], or healthcare [7], to inform decision-making, develop strategies, and optimize resource allocation. Over the years, significant steps have been made in enhancing the accuracy, robustness, and applicability of time series forecasting techniques, supported by advancements in computational power, new state-of-the-art methods, and data availability. In the energy domain, forecasting of electricity demand and consumption takes place at different scales, from single houses, or large-scale buildings, to cities, industries, and even countries, while also considering both short-term and long-term predictions.

A hybrid method that uses several statistical and machine learning methods is presented in [8] and incorporates empirical mode decomposition, particle swarm optimization, support vector regression, thermal reaction dynamics theory and an econometric model. This complex model is used to forecast the energy consumption of New South Wales state of Australia. Pursuing the sustainable development and regulation of electricity, the hybrid method was analyzed using Nash equilibrium and Porter's five-force model.

Using the Algerian market as a study case, the authors of [9] are using ensemble deep learning to forecast electricity consumption to enhance the planning and management of energy resources. To predict the monthly energy consumption, three models are successfully combined: LSTM, Temporal Convolutional Networks (TCN), and Gated Recurrent Unit neural network (GRU). With a dataset spanning 14 years, and almost 2000 consumers, their approach achieved an average error of 67.42 MWh, surpassing both traditional methods and the expectations of the local energy supplier.

For short-term forecasting of electricity consumption in office buildings, three ensemble learning methods are used [10]. The authors present random forests, gradient-boosted regression trees, and Adaboost and compare them to fuzzy-based systems and support vector machine approaches. On one-hour ahead forecasting, the adapted Adaboost methods showed the best performance.

A method for day-ahead electricity consumption forecast for residential buildings based on Gaussian mixture clustering, XGBoost and neural networks was proposed in [11]. First, Gaussian clustering is used to create user behavior clusters. Then, using an XGBoost classification model, a day-ahead prediction of user behavior is made that is fed into a Multi-Layer Perceptron (MLP) model that is used to make the final electricity consumption prediction. As their method is based on user behavior, a different MLP model with 1 hidden layer and 100 neurons was trained for each of the over 500 households in their datasets. Compared to a baseline with no behavior modeling, where only the final MLP is used, the proposed method achieved a smaller Mean Absolute Percentage Error (MAPE), with a difference of 7% on average. When looking only at the perfectly assigned behavior-modeled households, the MAPE was reduced by 20%, from 66% to 46.1%.

Daily electricity demand prediction for grid operation and management was studied in [12]. Similarly to other approaches, a hybrid multi-algorithm method was used that combined an artificial neural network, with an Encoder–Decoder-based LSTM (EDLSTM) and Improved Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (ICMD). The model was tested on datasets from four different electrical substations. To achieve the best results, the authors used Bayesian optimization for hyperparameter tuning. Using ICMD, the input data is decomposed into multiple Intrinsic Mode Functions (IMF), that are sorted by their frequency. An artificial neural network is used to predict complex, high-frequency IMFs, while for the more stable ones, the EDLSTM model is used. The final prediction is a sum of all predicted IMFs. Having a hyperparameter-optimized model for each electrical substation, the MAE result varied from 2.3 MWh to 11.2 MWh.

A comparative study of seven models was done in [13] to see their predictive performance over daily electricity use on datasets from three different metropolitan areas in the United States. The results showed that lightGBM provided the best results with a Coefficient of Variation of the Root Mean Square Error (CVRMSE) of 6.5% for Los Angeles, 4.6% for Sacramento, and 4.1% for New York. All the other tested models had their CVRMSE less than 10%. The authors also explored how extreme weather events like heat waves influence the electricity demand of the metropolitan area. The weather-sensitive component had a high influence, accounting for 30% to 50% of the daily electricity usage, where every Celsius degree increase in ambient temperature raised the daily electricity consumption by 5% on average.

While predicting the electricity consumption at household-level or city-level seems to be of high interest, directly benefiting a large number of users, large electricity companies or governments require country-wide electricity consumption prediction. The study from [14] evaluates four different models (multilayer perceptron, fuzzy time series, adaptive neuro-fuzzy inference system, and least squares support vector machines) on datasets containing monthly electricity consumption spanning 10 years from seven countries across two continents, covering different climate conditions. The models were evaluated on both short-term forecasting of one to three months ahead but also on long-term forecasting of up to one year ahead. On average, the fuzzy time series model performed best for most countries, but it was found that different models performed better depending on the dataset and different forecasting periods. Their conclusion emphasizes the necessity of evaluating multiple models when searching for the best approach for a specific dataset.

Recent deep-learning approaches for time series forecasting have focused on transformer methods [15], which have a self-attention mechanism at their core that allows the model to assess the significance of various input elements during prediction. Originally designed for natural language processing tasks, transformers have gained traction in time series forecasting because of their proficiency in modeling sequential data [16]. Transformer-based architectures for time series forecasting typically consist of encoder–decoder structures, where the encoder processes historical input sequences, and the decoder generates future predictions. Through multiple layers of self-attention and feed-forward neural networks, transformers can effectively learn complex patterns and relationships within the time series data.

A long-term forecasting transformer-based method, called Autoformer, was proposed in [17], where the authors add custom decomposition blocks and replace the attention mechanism with an autocorrelation one. With dependency discovery and representation aggregation at sub-series level, the autocorrelation mechanism outperforms the self-attention one, providing a relative improvement of 38% in long-term multivariate time series forecasting compared to other state-of-the-art methods of that time, like Informer [18], or N-BEATS [19].

Another improvement over the original transformer made specifically for long-term time series forecasting is proposed in [20] where global time series properties are captured using multiple decomposition blocks. The method, called Frequency Enhanced Decomposed Transformer (FEDformer), includes Fourier and Wavelet transformation blocks to map the time series in the frequency domain. With increased computational and memory efficiency due to linear complexity, FEDFormer achieves a reduction of the prediction error by 14.8% for multivariate and 22.6% for univariate time series forecasting.

With the fast-rising popularity of transformers applied to time series forecasting, some researchers have questioned their validity and computational complexity for this task. In [21], the authors propose a set of three simple one-layer models called LTSF-linear that outperform large transformer models. While the Vanilla Linear contains only one layer, the DLinear model contains a decomposition scheme that improves the performance of the Vanilla Linear when the data has a clear trend. To handle the distribution shift in the datasets, a model called NLinear is proposed that has simple normalization of the input sequence through subtraction and addition layers. With small computational and memory footprint, LTSF-linear proved to outperform large transformer models like Informer, Autoformer or FEDFormer in both terms of efficiency and forecasting error. In response to this challenge, the authors of [22] propose an improved transformer model for multivariate time-series forecasting called Patch Time Series Transformer (PatchTST) where the input data is divided into overlapping and non-overlapping patches and split into independent channels, reducing the spatio-temporal complexity of the attention map.

In [23–25], the authors evaluated supervised and semi-supervised artificial intelligence methods to detect electricity theft in smart urban grids for alleviating energy losses.

Previously, we explored the accurate prediction of electricity consumption at the urban level by comparing different models, including TBATS [26], fuzzy logic [26], Markov chains [27] and LSTM networks [28]. In the experiments, the statistical method TBATS, proved to achieve the smallest prediction error of 3.6 MWh on average, being followed by the LSTM network and the fuzzy controller. While the results are promising, no external factors like temperature, weather phenomena or social events were taken into account in these studies.

Similarly, a fuzzy controller is used in [29] and the TBATS in [30] for the prediction of electricity production and consumption of a smart house that is equipped with solar panels and batteries for energy storage. While achieving very good results, the fuzzy logic-based method was surpassed only by a Markov chain predictor, which has the advantage of context awareness, the TBATS being the third best model.

3. Methodology

Our approach methodology consists of several steps, as described below in Fig. 1. The first step is to analyze the dataset and preprocess it to encode and normalize the features. Next, the dataset is split into three subsets for validation, training and testing.

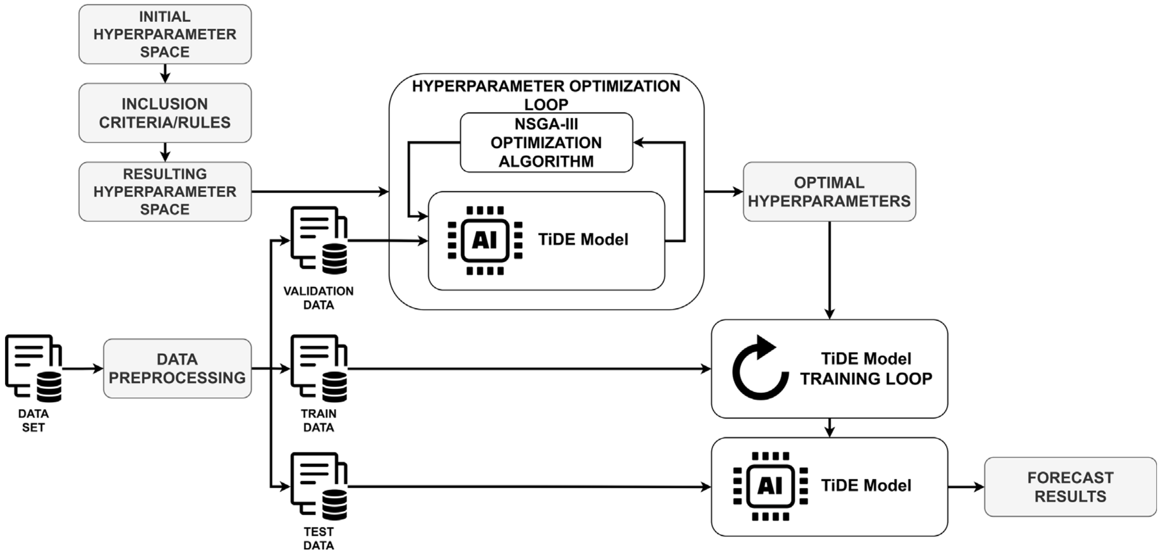

Fig. 1. Methodology workflow.

Having the initial hyperparameter space, we use different rules to eliminate unfeasible combinations that should not be evaluated, thus reducing the search space dramatically. Using the validation data as a reference, we optimize the hyperparameters of TiDE using NSGA-III, searching for the hyperparameter combination that provides the best results. Using this combination, the final TiDE forecasting model is trained and evaluated on the test data. The hyperparameter optimization and TiDE model are further detailed in this section.

3.1. TiDE model

The TiDE model introduced in [1] contains a novel encoder–decoder architecture, which employs a simple structure based on MLP for long-term time-series forecasting. TiDE exhibits capabilities in accommodating covariates and capturing non-linear dependencies while demonstrating superior computational efficiency compared to Transformer models and attaining error rates approaching optimality for linear dynamical systems that obey certain conditions. This involves the utilization of two distinct MLP networks: one for encoding past data points into a latent space representation, and another for decoding future predictions from this latent space representation jointly with the future values of the covariates. A temporal decoder block facilitates the incorporation of temporal information from past covariates, like seasonality or holidays, in forecasting future values within a given temporal sequence or trend. Notably, the TiDE architecture makes extensive use of linear skip connections, allowing it to effectively capture immediate dependencies and compliment them with the nonlinear predictions resulting from the encoder–decoder blocks. Empirical evaluations of TiDE substantiate the efficacy of this approach, indicating its superiority over previous methods in long-term time series forecasting. Furthermore, TiDE shows notable advantages in computational efficiency, exhibiting training speeds and inference latencies approximately 5–10 times faster than those observed with Transformer-based models such as PatchTST [22], or FedFormer [20].

However, the main disadvantage of TiDE is its reliance on a large number of hyperparameters. While this granularity allows for precise network configuration, it also significantly expands the hyperparameter search space, making tuning both time-consuming and labor-intensive. Our method addresses this issue by simplifying the tuning process.

3.2. Hyperparameter space exploration

The TiDE model [1], uses a broad set of hyperparameters that influence its learning capacity and effectiveness in generating predictions. However, the expansive nature of its hyperparameter space presents a significant challenge in determining the optimal configuration for the forecasting task on a specific dataset. As with any neural network model, the hyperparameters of the TiDE model directly impact its ability to learn. Inappropriate hyperparameter choices can induce overfitting, where the model memorizes the training data, resulting in poor generalization to unseen examples. Conversely, underfitting can occur if the chosen hyperparameters lead to an overly simplistic model, hindering its capacity to effectively capture the underlying patterns within the data.

Identifying the optimal hyperparameter configuration presents a sophisticated challenge. Besides being inherently complex, due to the intricate interactions between the hyperparameters, it is also computationally expensive. Evaluating the impact of each

Table 1
Hyperparameters together with their description and possible values.

Hyperparameter	Description	Possible values
batch_size	Size of mini-batch for each iteration	{16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192}
learning_rate	Learning rate for Adam optimizer	{1e−5, 3e−5, 3.82e−5, 6.55e−5, 8.39e−5, 1e−4, 2.24e−4, 2.52e−4, 3e−4, 9.99e−4, 1e−3, 1e−2, 1e−1}
dropout_rate	Dropout rate for regularization	{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9}
layer_norm	Apply layer normalization or not	{True, False}
hidden_size	Size of the hidden layers	{16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192}
decoder_output_dim	Output dimension of the decoder	{1, 2, 4, 8, 16, 32, 64, 128}
final_decoder_hidden	Hidden layer size for final decoder	{16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192}
hist_len	Length of input history (dynamic)	{24, 48, 96, 192, 336, 384, 720, 1440, 2880, 5760}
pred_len	Length of prediction period (dynamic)	{24, 48, 96, 192, 336, 384, 720, 1440, 2880, 5760}

hyperparameter combination necessitates a complete training run of the TiDE model. Given the vastness of the hyperparameter space, this iterative process necessitates substantial computational resources.

To mitigate these challenges, we use an automated hyperparameter space explorer system that streamlines the hyperparameter tuning process for the TiDE model by leveraging advanced optimization algorithms. These algorithms enable efficient exploration of the hyperparameter landscape, significantly reducing the computational burden associated with traditional approaches like grid search. We opted to utilize NSGA-III [2,3]. This algorithm, as a refinement of the well-known NSGA II, is particularly adept at handling multi-objective optimization problems. Its effectiveness stems from its ability to efficiently identify a diverse set of solutions that represent trade-offs between the competing objectives, in this case, model performance metrics. Another big advantage of NSGA-III is that the Pareto-optimal solutions are not missed during the optimization process [31].

To initiate the hyperparameter tuning process, a pre-defined search space was established for each hyperparameter. This space was constructed by meticulously combining three sources of information: best practices from the broader field of machine learning, the inherent characteristics of the TiDE model architecture, and the recommendations outlined in the seminal TiDE paper [1]. This multifaceted approach ensured that the explored hyperparameter configurations were not only aligned with recognized machine learning principles but also specifically catered to the unique functionalities of the TiDE model. Table 1 provides a list of the hyperparameters together with a description and possible values. The initial parameter space encompasses 208 million possible hyperparameter settings. Effectively navigating such a large search space requires not only significant computational resources but also underscores the critical role of employing efficient and intelligent optimization algorithms.

The extensive number of hyperparameters needed a strategic reduction in the dimensionality of the search space. This dimensionality reduction was driven by practical considerations grounded in the model's operational environment and the inherent characteristics of the data being analyzed. By focusing on the most relevant hyperparameters to the specific application and data properties, we aimed to achieve a more efficient exploration of the most promising regions within the vast hyperparameter landscape.

Within the hyperparameter tuning process, the length of the sequence of historical data employed for predictions (history length) was constrained to not exceed the size of a single training batch. This deliberate limitation ensures that the model learns from a temporally cohesive data range that falls entirely within its immediate processing window. By maintaining consistency between the model's input size and the available training data, we aimed to enhance the model's capacity to effectively capture the relevant temporal dependencies within the historical data.

To maintain practical relevance to our forecasting task, we imposed a constraint on the prediction length. This limitation ensured that predicted values did not extend beyond the maximum number of data points present in the test dataset. By adhering to this constraint, we safeguarded against generating forecasts that surpassed the testable range, thereby precluding the generation of impractical and unverifiable results.

The batch size was meticulously chosen after careful consideration of both the volume and inherent structure of the employed datasets. This strategic selection aimed to achieve a well-balanced approach, optimizing the trade-off between computational efficiency and the model's learning stability. By establishing an appropriate batch size, we sought to avoid potential inefficiencies that could arise during the training process, such as slow training times or unstable learning dynamics.

The learning rate and dropout rate hyperparameters were initialized within established boundaries, commonly employed in machine learning. The learning rate search space was constructed using a logarithmic scale. This approach ensures adequate exploration across various orders of magnitude, crucial for identifying optimal learning rates that can effectively guide the model's optimization process. In contrast, the dropout rate search space was established with uniform spacing. This even distribution allows for a thorough exploration of the conventional range of dropout rates, enabling the identification of dropout values that effectively promote model generalization and mitigate overfitting.

The configuration of the neural network components, including the number of hidden layers and output dimensions, was constrained based on our practical observations and insights gleaned from experimentation with the model, but we also carefully considered the architectural recommendations outlined in the original TiDE paper. This approach allowed us to leverage established best practices for network design while simultaneously adapting the architecture to the specific characteristics and requirements of our data.

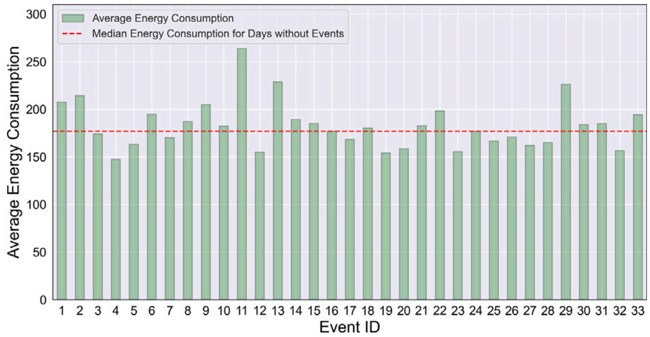

Fig. 2. Average energy consumption in the CC dataset for each event.

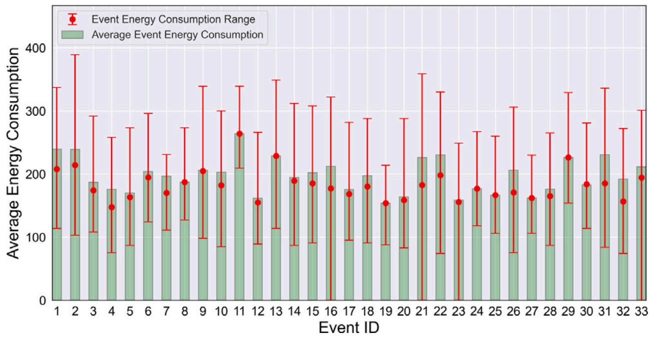

Fig. 3. Average energy consumption in the CC dataset for each event.

4. Results

4.1. Dataset description

The dataset contains energy consumption data, weather events, temperature, and events for the city of Sibiu, Romania, collected over 6 years from 2014 to 2019. Sibiu is a medieval city with a population of approximately 133 000, making it the 15th largest city in Romania. Geographically, the city is close to the mountains at an average height of 415 meters above sea level and has a humid continental climate. According to the Romanian statistical yearbook [32], in 2019 Sibiu had an average air temperature of 11.1 °C with a maximum of 34.4 °C and a minimum of −22.9 °C.

The dataset contains data with an hourly frequency and includes three distinct types of energy consumption data:

- Captive Consumers (CC): This category primarily consists of energy used by households and small-scale consumers.
- Own Technological Consumption (OTC): This category refers to the energy consumed by energy distribution and generation utilities for their internal operational needs, including maintenance and process energy requirements.
- CC_OTC: This composite metric represents the aggregate energy consumption, encompassing both the energy utilized by end consumers (CC) and the operational consumption by utilities (OTC).

4.2. Dataset analysis

Events, such as holidays, that affect energy consumption were recorded in the dataset as covariates. Table 9 lists all the events along with their IDs. In Fig. 2 we show the average energy consumption for each event in the dataset. The dotted red line shows the median energy consumption in common days without events, specifically 177 MWh. The top-5 events with the highest deviation in energy consumption from the baseline are:

- New Year's Day (1 January, event no. 11) with 85.28 MWh;
- New Year's Eve (31 December, event no. 13) with 50.12 MWh;
- Three Holy Hierarchs (30 January, event no. 29) with 47.49 MWh;
- Christmas Day (25 December, event no. 2) with 35.50 MWh;
- Ascension Day (variable date, 40 days after Orthodox Easter, event no. 4) with 31.19 MWh.

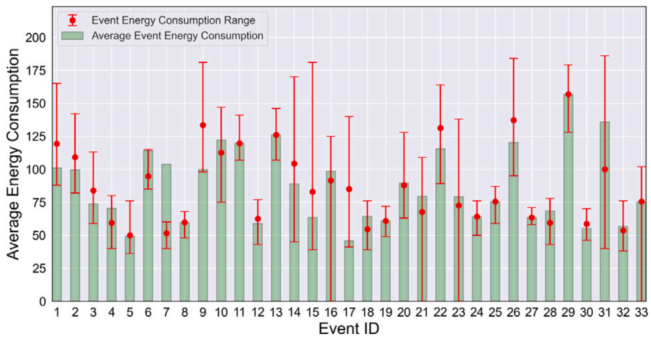

Fig. 4. Average energy consumption in the OTC dataset by relevant event.

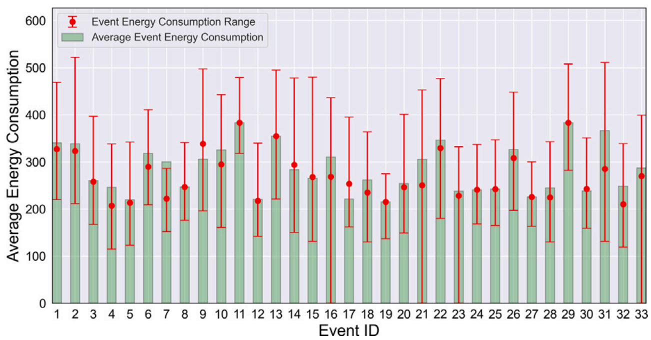

Fig. 5. Average energy consumption in the CC_OTC dataset by relevant event.

Figs. 3–5 depict, using the error bar, the minimum and maximum of the energy consumption for an event across the dataset, with the dot being the average. The average event consumption is calculated as the average of the 7 days window around the event, with the relevant event in the middle.

As can be seen from Figs. 3–5, in terms of the OTC, the minimum and maximum range for the event can vary a lot compared to the neighboring period resulting in an unpredictable jump or drop in consumption, which can contribute negatively to the prediction of the relevant event being used as a prediction factor. We can also note that in a lot of cases, while the range of the event varies a lot, the average of the event is often significantly below or above the average of the period resulting in a further deviation. This can be a cause of the deviations in the results when the events were added to the input data for predictions.

4.3. Data preprocessing

In this section of our study, we detail the methodology employed for structuring the dataset to facilitate subsequent analysis. The dataset contains a variety of features to comprehensively represent the factors influencing energy consumption, such as:

- Temporal *Feature*: The date column serves as the primary temporal feature. This column records the timestamp of data collection in the format *DD-MM-YY HH:MM*. This high-resolution temporal data is important for identifying consumption patterns and trends over time.
- *Numerical Features*: Energy consumption, minimum and maximum temperature readings, are quantified as numerical features. These features are essential for modeling the correlation between energy usage and temperature variations, providing a quantitative basis for assessing the impact of climatic conditions on energy demand.
- *Weather Events*: To capture the impact of specific weather events on energy consumption, each event is represented as a separate feature within the dataset. These features are encoded using a binary scheme, where "0" denotes the absence of an event and "1" indicates its occurrence. This encoding strategy simplifies the integration of weather events into our model, enabling the assessment of their individual and combined effects on energy consumption. There are a total of 16 different weather events, grouped into 2 categories: primary events such as "*rainy*", "*cloudy*", "*thermal inversion*", etc., and secondary events such as "*heavy rain*", "*partially cloudy*" etc.
- *Event Feature*: The representation of relevant events (cultural, religious, local, etc.) within the dataset is handled through a categorical feature, where each event is assigned a Unique Identifier (ID). This approach allows for the differentiation of events and the evaluation of their respective impacts on energy consumption patterns. The categorization of events as a distinct feature is essential for analyzing variations in energy demand during these periods, reflecting changes in social activity and public behavior. There are a total of 33 encoded events and 1 ID is reserved for no events (encoded as 0).

Table 2
Reduced hyperparameters space.

Hyperparameter	Values
batch_size	{256, 512, 1024}
learning_rate	{1e−5, 3e−4, 1e−4, 1e−3, 1e−2}
dropout_rate	{0.0, 0.1, 0.2, 0.3, 0.4, 0.5}
layer_norm	{True, False}
hidden_size	{256, 512, 1024}
decoder_output_dim	{4, 8, 16, 32}
final_decoder_hidden	{32, 64, 128}
hist_len	{96, 192, 336, 720}
pred_len	{96, 192, 336, 720}

To facilitate effective learning and improve the algorithm's performance, the numerical features of the data were normalized using Standard Scaler from the scikit-learn library. This rescales the features to have zero mean and unit variance. The normalization was performed based on the statistics of the training dataset, where μ is the mean and σ is the standard deviation:

$$X_{normalized} = \frac{X - \mu}{\sigma}$$

Finally, the datasets were split into three subsets: 70% for the training set, 10% for the validation set, and 20% for the testing set. This convention was set by several published papers. This split ensures adequate data for learning while allowing for effective validation and testing.

4.4. Hyperparameters tuning

The strategic application of dimensionality reduction techniques yielded a significant decrease in the size of the hyperparameter space. The initial staggering number of 208 million possible combinations was effectively reduced to a considerably more manageable size of 103 680. This dramatic reduction from eight orders of magnitude to five underscores the effectiveness of the implemented rules. By focusing on the most relevant hyperparameter settings within the context of the model's application and data, we were able to significantly simplify the hyperparameter tuning process, rendering it computationally feasible and enabling a more efficient exploration of the most promising regions within the search space defined by the values from Table 2.

4.5. Model configuration

Our prediction model is trained from scratch over a span of maximum 20 epochs using Adam optimizer and Mean Squared Error (MSE) as loss function:

$$MSE\left(\left\{y_{L+1:L+H}^{(i)}\right\}_{i=1}^{N}, \left\{\hat{y}_{L+1:L+H}^{(i)}\right\}_{i=1}^{N}\right) = \frac{1}{NH}\sum_{i=1}^{N} \left\| y_{L+1:L+H}^{(i)} - \hat{y}_{L+1:L+H}^{(i)} \right\|_2^2$$

where:

- $y_{L+1:L+H}^{(i)}$ is the true value of the time series from time $L+1$ to $L+H$ for the ith example in the dataset.
- $\hat{y}_{L+1:L+H}^{(i)}$ is the predicted value of the time series from time $L+1$ to $L+H$ for the ith example.
- N is the number of examples in the dataset.
- H is the forecasting horizon, the number of time steps we are predicting into the future.
- $\| y_{L+1:L+H}^{(i)} - \hat{y}_{L+1:L+H}^{(i)} \|_2^2$ denotes the squared Euclidean norm (L2 norm).

During training, the model iterates over the training data in each epoch, computing the MSE loss on a batch-by-batch basis. The model weights are updated using the backpropagation algorithm. An early stopping mechanism was implemented based on a patience parameter (set to 5 epochs). If the validation loss did not improve for the specified number of epochs, training was halted to prevent overfitting. Additionally, checkpoints of the model were saved whenever it achieved a better performance than the previous one on the validation set. Predictions for the validation and test sets on each checkpoint were also saved for subsequent analysis or post-processing.

4.6. Model evaluation

We used the Mean Absolute Error (MAE) to evaluate and compare our model with previous approaches to forecasting electricity consumption.

$$MAE\left(y_{pred}, y_{true}\right) = \frac{1}{n}\sum_{i=1}^{n} \left| y_{pred,i} - y_{true,i} \right|$$

Table 3
Results of next hour prediction (MAE in MWh).

Method	CC	OTC	CC_OTC	Average
Fuzzy Controller	6.80	3.05	8.63	6.16
TBATS	4.64	1.46	5.45	3.85
LSTM	6.78	2.09	8.14	5.67
Markov	10.36	4.18	12.73	9.09
TiDE	**3.44**	**0.89**	**3.24**	**2.52**

Table 4
Optimal hyperparameter values and network configuration for next hour prediction.

Hyperparameter	CC	OTC	CC_OTC
batch_size	521	256	1024
learning_rate	0.0001	0.0003	0.0001
dropout_rate	0.1	0.4	0.2
layer_norm	FALSE	FALSE	FALSE
hidden_size	1024	512	512
decoder_output_dim	8	4	8
final_decoder_hidden	32	128	64
hist_len	720	96	96
pred_len	1	1	1

4.6.1. Next hour forecasting

In this section, we initially focused on conducting next-hour energy consumption predictions on our test set. This was aimed at benchmarking the performance of various established predictive methods against our proposed model, specifically within the context of short-term forecasting.

The analysis of next-hour energy consumption predictions reveals meaningful insights, particularly when assessing the performance of our proposed method against other widely used forecasting models. The MAE, measured in MegaWatt-hours (MWh), serves as the metric for evaluating predictive accuracy across different methods, including Fuzzy Controller, TBATS, LSTM, and Markov models.

As presented in Table 3, our method outperformed the competing approaches, achieving an average MAE of 2.52 MWh, indicating a high level of precision in its predictions. Compared to the next best model, TBATS, which had an average MAE of 3.85 MWh, TiDE demonstrated a 34.55% improvement in predictive accuracy, underscoring its superior capability in short-term energy consumption forecasting. The optimal hyperparameter values, along with the detailed network configuration for next hour prediction are detailed in Table 4.

4.6.2. Next month forecasting

In the subsequent phase of our study, the focus shifted toward evaluating the long-term forecasting capabilities of our model through next month (30 days) predictions, where all the 720 (30 days * 24 h) data points were forecasted simultaneously. For this analysis, the TiDE model, identified as the most effective method from our initial short-term forecasting comparison, was exclusively used to examine prediction performance over an extended horizon.

The evaluation aimed not only to assess TiDE's ability in long-term energy consumption forecasting but also to explore the impact of incorporating additional training data beyond historical energy consumption records. To comprehensively understand how each different type of auxiliary data influences forecasting accuracy, separate tests were conducted including weather events, minimum and maximum temperature readings, and events. Afterward, we trained and evaluated a model that combined all these data types to observe the cumulative effect on the model's performance. This approach allowed for a detailed analysis of the impact each data category had on the forecasting model.

We applied the optimal hyperparameters identified during the next month of forecasting experiments across all tests. This approach ensures that any observed differences in model performance are attributable to the data inputs rather than variations in model configuration.

To address the variability observed across training runs, each model was trained 3 times. The results were then aggregated, and the reported values from Table 5 represent the mean performance accompanied by the standard error, providing a more reliable and statistically sound measure of each model's performance, while Table 6 presents the optimal hyperparameters and network configuration for next month prediction across all experiments.

4.6.3. Statistical analysis of forecasting results

In our scientific exploration into the variations of MAE across different forecasting methods and datasets, a factorial Analysis of Variance (ANOVA) was conducted. In Table 7 we present several statistical aspects. The "Source" column classifies the variance origins into different methods, with the "Residual" row capturing variance unexplained by these categories. The "Sum of Squares" column quantifies the variation each source contributes to MAE, with larger values indicating greater contributions. Degrees of Freedom ("df" column), dependent on the number of categories, shows values free to vary in each source, including residuals

Table 5
Results of next month prediction (MAE in MWh).

Method	CC	OTC	CC_OTC	Average
TiDE + Weather Events	3.96 ± 0.09	1.26 ± 0.05	5.42 ± 0.02	3.55 ± 0.05
TiDE + Temperature	4.38 ± 0.05	1.20 ± 0.01	5.63 ± 0.02	3.74 ± 0.03
TiDE + Events	5.69 ± 0.03	1.16 ± 0.03	7.04 ± 0.04	4.63 ± 0.03
TiDE + All	4.13 ± 0.15	1.22 ± 0.08	**5.06 ± 0.03**	3.47 ± 0.09
TiDE	**3.86 ± 0.05**	**1.09 ± 0.07**	5.40 ± 0.03	**3.45 ± 0.05**

Table 6
Optimal hyperparameter values and network configuration for next month prediction.

Hyperparameter	CC	OTC	CC_OTC
batch_size	1024	256	1024
learning_rate	0.001	0.001	0.0001
dropout_rate	0.4	0.4	0
layer_norm	FALSE	FALSE	FALSE
hidden_size	256	512	512
decoder_output_dim	16	8	16
final_decoder_hidden	32	128	32
hist_len	720	720	720
pred_len	720	720	720

Table 7
ANOVA test results.

Source	Sum of Squares	df	F-value	p-value
Method	12.01	4	748.95	< 0.0001
Dataset	164.99	3	13 727.99	< 0.0001
Method:Dataset	5.12	12	106.52	< 0.0001
Residual	0.16	40	–	–

for unexplained variance. The F-value assesses the ratio of variance explained by groups to that within groups, with higher values indicating significant impacts on MAE. Lastly, the p-value evaluates the likelihood of observing the F-value under the null hypothesis, with values less than 0.05 indicating strong evidence of significant group differences.

The test results highlight the statistical significance of differences in the MAE when comparing various TiDE enhancements across different prediction categories (CC, OTC, CC_OTC). For instance, the "Method" factor, with an F-value of 748.95, and a p-value of less than 0.0001, illustrates the significant impact that different TiDE data enhancements (such as weather events, temperature, events, and their combination) have on prediction accuracy. This confirms that each method variant influences the error margin distinctly.

Similarly, for the "Dataset" factor, the variation among datasets is big, suggesting that some datasets are more sensitive to certain types of data than others.

The interaction between the "Method" and "Dataset" factors (Method:Dataset), with an F-value of 106.52, and a p-value of less than 0.0001 also shows significant differences. This indicates that the effectiveness of a TiDE data enhancement can vary depending on which category it is applied to, highlighting the necessity of tailoring the forecasting approach to the specific characteristics of each dataset.

Finally, the low residual sum of squares (0.16) implies that the model, with the inclusion of all these factors, accounts for most of the variability in MAE, leaving only a small portion of the variation unaccounted for. This suggests a good fit of the model to the data, reflecting that our analysis captures the primary influences on prediction accuracy effectively.

Given the significant ANOVA results, Tukey's Honest Significant Difference (HSD) post-hoc tests were performed to identify which specific group differences are statistically significant. The detailed results are presented in Table 8.

The baseline TiDE method produced the most consistent results, with the OTC category benefiting most from the unenhanced approach, as indicated by its lowest MAE.

When weather events were factored in, there was no significant improvement, suggesting that the weather data does not contribute to the model's general predictability. However, the incorporation of temperature data led to a slight performance decrease in all datasets, implying that temperature data might introduce noise or cause overfitting in our context. Event data inclusion resulted in a significant decrease in MAE across all categories.

The combination of all data types (TiDE + All) showed a mixed impact: it slightly increased the MAE for CC, yet improved it for CC_OTC, suggesting that while a broad dataset can be beneficial, it may not uniformly enhance model performance across different prediction categories.

Hence, to achieve optimal performance in forecasting energy consumption, our analysis suggests employing the TiDE method focused solely on historical energy consumption data, without any additional data enhancements. This streamlined approach appears to yield the most accurate predictions across most datasets, as indicated by the lowest MAE values.

Table 8
Statistically significant results from Tukey's HSD tests.

Dataset	Method Comparison	Mean difference	Adjusted p-value	Confidence interval
CC	TiDE vs. TiDE + All	0.4233	0.0006	0.2031 to 0.6434
CC	TiDE vs. TiDE + Events	1.8605	< 0.0001	1.6404 to 2.0806
CC	TiDE vs. TiDE + Temperature	0.5627	0.0001	0.3425 to 0.7828
CC	TiDE + All vs. TiDE + Events	1.4372	< 0.0001	1.2171 to 1.6574
CC	TiDE + All vs. TiDE + Weather Events	−0.3056	0.0071	−0.5257 to −0.0855
CC	TiDE + Events vs. TiDE + Temperature	−1.2978	< 0.0001	−1.518 to −1.0777
CC	TiDE + Events vs. TiDE + Weather Events	−1.7428	< 0.0001	−1.963 to −1.5227
CC	TiDE + Temperature vs. TiDE + Weather Events	−0.445	0.0004	−0.6651 to −0.2249
OTC	TiDE vs. TiDE + All	0.1396	0.0258	0.016 to 0.2631
OTC	TiDE vs. TiDE + Temperature	0.1733	0.0066	0.0498 to 0.2968
OTC	TiDE vs. TiDE + Weather Events	0.2334	0.0007	0.1098 to 0.3569
CC_OTC	TiDE vs. TiDE + All	−0.3117	< 0.0001	−0.4163 to −0.207
CC_OTC	TiDE vs. TiDE + Events	1.7057	< 0.0001	1.601 to 1.8103
CC_OTC	TiDE vs. TiDE + Temperature	0.2404	0.0001	0.1358 to 0.345
CC_OTC	TiDE + All vs. TiDE + Events	2.0173	< 0.0001	1.9127 to 2.122
CC_OTC	TiDE + All vs. TiDE + Temperature	0.5521	< 0.0001	0.4474 to 0.6567
CC_OTC	TiDE + All vs. TiDE + Weather Events	0.3437	< 0.0001	0.2391 to 0.4483
CC_OTC	TiDE + Events vs. TiDE + Temperature	−1.4653	< 0.0001	−1.5699 to −1.3606
CC_OTC	TiDE + Events vs. TiDE + Weather Events	−1.6736	< 0.0001	−1.7783 to −1.569
CC_OTC	TiDE + Temperature vs. TiDE + Weather Events	−0.2084	0.0005	−0.313 to −0.1037
Average	TiDE vs. TiDE + Events	1.1728	< 0.0001	0.9702 to 1.3754
Average	TiDE vs. TiDE + Temperature	0.3142	0.0033	0.1116 to 0.5168
Average	TiDE + All vs. TiDE + Events	1.1591	< 0.0001	0.9565 to 1.3617
Average	TiDE + All vs. TiDE + Temperature	0.3006	0.0045	0.098 to 0.5031
Average	TiDE + Events vs. TiDE + Temperature	−0.8586	< 0.0001	−1.0612 to −0.656
Average	TiDE + Events vs. TiDE + Weather Events	−1.0292	< 0.0001	−1.2318 to −0.8266

Exceptionally, for the CC_OTC dataset, the incorporation of all data enhancements significantly enhances the model's performance. In this particular scenario, the comprehensive approach that leverages a multifaceted array of data points surpasses the accuracy of the baseline TiDE method. This suggests that when predicting across a merged dataset that spans multiple categories, a more intricate model that utilizes a broader spectrum of data can more effectively capture the complex dynamics at play in energy consumption patterns.

These findings emphasize the need for careful consideration of data relevance to the predictive model's context, as indiscriminate inclusion of variables may lead to suboptimal outcomes.

5. Conclusions and further work

In this paper, we showed that a hyperparameter-optimized TiDE model provides significant improvements in prediction error compared to LSTM or traditional statistical models for next-hour electricity consumption forecasting. The comparative study of additional features' (weather, events, temperature) influence on the prediction results demonstrates that different consumption components (CC, OTC, CC_OTC) of an electricity supply station require different models. When looking at next month forecasting, simpler electricity consumption curves like CC and OTC are better forecasted with a simple model, while the CC_OTC required all the three additional features included as input to achieve the best results.

Going forward, we want to evaluate the time series dense encoder method on other use cases, but also at different scales. For smart houses equipped with renewable energy production and storage equipment, a time series forecasting model like TiDE could provide valuable information in terms of both electricity consumption and production that can further be used to drive the decision-making process required to optimize the grid electricity consumption. This paper's case study is the electricity supplier of a small city that requires an hour-ahead and a month-ahead forecast, but other electricity distributors and suppliers have a larger customer base and might require different forecast intervals.

In conclusion, the integration of smart grids within the framework of smart cities represents a synergistic approach toward advancing urban sustainability, resilience, and efficiency. By harnessing the capabilities of smart grid technologies, cities can foster innovation, enhance the quality of life, and promote sustainable growth, ultimately contributing to the creation of more liveable and resilient urban environments.

CRediT authorship contribution statement

Darius Peteleaza: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Alexandru Matei:** Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Radu Sorostinean:** Writing – review & editing, Writing – original draft, Visualization, Validation, Investigation, Formal analysis, Data curation. **Arpad Gellert:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Resources,

Table 9
Mappings of events and their ID.

Event type	Event	Date	ID
–	No Event	–	0
Religious	Saint Stephen's Day	December 27th	1
Religious	Christmas	December 25th	2
Religious	Saint George	April 23rd	3
Religious	Ascension of Jesus	40 days after Easter	4
Religious	Pentecost	50 days after Easter	5
National	General Elections	Variable	6
Religious	Saint Peter's Day	June 29th	7
Religious	Transfiguration of Jesus	August 6th	8
Religious	Epiphany	January 6th	9
Religious	The Entrance of the Virgin Mary into the Temple	November 21st	10
International	New Year's Day	January 1st	11
Religious	The Exaltation of the Holy Cross	September 14th	12
International	New Year's Eve	December 31st	13
Weather	Thermal Inversion	–	14
National	Public Sector Holiday	Variable	15
Religious	The Annunciation	March 25th	16
–	Other	–	17
International	Labor Day	May 1st	18
Local	Sibiu International Theater Festival	Variable (typically in June)	19
Religious	Saint Parascheva's Day	October 14th	20
Religious	Orthodox Easter	Variable (first Sunday after the first full moon occurring on or after the vernal equinox)	21
National	Romanian National Day	December 1st	22
International	Daylight Saving Time Change	Variable (last Sunday in March and last Sunday in October)	23
Religious	Midsummer Day	June 24th	24
Religious	The Life-Giving Spring of the Mother of God	First Friday after Easter	25
Religious	Saint Nicholas's Day	December 6th	26
Religious	Saint Constantine's Day	May 21st	27
Religious	Saint Helen's Day	May 21st	28
Religious	The Three Holy Hierarchs	January 30	29
Religious	Saints Joachim and Anne's Day	September 9th	30
Religious	Saint John's Day	January 7th	31
Religious	The Dormition of the Mother of God	August 15th	32
Religious	Palm Sunday	The Sunday before Easter	33

Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Ugo Fiore:** Writing – review & editing, Writing – original draft, Validation. **Bala-Constantin Zamfirescu:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Francesco Palmieri:** Writing – review & editing, Writing – original draft, Validation.

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

Data will be made available on request.

Acknowledgments

This work was financed by Lucian Blaga University of Sibiu, Romania (Knowledge Transfer Center) & Hasso Plattner Foundation research grant LBUS-HPI-ERG-2023-03.

References

[1] A. Das, W. Kong, A. Leach, S.K. Mathur, R. Sen, R. Yu, Long-term forecasting with TiDE: Time-series dense encoder, Trans. Mach. Learn. Res. (2023) http://dx.doi.org/10.48550/arXiv.2304.08424.

[2] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints, IEEE Trans. Evol. Comput. 18 (4) (2014) 577–601, http://dx.doi.org/10.1109/TEVC.2013.2281535.

[3] H. Jain, K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach, IEEE Trans. Evol. Comput. 18 (4) (2014) 602–622, http://dx.doi.org/10.1109/TEVC.2013.2281534.

[4] O.B. Sezer, M.U. Gudelek, A.M. Ozbayoglu, Financial time series forecasting with deep learning : A systematic literature review: 2005–2019, Appl. Soft Comput. 90 (2020) 106181, http://dx.doi.org/10.1016/j.asoc.2020.106181.

[5] A.Y. Barrera-Animas, L.O. Oyedele, M. Bilal, T.D. Akinosho, J.M.D. Delgado, L.A. Akanbi, Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting, Mach. Learn. Appl. 7 (2022) 100204, http://dx.doi.org/10.1016/j.mlwa.2021.100204.

[6] J. Wang, C. Xu, J. Zhang, R. Zhong, Big data analytics for intelligent manufacturing systems: A review, J. Manuf. Syst. 62 (2022) 738–752, http://dx.doi.org/10.1016/j.jmsy.2021.03.005.

[7] A. Zeroual, F. Harrou, A. Dairi, Y. Sun, Deep learning methods for forecasting COVID-19 time-series data: A comparative study, Chaos Solitons Fractals 140 (2020) 110121, http://dx.doi.org/10.1016/j.chaos.2020.110121.
[8] G.-F. Fan, X. Wei, Y.-T. Li, W.-C. Hong, Forecasting electricity consumption using a novel hybrid model, Sustainable Cities Soc. 61 (2020) 102320, http://dx.doi.org/10.1016/j.scs.2020.102320.
[9] D. Hadjout, J.F. Torres, A. Troncoso, A. Sebaa, F. Martínez-Álvarez, Electricity consumption forecasting based on ensemble deep learning with application to the Algerian market, Energy 243 (2022) 123060, http://dx.doi.org/10.1016/j.energy.2021.123060.
[10] T. Pinto, I. Praça, Z. Vale, J. Silva, Ensemble learning for electricity consumption forecasting in office buildings, Neurocomputing 423 (2021) 747–755, http://dx.doi.org/10.1016/j.neucom.2020.02.124.
[11] F. Lazzari, G. Mor, J. Cipriano, E. Gabaldon, B. Grillone, D. Chemisana, F. Solsona, User behaviour models to forecast electricity consumption of residential customers based on smart metering data, Energy Rep. 8 (2022) 3680–3691, http://dx.doi.org/10.1016/j.egyr.2022.02.260.
[12] S. Ghimire, R.C. Deo, D. Casillas-Pérez, S. Salcedo-Sanz, Efficient daily electricity demand prediction with hybrid deep-learning multi-algorithm approach, Energy Convers. Manage. 297 (2023) 117707, http://dx.doi.org/10.1016/j.enconman.2023.117707.
[13] Z. Wang, T. Hong, H. Li, M. Ann Piette, Predicting city-scale daily electricity consumption using data-driven models, Adv. Appl. Energy 2 (2021) 100025, http://dx.doi.org/10.1016/j.adapen.2021.100025.
[14] M.H.L. Lee, Y.C. Ser, G. Selvachandran, P.H. Thong, L. Cuong, L.H. Son, N.T. Tuan, V.C. Gerogiannis, A comparative study of forecasting electricity consumption using machine learning models, Mathematics 10 (8) (2022) 1329, http://dx.doi.org/10.3390/math10081329.
[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc., 2017, [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
[16] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, L. Sun, Transformers in time series: A survey, in: Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, Macau, SAR China: International Joint Conferences on Artificial Intelligence Organization, 2023, pp. 6778–6786, http://dx.doi.org/10.24963/ijcai.2023/759.
[17] H. Wu, J. Xu, J. Wang, M. Long, Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting, in: M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, J.W. Vaughan (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc., 2021, pp. 22419–22430, [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2021/file/bcc0d400288793e8bdcd7c19a8ac0c2b-Paper.pdf.
[18] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, W. Zhang, Informer: Beyond efficient transformer for long sequence time-series forecasting, in: AAAI, Vol. 35, No. 12, 2021, pp. 11106–11115, http://dx.doi.org/10.1609/aaai.v35i12.17325.
[19] B.N. Oreshkin, D. Carpov, N. Chapados, Y. Bengio, N-BEATS: Neural basis expansion analysis for interpretable time series forecasting, in: International Conference on Learning Representations, 2020, [Online]. Available: https://openreview.net/forum?id=r1ecqn4YwB.
[20] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, R. Jin, FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting, in: K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, S. Sabato (Eds.), Proceedings of the 39th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 162, PMLR, 2022, pp. 27268–27286, [Online]. Available: https://proceedings.mlr.press/v162/zhou22g.html.
[21] A. Zeng, M. Chen, L. Zhang, Q. Xu, Are transformers effective for time series forecasting? in: AAAI, Vol. 37, No. 9, 2023, pp. 11121–11128, http://dx.doi.org/10.1609/aaai.v37i9.26317.
[22] Y. Nie, N.H. Nguyen, P. Sinthong, J. Kalagnanam, A time series is worth 64 words: Long-term forecasting with transformers, in: International Conference on Learning Representations, 2023, http://dx.doi.org/10.48550/arXiv.2211.14730.
[23] N. Shahzadi, N. Javaid, M. Akbar, A. Aldegheishem, N. Alrajeh, S.H. Bouk, A novel data driven approach for combating energy theft in urbanized smart grids using artificial intelligence, Expert Syst. Appl. 253 (2024) 124182, http://dx.doi.org/10.1016/j.eswa.2024.124182.
[24] Z. Aslam, N. Javaid, M.U. Javed, M. Aslam, A. Aldegheishem, N. Alrajeh, A new clustering-based semi-supervised method to restrict the users from anomalous electricity consumption: supporting urbanization, Electr. Eng. (2024) http://dx.doi.org/10.1007/s00202-024-02362-3.
[25] A. Naeem, N. Javaid, Z. Aslam, M.I. Nadeem, K. Ahmed, Y.Y. Ghadi, T.J. Alahmadi, N.A. Ghamry, S.M. Eldin, A novel data balancing approach and a deep fractal network with light gradient boosting approach for theft detection in smart grids, Heliyon 9 (9) (2023) http://dx.doi.org/10.1016/j.heliyon.2023.e18928.
[26] A. Gellert, L.-M. Olaru, A. Florea, I.-I. Cofaru, U. Fiore, F. Palmieri, Estimating electricity consumption at city-level through advanced machine learning methods, Connect. Sci. 36 (1) (2024) 2313852, http://dx.doi.org/10.1080/09540091.2024.2313852.
[27] A. Gellert, A. Florea, U. Fiore, F. Palmieri, P. Zanetti, A study on forecasting electricity production and consumption in smart cities and factories, Int. J. Inf. Manage. 49 (2019) 546–556, http://dx.doi.org/10.1016/j.ijinfomgt.2019.01.006.
[28] M.-A. Bachici, A. Gellert, Modeling electricity consumption and production in smart homes using LSTM networks, Int. J. Adv. Stat. IT & C Econ. Life Sci. 10 (1) (2020) 80–89, http://dx.doi.org/10.2478/ijasitels-2020-0009.
[29] L.M. Olaru, A. Gellert, U. Fiore, F. Palmieri, Electricity production and consumption modeling through fuzzy logic, Int. J. Intell. Syst. 37 (11) (2022) 8348–8364, http://dx.doi.org/10.1002/int.22942.
[30] A. Gellert, U. Fiore, A. Florea, R. Chis, F. Palmieri, Forecasting electricity consumption and production in smart homes through statistical methods, Sustainable Cities Soc. 76 (2022) 103426, http://dx.doi.org/10.1016/j.scs.2021.103426.
[31] S. Wietheger, B. Doerr, A mathematical runtime analysis of the non-dominated sorting genetic algorithm III (NSGA-III), in: Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, Macau, SAR China: International Joint Conferences on Artificial Intelligence Organization, 2023, pp. 5657–5665, http://dx.doi.org/10.24963/ijcai.2023/628.
[32] Institutul Naţional de Statistică, 2020 Romanian statistical yearbook, 2021, [Online]. Available: https://insse.ro/cms/sites/default/files/field/publicatii/anuarul_statistic_al_romaniei_carte_1.pdf. (Accessed 11 April 2024).