

Hybrid transformer model with liquid neural networks and learnable encodings for buildings' energy forecasting

Gabriel Antonesi^a, Tudor Cioara^{a,*}, Ionut Anghel^a, Ioannis Papias^b,
Vasilis Michalakopoulos^b, Elissaios Sarmas^b

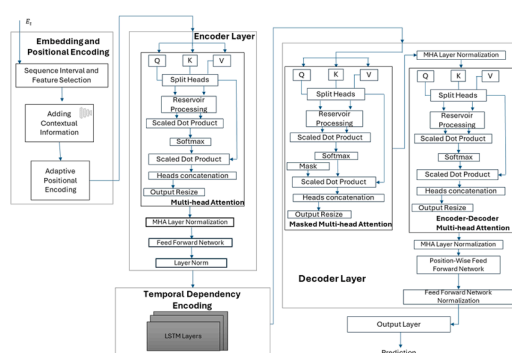
^a Distributed Systems Research Laboratory, Computer Science Department, Technical University of Cluj-Napoca, G. Barițiu 26-28, 400027 Cluj-Napoca, Romania

^b Decision Support Systems Laboratory, School of Electrical & Computer Engineering, National Technical University of Athens, Ir. Politechniou 9, 157 73 Athens, Greece

HIGHLIGHTS

- Hybrid transformer with liquid neural networks model for building energy forecasting.
- Convolutional neural network encoders to understand temporal dynamics in energy data through spatial mappings.
- Reservoir processing module implemented with liquid neural networks to capture non-linear relations in energy data.
- Validation on various building contexts, including large apartment buildings and small households.

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:

Residential building
Commercial building
Households
Energy forecasting
Transformer model
Liquid neural network

ABSTRACT

Accurate forecasting of buildings' energy demand is essential for building operators to manage loads and resources efficiently, and for grid operators to balance local production with demand. However, nowadays models still struggle to capture nonlinear relationships influenced by external factors like weather and consumer behavior, assume constant variance in energy data over time, and often fail to model sequential data. To address these limitations, we propose a hybrid Transformer-based model with Liquid Neural Networks and learnable encodings for building energy forecasting. The model leverages Dense Layers to learn non-linear mappings to create embeddings that capture underlying patterns in time series energy data. Additionally, a Convolutional Neural Network encoder is integrated to enhance the model's ability to understand temporal dynamics through spatial mappings. To address the limitations of classic attention mechanisms, we implement a reservoir processing module using Liquid Neural Networks which introduces a controlled non-linearity through dynamic reservoir computing, enabling the model to capture complex patterns in the data. For model evaluation, we utilized both pilot data and state-of-the-art datasets to determine the model's performance across various building contexts, including large apartment and commercial buildings and small households, with and without on-site energy production. The proposed transformer model demonstrates good predictive accuracy and training time efficiency across various types of buildings and testing configurations. Specifically, SMAPE scores indicate a

* Corresponding author.

E-mail addresses: gabriel.antonesei@cs.utcluj.ro (G. Antonesei), tudor.cioara@cs.utcluj.ro (T. Cioara), ionut.anghel@cs.utcluj.ro (I. Anghel), jpapias@epu.ntua.gr (I. Papias), vmichalakopoulos@epu.ntua.gr (V. Michalakopoulos), esarmas@epu.ntua.gr (E. Sarmas).

<https://doi.org/10.1016/j.egyai.2025.100489>

Available online 18 February 2025

2666-5468/© 2025 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

reduction in prediction error, with improvements ranging from 1.5 % to 50 % over basic transformer, LSTM and ANN models while the higher R^2 values further confirm the model's reliability in capturing energy time series variance. The 8 % improvement in training time over the basic transformer model, highlights the hybrid model computational efficiency without compromising accuracy.

1. Introduction

The decarbonization of the smart grid requires not only the large-scale deployment of renewable energy sources but also better planning and demand-side management [1,2]. Therefore, the grid operators have put in place demand management programs to encourage consumers to reduce or shift energy usage during peak periods. As a significant amount of the energy demand is attributed to the buildings, efforts must be committed to increase their interoperability with the smart grid and transform them into active players participating in demand response programs [3,4]. However, to make this vision possible it is important to have accurate predictions of the building's energy demand. They will enable the building operators to anticipate the load and efficiently allocate resources to achieve cost reductions, and the system operators to plan and balance the local production with the demand [5].

Statistical methods such as Autoregressive Integrated Moving Average (ARIMA) and its derivatives rely only on the past energy values making them relatively easy to interpret and effective in linear contexts [6–8]. However, they have problems in capturing nonlinear relationships in energy consumption patterns, which can be influenced, among others, by external factors such as weather variability and consumer behavior. Additionally, these models assume that the variance in energy data remains constant over time, which may not always be true. Therefore, machine learning models have been used for energy predictions as they excel in capturing complex, nonlinear patterns in data and may easily integrate external variables such as weather data [9]. Energy prediction models based on Support Vector Machines (SVMs) [10,11] decision trees [12] and Random Forests [13,14] have been successful for short-term load forecasting, offering improved generalization compared to autoregression ones. However, these models cannot often model sequential energy data as effectively as deep learning techniques.

Deep learning models can handle multiple inputs, (i.e. time series energy data, weather, consumer behavior), without the need for manual feature engineering while addressing the challenge of sequential data processing (e.g. Recurrent Neural Networks - RNNs) [15,16]. However, RNNs often suffer from the vanishing gradient problem, limiting their effectiveness over long energy data sequences [17,18]. Long Short-Term Memory (LSTM) models incorporate gating mechanisms to capture long-term dependencies in energy data more effectively showing better performance in energy demand prediction and capturing temporal dynamics like daily and seasonal variations [17]. Liu et al. [19] showed that LSTM models achieve greater accuracy in electricity load forecasting compared to traditional RNNs. Similarly, Bouktif et al. [20] enhance forecasting precision by employing an LSTM model that incorporates optimally selected time-lagged features. Sarma et al. [21] successfully developed and fine-tuned four distinct LSTM models to accurately predict the output of three rooftop PV plants in Portugal. Additionally, researchers in Michalakopoulos et al. [22] optimized LSTM architectures using the Optuna framework for hyperparameter tuning to improve the performance across various PV scenarios.

Recently Transformer models have shown promise in overcoming these challenges by using self-attention mechanisms to weigh the importance of different parts of the input energy data sequence, allowing them to capture relationships between distant data elements more effectively [23–25]. The multi-head attention mechanism proved to be successful in handling complex time-series data scenarios, such as wind and solar power forecasting with improved accuracy [26]. Zhou et al. applied this concept in the context of energy forecasting [27] and their

Informer model improves the efficiency of Transformers by using ProbSparse attention. This enables scalable forecasting for large-scale energy datasets without getting an accuracy penalty. Additionally, they had introduced positional encoding to make them better suited for time series tasks. The multi-head attention mechanism can handle long-term dependencies better than traditional RNN-based models like LSTM and GRU, obtaining higher accuracy [28–30]. Also, attention mechanisms have been applied to data imputation tasks, being able to keep temporal dynamics and improve forecasting accuracy [31]. Furthermore, adaptive models using Principal Component Analysis for dimensionality reduction before passing data into the Transformer model have been shown to focus only on the most informative features [32]. To enhance the interpretability of the model, the Interformer [33] was proposed for probabilistic short-term forecasting for the residential load. By incorporating sparse self-attention, the model reduces the computational complexity, while still providing important information about the variables that have the highest impact on the predictions. For wind energy generation forecasting, Li et al. [34] have shown that integrating the Transformer's self-attention mechanism with causal convolution and empirical mode decomposition improves the forecasting accuracy by capturing the local dependencies and the sharp variations in wind power data. Similarly, Harrou et al. [35] integrate a Variational Autoencoder with a self-attention mechanism for both the encoder and decoder layers. The model enhances the short-term wind power prediction accuracy by focusing on the most informative features provided in the input and using a nonlinear approximation in modelling the energy time series. Yu et al. integrated a dynamic self-attention mechanism (i.e., Temporal Fusion Transformer) into a solar power forecasting model to improve short-term predictions [36]. The attention mechanism's dynamic nature enables the model to gradually modify its focus on relevant features, hence assisting in the management of the various trends monitored in energy generation prediction.

However, the Transformers are more computationally complex than traditional neural network models especially as the time series sequence length and model size increase. To reduce the computational complexity, adaptive mechanisms for encoders, decoders, and attention mechanisms have been introduced. Mo et al. [37] use sparse self-attention and temporal pooling to improve computational efficiency, while not compromising the prediction accuracy. They use LSTM-based embedding to address the weak temporal prediction capabilities of the transformer model. Adaptive attention mechanisms have also been introduced to handle fluctuations in energy data and they perform well in terms of medium-term wind power forecasting [38]. Similarly, models that optimize the attention mechanism for multi-horizon, multi-source load forecasting in integrated energy systems, have been shown to improve prediction accuracy [39]. Wang et al. [25] proposed a cross-entity attention mechanism and shared variable selection networks to handle relationships between different entities in a smart grid (i.e., buildings, single prosumers, etc.). The mechanism can learn relationships among entities to inform the decoder about which entity the encoder should focus on. Qu et al. [40] proposed a transformer model with three key features: better long-distance dependence than LSTM, a multi-headed attention mechanism, and parallel computation for faster training. The model has been successfully applied to forecast the power output of multiple wind farms showing better accuracy compared to LSTM-based models. Dai et al. [41] proposed the use of self-attention-based neural network to capture the temporal relations in power sequences. Such models have shown good adaptation capabilities to real-time changes in wind power data being able to handle

ultra-short-term forecasting. Wu et al. [42], proposed Autoformer which adds a decomposition block to describe the trend and seasonal components in time-series data, is another significant example. By dividing the input data into distinct components, the Autoformer improves forecasting accuracy for solar and wind energy.

To further improve the model temporal relationships in sequential energy data, generalization capabilities and reduce the computational complexity concerning the sequence length, recent studies have explored the combination of Transformers with other architectural models like CNNs and GRUs. Tian et al. [43] combine LSTM with Informer to improve the representation capabilities in the case of time series and a parallel encoder module to improve the robustness. A hybrid CNN-GRU-Transformer model has been shown to surpass traditional models in terms of accuracy in solar power forecasting [44]. Also, a hybrid CNN-Transformer was proposed using the CNN abilities to extract spatial features and combine them with the Transformer's attention mechanism [45]. This hybrid approach shows promising results in wind power forecasting, improving both accuracy and computational resource use. Graph-based models have also been employed to capture spatial dependencies between different wind farms [46,47]. For instance, a Graph-Convolution Network (GCN) combined with Multi-Resolution Convolution Networks (MRCN) for short-term wind power prediction resulted in higher accuracy, compared to standalone CNN or RNN models. The Graph Patch Informer (GPI) extends the Transformer model by integrating Graph Attention Networks (GATs) that allow for the modeling of spatial dependencies [48]. Liu and Fu applied this hybrid approach to renewable energy forecasting, although the model has increased complexity and demands substantial computational power to train, which limits its scalability for widespread use. Additionally, the application of self-supervised techniques can help in discovering and learning more features from energy data. Xiang et al. proposed a hybrid model combining a Self-Attention Temporal Convolutional Network (SATCN) [49] with an LSTM to predict ultra-short-term wind power. This outperforms existing methods like LSTM, CNN-LSTM, and TCN-LSTM in terms of root mean square error (RMSE) and mean absolute error (MAE). Huang et al. demonstrated that this approach, when combined with Transformer models, has the potential to increase the generalization capabilities in multi-step forecasting tasks [50]. For residential buildings, a hybrid model combining multivariate variational mode decomposition, and a Temporal Fusion Transformer has been developed for short load forecasting and it decomposes the load series into intrinsic mode functions, which makes the model handle variations within the input data better [51]. This, combined with a whale optimization algorithm, results in higher forecasting accuracy. Finally, Pan et al. [52] introduced a transformer framework that uses a local-global scale architecture. The model is highly applicable due to the integration of edge computing and IoT devices that enable real-time data collection and processing.

Despite their strengths for natural languages processing, the state-of-the-art transformers models have specific drawbacks in time series energy predictions, regarding computation, temporal dynamics and noise handling. The computational cost of these models, especially when applied to large energy datasets, remains a major drawback [53,54]. Wang et al. [55] note that the high computational requirements for multi-source data fusion models may lead to a decline in performance in real-time applications. Furthermore, for specific scenarios, LSTMs or linear regression can outperform the Transformer model and in environments where data is low-dimensional, the marginal accuracy improvements do not always justify the computational costs. To address these limitations, we propose a hybrid approach that integrates a Transformer-based architecture with Liquid Neural Networks (LNNs) and learnable encodings to handle both the complex temporal dependencies and non-linear patterns found in energy consumption data. The model leverages Dense Layers to learn non-linear mappings, enabling it to create embeddings that capture underlying patterns in time series energy data. A Convolutional Neural Network (CNN) encoder

is integrated to enhance the model's ability to understand temporal dynamics through spatial mappings. At the same time, an adaptive positional encoding combines sinusoidal, temporal, and learnable encodings. To address the limitations of classic attention mechanisms, which primarily rely on linear transformations and simple dot products and may not capture these non-linear relationships effectively, we introduce a Reservoir Processing module implemented using LNNs. A controlled non-linearity is introduced through dynamic reservoir computing, which can model complex patterns in the data that are not immediately apparent or are obscured by noise and fluctuations. The proposed transformer model shows good prediction accuracy with good training time efficiency over various pilot data and state-of-the-art datasets while considering different building contexts, such as large buildings of apartments or small households with and without on-site energy production.

The paper's contributions beyond the state of the art are:

- A hybrid transformer model for building energy forecasting that combines the transformer architecture with LNNs, and learnable encodings achieving good prediction accuracy and training efficiency across different building contexts.
- Integration of CNN encoders with adaptive positional encoding to capture non-linear mappings and enhance the understanding of temporal dynamics in time series energy data.
- A Reservoir Processing module implemented using Liquid Neural Networks to capture more effective complex non-linear energy data patterns that can be masked by noise and fluctuations.
- Evaluation of predictive performance across different types of buildings including apartments building, residential and commercial buildings and households prosumers showing good generalization features and energy prediction accuracy.

The rest of the paper is organized as follows: Section 2 describes the architecture of our proposed hybrid transformer model detailing its key components; Section 3 details the experimental setup and energy prediction results for various building cases and Section 4 discuss the prediction accuracy and training time results in comparison with basic transformer, LTSM and ANN, whilst Section 5 concludes the paper and presents potential future directions.

2. Hybrid transformer model for energy prediction

We have adapted the transformer model to the specific needs of energy consumption or production prediction (as illustrated in Fig. 1) by designing each component to maximize its efficacy in learning from time series data. The description of the proposed hybrid model focuses on the core building blocks, their interplay, and the rationale behind their selection, emphasizing their contributions and potential limitations within the context of time series energy prediction.

2.1. Embeddings and adaptive encodings

The input data is a sequence of historical energy consumption data, $E_t \in R$ with Δt the interval between two data samples and k the length of the sequence:

$$E_t = (e_{t-1}, e_{t-2}, \dots, e_{t-k}) \quad (1)$$

where e_{t-k} , is the energy data value sampled at time $t - k$. For each energy data value, we derive and use a set of d features based on the time stamp value:

$$e_{t-k} \rightarrow X_{t-k} = (x_1, x_2, \dots, x_d) \quad (2)$$

where x_d is a feature considered such as weather, time of day, day of week, etc. We define EF a matrix with d rows and $(t-k)$ columns, storing E_t and X_t for each t from $t-1$ to $t-k$:

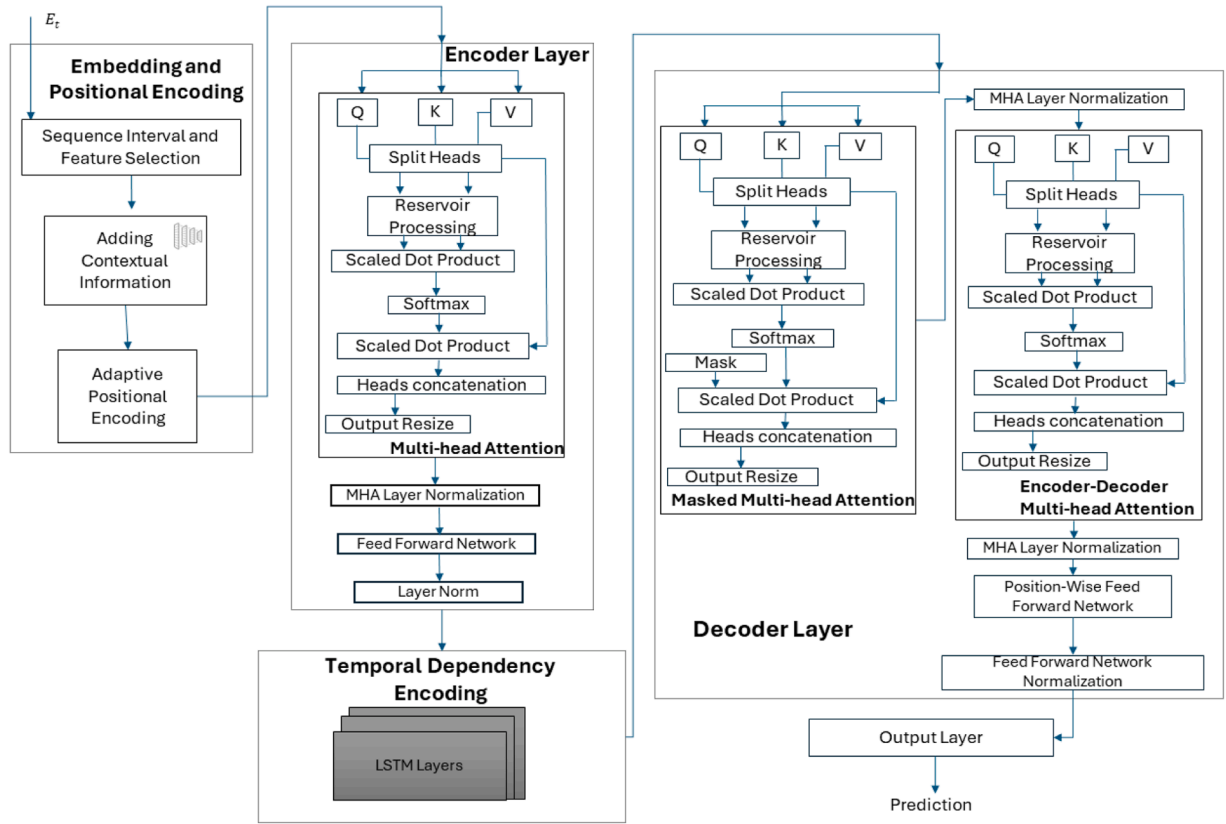


Fig. 1. The architecture of the hybrid transformer model for building energy forecasting.

$$EF \in \mathbb{R}^{d \times (t-k)} \quad (3)$$

For the initial transformation of input data, we employ Dense Layers which are fully connected layers capable of learning non-linear mappings. This enables the model to create representations of input sequences called embeddings that might capture underlying energy patterns or trends. Each time series input E_t is first mapped to a high-dimensional space through an embedding layer:

$$EO_t = EF * W_{EO} + b_{EO} \quad (4)$$

where $W^E \in \mathbb{R}^{(t-k) \times d}$ and $b^E \in \mathbb{R}^d$ are the learnable parameters, with $(t-k) * d$ represents the dimensionality of the hyperparameter, representing the size of the internal representation vectors within the model. As the activation function for this layer, we have used the SELU (Scaled Exponential Linear Unit) designed to induce self-normalizing properties in neural networks. It helps in maintaining a mean of zero and standard deviation of one for the inputs to each layer, leading to a more stable and faster convergence during training.

We have integrated a CNN encoder which takes as input the generated embedded time series energy data and adds supplementary context related to embeddings related to the sequential data input. Each input sequence of the CNN encoder is represented as a matrix $EO_t \in \mathbb{R}^{n \times k}$, is passed through a set of convolutional layers. Each convolutional layer applies filters to the input data to extract local patterns. For the i -th convolutional layer, the operation is formalized as:

$$C_i = \text{ReLU}(W_i * EO_t + b_i) \quad (5)$$

where C_i is the output of the i -th convolutional layer, W_i represents the weights of the filters in the i -th convolutional layer. Afterward, a pooling layer is used to reduce the dimensionality of the data, enhancing the model's efficiency in integrating the extra context. The operation is:

$$P_i = \text{MaxPool}(C_i) \quad (6)$$

Before passing the output to the Encoder layer of the model, the pooled output is flattened to match the model's input requirements:

$$C_{out} = \text{Flatten}(P_m) \quad (7)$$

where P_m is the output of the last pooling layer.

Since basic transformer architectures do not inherently process sequential data in order, positional encodings are added to the input embeddings to provide information about the order of the energy data samples. This ensures that our hybrid model understands the temporal dynamics of the data. We employ an adaptive positional encoding approach, combining sinusoidal and learnable encodings. This method leverages the strengths of both types to capture temporal patterns more effectively.

The Sinusoidal Encoding uses sine and cosine functions to generate encodings, which are added to the input embeddings. The *cycle length* parameter adapts to different temporal patterns, such as daily, weekly, or monthly cycles:

$$PE_{\text{sinusoidal},os,2i} = \sin\left(\frac{pos}{10000^{2i/d}} \cdot \log(\text{cycle_length}_i)\right) \quad (8)$$

$$PE_{\text{sinusoidal},os,2i+1} = \cos\left(\frac{pos}{10000^{2i/d}} \cdot \log(\text{cycle_length}_i)\right) \quad (9)$$

where *pos* represents the position in the energy data sampled considered in the input sequence and i is the dimension of the positional encoding.

The Learnable Encoding is defined to enhance the adaptability of the sinusoidal encoding, learnable parameters α and β are introduced:

$$PE_{\text{learnable},pos} = \alpha * PE_{\text{sinusoidal},pos} + \beta \quad (10)$$

Additionally, the Time-based Encoding is leveraged to explicitly encode time-based features like hour of the day, day of the week, day of the month and month:

$$PE_{time, pos} = [f_{hour}(pos), f_{day_week}(pos), f_{day_month}(pos), f_{month}(pos)] \quad (11)$$

where each feature function $f_{type}(pos) = \frac{pos \% type_{rec}}{type_{rec}}$ normalizes the position by its recurrence period. Finally, the positional encoding vector is an **adaptive** combination of **learnable** and **time-based** encodings:

$$PE_{adaptive, pos} = PE_{learnable, pos} + PE_{time, pos} \quad (12)$$

while the modified embedding matrix after adding positional encodings is:

$$EF'.e'_t = C_{out} \oplus PE_{adaptive}(t) \quad (13)$$

2.2. Encoder layer with LLN

It implements a multi-head attention mechanism that computes a representation of a modified embedding matrix by relating to different positions of the same input sequence. The process involves three main steps: computing query (Q), key (K), and value (V) matrices from the input data, computing the attention scores, and then applying the attention scores to the values:

$$Q = EF'.e'_t * W^Q, K = EF'.e'_t * W^K, V = EF'.e'_t * W^V \quad (14)$$

where $W^Q, W^K, W^V \in \mathbb{R}^{d_k \times d_{model}}$ trainable weight matrices and d_k represents the dimensionality of the query/key vectors. These projections are meant to transform the inputs into spaces that reflect different aspects of the input necessary for attention calculation.

The **Split Heads** operation enables the model to attend to information at different positions, increasing the model's flexibility and capacity. The projections are then split into multiple heads, allowing the model to capture information from different representation subspaces at different positions.

$$head_i(Q) = Q_i, head_i(K) = K_i, head_i(V) = V_i \quad (15)$$

where (Q_i, K_i, V_i) represent the portions of Q, K, V corresponding to the i -th head and d_{model} is divisible by the number of heads.

A significant drawback of this approach is that the energy consumption data often exhibit non-linear patterns due to various factors such as variable consumer behavior, sudden changes in weather, or differing responses to price fluctuations. Classic attention mechanisms, which primarily rely on linear transformations and simple dot products, may not capture these non-linear relationships as effectively. To bypass this, we added a **Reservoir Processing module** implemented using Liquid Neural Networks. This introduces a controlled non-linearity through its dynamic reservoir computing approach, which can model complex patterns in the data that are not immediately apparent or are obscured by noise and fluctuations. By processing queries and keys through the Liquid Neural Networks, the attention mechanism can dynamically adjust how different parts of the sequence influence each other. Also, the Liquid Neural Networks can function as a selective attention outlet, by offering adaptive attention to more important features and reducing convergence times.

The Liquid Neural Networks processes the queries and keys to introduce the controlled non-linearity and dynamic temporal dependencies by:

$$Q'_t = LiquidLayer(Q_t), K'_t = LiquidLayer(K_t) \quad (16)$$

The Liquid Neural Networks involve random initialization of reservoir weights, adjustment to control the spectral radius, and a non-linear activation function (tanh):

$$reservoir_{weights} \sim Uniform(-1.0, 1.0) \quad (17)$$

After initialization, it adjusts its reservoir weights to control the spectral radius, ensuring stable dynamics during computation. This adjustment involves rescaling the weights such that the maximum

eigenvalue of the weight's matrix matches a desired spectral radius, specified by the $spectral_{radius}$ parameter. Mathematically, this operation is expressed as:

$$reservoir_{weights} \leftarrow \frac{reservoir_{weights}}{\max(eigenvalue)} \times spectral_{radius} \quad (18)$$

where $\max(eigenvalue)$ is the maximum eigenvalue of the reservoir weights matrix.

During computation, the Liquid Neural Network updates its reservoir state by performing matrix multiplication of the input with the reservoir weights. This pre-activation step results in a linear transformation of the input data. Subsequently, the computed pre-activation values are passed through a hyperbolic tangent (tanh) activation function elementwise, introducing non-linearity into the transformation:

$$pre - activation = Q_t * reservoir_{weights}, Q'_t = \tanh(pre - activation) \quad (19)$$

The attention scores determine how much focus should be placed on other parts of the input sequence when encoding a particular part of the sequence. We have used the **Scaled Dot-Product** attention that computes the attention score for a given element by taking the dot product of its query vector with the key vector of every other element, followed by a softmax operation to obtain the attention scores (weights). For each head, attention weights are computed and scaled.

$$attention_{scores_t} = \frac{Q'_t \times K'^T}{\sqrt{d_k}} \quad (20)$$

$$attention_{weights_t} = \text{softmax}(attention_{scores_t}) \quad (21)$$

where d_k is the dimension of the keys. The scaling factor $(\sqrt{d_k})$ prevents the softmax function from having extremely small gradients during backpropagation, which can occur with large values of dot products.

The values V are weighted by the attention weights for each input $EF'.e'_t$ to produce the output of the self-attention mechanism.

$$attention_{out_t} = attention_{weights_t} * V \quad (22)$$

The outputs of each **head are concatenated** and once again projected, resulting in the final value of the Multi-Head Attention layer's output, which is then resized to the necessary dimensions:

$$MultiHead(Q, K, V) = \text{Concat}(head_1, \dots, head_h) * W_O \quad (23)$$

where $W_O \in \mathbb{R}^{hd_v \times d_{model}}$ is a trainable weight matrix comprised of the weight matrixes for each head, h is the number of heads, and d_v is the dimensionality of the values.

After attention processing, the FFN (**Feed-Forward Network**) further transforms the attention output. The inclusion of gating mechanisms, such as Gated Linear Units (GLUs), allows the model to regulate the flow of information through the network, adding an additional layer of non-linearity and control. Applied before each sub-layer (attention and FFN) and to their outputs, layer normalization helps stabilize the neural network's learning process, mitigating issues related to varying scales and distribution of layer inputs. Finally, dropout was used as a regularization technique, randomly setting a fraction of the input units to 0 at each update during training time, which helps prevent overfitting by ensuring that no subset of neurons becomes too reliant on each other. Given the input ME' resulting from the positional encoding phase, the encoder layer's operations can be formalized as follows:

$$A = \text{Dropout}(MHA(EF')) + EF'.e'_t \quad (24)$$

where $MHA(ME'.e'_t)$ represents the Multi-Head Attention operation applied to $ME'.e'_t$, and A is the attention output after dropout is applied and added to the original input through residual connection. Then we apply the first normalization on the attention output:

$$L_1 = \text{LayerNorm}(A) \quad (25)$$

where L_1 is the result of applying the first layer normalization on the attention output. The normalized result will be then passed through the **feed-forward neural network**.

$$\text{FFN}_{\text{out}} = \text{Dropout}(\text{FFN}(L_1)) + L_1 \quad (26)$$

$\text{FFN}(L_1)$ represents the Feed-Forward Network operation and FFN_{out} is the FFN output after dropout is applied and added to L_1 through another residual connection. In the next step, we add the second layer of normalization:

$$\text{Enc}_{\text{out}} = L_2 = \text{LayerNorm}(\text{FFN}_{\text{out}}) \quad (27)$$

where L_2 is the final output of the encoder layer after applying the second layer normalization on the FFN output.

The encoder layer's output L_2 is then passed to the subsequent layer in the encoder stack or used as the final encoding of the input sequence in the case of the last encoder layer. This process enables our hybrid transformer model to capture and refine a rich set of features and contextual information from the input time series data.

After processing the input through the encoder layers, the next step is to capture long-term dependencies and temporal patterns using LSTM layers. We have used them due to their ability to maintain and update cell states, which helps in remembering information over long sequences. In our hybrid transformer model, the LSTM layers are used to further process the encoded sequence from the encoder layers. The hidden state and cell state are computed as follows, and the output of the LSTM module (LSTM_{out}) will be the hidden state output h_t :

$$\text{LSTM}_{\text{out}} = h_t, C_t = \text{LSTM}(\text{Enc}_{\text{out}}, h_{t-1}, C_{t-1}) \quad (28)$$

2.3. Decoder layer

The decoder layer is responsible for generating the output sequence from the encoded input. It consists of several sub-layers like the encoder, but with additional mechanisms to incorporate the encoder's output. Given the **LSTM output** LSTM_{out} , and the initial decoder input D_{input} , which is the output of the encoder layer, the integration can be formalized as follows:

$$D_t = \text{LSTM}_{\text{out}} + \text{Enc}_{\text{out}} \quad (29)$$

Similarly to the encoder, the input of the decoder is passed through a multi-head self-attention mechanism which is linearly projected into queries, keys, and values. After the projection, a masking mechanism is applied to prevent attending to future values and the values (V) are weighted by the attention weights to produce the **masked self-attention** (MA_t) output which is then normalized:

$$MA_t = \text{softmax}\left(\frac{Q_t \times K_t^T}{\sqrt{d_k}} + \text{mask}\right) \cdot V \quad (30)$$

The **Encoder-Decoder Attention** mechanism that uses as input the normalized masked self-attention matrix deriving Q_t , while the keys and values are derived from the encoder's output (Enc_{out}) and will be noted as KA_t and VA_t . The **Encoder-Decoder Attention** output is noted with EDA_t . The mechanism allows the decoder to pay attention to the encoder's output, incorporating contextual information from the input sequence:

$$EDA_t = \left(\text{softmax}\left(\frac{Q_t \times KA_t^T}{\sqrt{d_k}}\right) \cdot VA_t\right) \quad (31)$$

The **Encoder-Decoder Attention Mechanism** output is then passed to the **position-wise feed-forward network** to introduce non-linearity and project the data to the desired dimensions. This applies two linear transformations (using Dense layers) with a ReLU activation in between. This step involves a Feed-Forward Network (FFN) with gating

mechanisms, followed by dropout and layer normalization, integrated with a residual connection, which allows the original input to the FFN to be added back to the FFN's output, facilitating smoother gradient flow, and aiding in training deep networks:

$$\text{FFN}_{\text{out}} = \text{Dense}(\text{ReLU}(\text{Dense}(\text{EDA}_t))) \quad (32)$$

The decoder layer's output is computed in the next step when the feed-forward network output is passed through a normalization layer while the output layer uses a dense layer to apply linear transformation to generate the forecasted energy values:

$$E_{\text{predicted}} = \text{Dense}(\text{LayerNorm}(\text{FFN}_{\text{out}})) \quad (34)$$

3. Evaluation results

We utilized both pilot data and state-of-the-art datasets to validate the model's performance across different building contexts, such as large buildings of apartments or small households with and without on-site energy production.

For all evaluation cases, we have considered a wide array of features each contributing to the predictive performance of the proposed hybrid transformer model. Temporal features like the hour of the day, day of the week, week of the year, month of the year, and day of the year are used to capture patterns in energy consumption. Next, we incorporated exogenous variables such as weather data, including outside temperature, humidity, wind speed, cloud cover, and precipitation amount. They directly influence the heating and cooling loads of buildings, which are major components of energy consumption. For instance, higher outside temperatures increase the demand for air conditioning, while lower temperatures increase heating requirements. Humidity levels can affect indoor climate control systems, leading to variations in energy usage. To understand autocorrelations in buildings' energy usage we have used lagged values of consumed energy from the last 6-hour window, the previous day and week. Statistical features such as moving averages, moving medians, exponentially weighted moving averages, and rolling standard deviations over 6-hour and 24-hour windows were used to assess short-term trends and variability in buildings' energy consumption. Additionally, rolling skew and kurtosis measures were used to understand changes in the distribution shape of energy consumption patterns. Trends in building energy consumption profiles are analyzed using the average hourly consumption for each day of the week, the rate of change in energy consumption, and components extracted from time series decomposition, such as trend, seasonality, and residuals. We have included anomaly detection features like outlier indicators, spike identification, spike magnitude to detect unusual events, and frequency features derived from the Fourier Transform of the energy consumption signal to capture cyclical patterns in energy consumption. Finally, we have explored potential nonlinear relationships between consumption and its statistical measures.

From the extended set of features, we selected the specific and relevant subset for each evaluation case using the Firefly Algorithm [56]. The algorithm iteratively refined the feature subsets by attracting less optimal solutions toward better ones, based on a fitness function, ensuring diversity in the search, while converging toward the optimal subset of features. This metaheuristic optimization technique permitted us to identify the most critical features for each case by maximizing the evaluation metric while minimizing redundancy and irrelevant information. By employing the feature selection algorithm for each scenario, we ensured that each dataset's particularities will be considered, to enhance the model's predictive performance. Also, this approach improved training efficiency and reduced the risk of overfitting, as it allowed the model to focus on the most significant inputs. Moreover, we have employed regularization techniques such as early stopping to halt training when the performance no longer improves and model checkpoint to save best model parameters during training and resume training in case of interruptions. The selection of the hyperparameters was

conducted using a grid search approach and domain knowledge to ensure optimal performance. The selection of a 24-dimensional input corresponds to using a sequence length of 24, representing a day of hourly consumption data. The choice was grounded in the daily periodicity presented by our datasets, given especially by the 1-hour granularity. The rest of the hyperparameters were determined through a guided search, where we tested different values selected from pre-defined ranges that were chosen based on prior experience, domain knowledge and model behavior. Table 1 shows the structural hyperparameters utilized for the hybrid transformer model training process that resulted in the best trade-off between training time and model accuracy, in all evaluation cases. It is also important to note that to prevent overfitting, we relied on a strategy that combined model checkpointing to save the best model in terms of accuracy on the validation set and early stopping to halt training when validation loss stopped decreasing.

3.1. Building of apartments

We utilized energy consumption data from two residential buildings (i.e., 1F and 1B) where students can rent 234 apartments yearly. To monitor electricity usage, a Building Energy Management System (BEMS) has been installed. MQTT-based data integration is used, where producers send data to a queue at a fixed rate. MQTT consumers then collect data from the queue and perform the necessary energy conversions. Energy consumption data was collected over an extended period throughout the year 2022, spanning from January 1st to December 31st. The building is used throughout the whole year, however with different energy consumption. Fig. 2 monthly average energy consumption shows seasonal trends, with noticeable dips around the summer months (June to August), and peaks towards the end of the year (October to December). The energy demand in both buildings' data align, suggesting similar usage patterns and shared external factors influencing consumption.

Table 2 summarizes statistical metrics for the energy consumption of the two buildings. The metrics indicate that the energy consumption patterns in both buildings are quite similar, in both range and value.

The feature subset selected for both buildings, as determined by the Firefly Algorithm, encompasses comprehensive information about the energy consumption patterns. This subset includes exogenous variables such as external temperature, temporal dependencies, and methods for smoothing short-term fluctuations. The selection process ensures the model's capability to accurately account for daily and weekly usage patterns, seasonal trends, and anomalous consumption events, as shown in Table 3.

Fig. 3 shows the energy consumption predictions for building 1F and building 2B revealing key insights into the hybrid transformer model's performance. The model excels particularly during stable periods where the energy consumption follows typical daily or seasonal trends with minor fluctuations, the model performs better, maintaining a good level of accuracy.

Table 1
Hybrid transformer model hyperparameters.

Hyperparameter	Range	Values selected
Training epochs	-	75 epochs
Input dimensions	-	24-dimensional inputs
LSTM units	Between 32 and 512 in increments of 32	128 LSTM units
Attention heads	Between 1 and 10, increasing by 1	8 attention heads
Stacked encoder layers	Between 1 and 12, increasing by 1	6 stacked encoder layers
Liquid layers	Between 10 and 100, increasing by 10	50 liquid layers
Learning rate	-	Adaptive learning rate (0.001 decreasing to 1e-7)
Sequence length	24	24

This suggests that while the model is robust for most routine scenarios, enhancements are needed to improve its adaptability to unusual or rapidly changing consumption patterns. However, the model's performance tends to degrade during periods of sudden changes or anomalies in energy usage. For instance, during unexpected spikes or drops in consumption, such as holiday periods (late December), the predicted values can significantly diverge from the actual data. These instances highlight areas where the model could benefit from further refinement, possibly by incorporating additional contextual data or advanced techniques to better capture these abrupt changes. In average cases, where the energy consumption. On average, the model performs quite well, with mean sMAPE values of approximately 10.75 % for building 1F and 8.52 % for building 2B (see Table 4). This indicates that the predictions are generally close to the actual values, reflecting a good overall accuracy.

We took a more in-depth look at the hourly energy prediction accuracy distribution throughout the day (see Fig. 4). In the case of building 1F peak errors occur during the early morning hours (00:00 - 02:00), indicating challenges in making accurate predictions during this period. This may be due to unexpected load spikes from late-night activities or equipment usage or changes in resident behavior such as late-night energy usage. Errors tend to decrease and stabilize during the middle of the day, suggesting better model performance during these hours. This stability could be attributed to predictable routines, as energy consumption likely follows a more consistent pattern due to daily routines and business operations. However, another rise in errors is observed in the late evening, indicating a second period of increased prediction difficulty. This could be due to increased usage of electrical devices as residents return home and engage in various evening activities, leading to more variable load patterns. Building 2B's hourly error distribution is more stable compared to building 1F, resulting in a lower prediction error. There are moderate peaks in errors during the early hours (00:00 - 01:00), like building 1F. Additionally, during morning hours (07:00), there is a peak, possibly due to early-day activities such as cooking, heating or cooling adjustments. Throughout the rest of the day, errors are relatively lower and consistent, indicating more reliable predictions during these hours. This consistency might be due to the predictable routine of energy consumption following daily patterns and business operations. Like building 1F, another rise in errors is observed in the late evening (23:00 - 00:00).

3.2. Residential building

The residential building selected is in Wiener Neustadt, Austria. The building consists of two floors, each equipped with a dedicated PV system supplying electricity. The data includes energy consumption data from the energy meter of the first floor and energy production data from the PV system supplying this floor. The data spans two years with one-hour granularity, providing a comprehensive foundation for our analyses and subsequent modeling efforts.

The data underwent extensive pre-processing to ensure its suitability for our hybrid transformer model training. Missing and anomalous values were meticulously handled: zero and negative consumptions were identified, with negative consumptions replaced by the mean of neighbouring non-negative values. Specific dates with negative consumptions were excluded, and a similar approach was applied to energy production anomalies, such as stuck values. In the case of energy consumption prediction feature engineering was performed to enhance the data. Table 5 summarize the selected features identified by the Firefly algorithm as the most significant for predictions enabling the simplification of the model by focusing on the most impactful variables. The features include direct measurements or time-based features that are essential inputs into the model. The derived features are calculated values such as Z-scores or rolling statistics, which help capture trends or anomalies in the data. The interaction features involve the combination or interaction of other features to provide deeper insights, such as

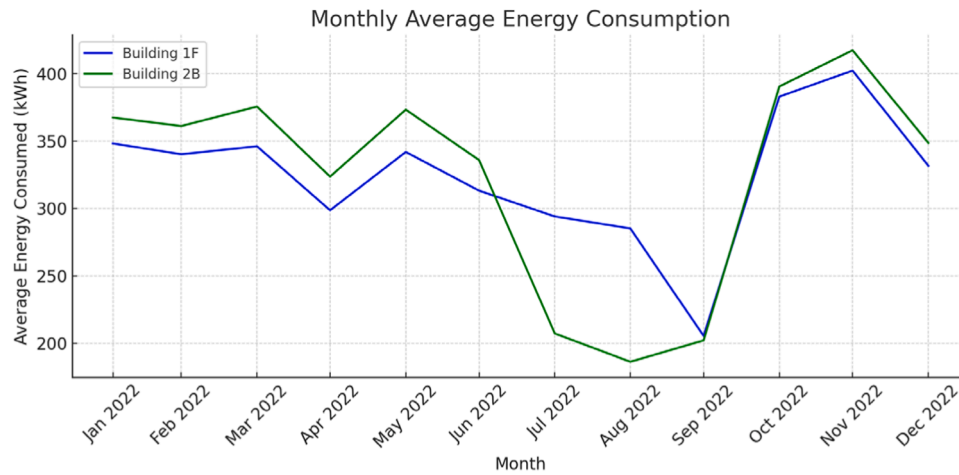


Fig. 2. Energy consumption patterns of the buildings.

Table 2

Collected energy data statistics.

Metric	Building	
	1F	2B
Mean value	326.58	323.25
Standard Deviation	163.60	178.38
[Min-Max] kWh	[0.0–963.60]	[0.0–948.1]

detecting spikes or examining interactions between rolling statistics.

Fig. 5 shows the comparison between actual energy consumption (in blue) and the predicted energy consumption (in red) over a period represented by the timestamps on the x-axis.

Our hybrid transformer model demonstrates the ability to closely follow the actual consumption patterns, though some deviations are visible, especially during periods of higher variability in energy use. This visualization highlights the effectiveness of the predictive model in capturing the overall trends and fluctuations in energy consumption.

For the PV production prediction model, the set of relevant features focuses primarily on critical weather variables (like cloud cover, wind speed at 100 m, and various radiation measures), specific temporal features (hour of day and day of the week), essential lagged features, and a few key statistical and frequency-based features. These selected features were deemed most influential for accurately predicting PV production, enhancing the model's efficiency and predictive power (see Table 6).

Fig. 6 compares the actual PV production (shown in blue) with the predicted PV production (shown in red) across the given timestamps. Our hybrid transformer model demonstrates a strong alignment with the actual production data, accurately capturing the cyclical nature of solar

energy generation throughout the day. The figure shows that the predicted production closely follows the actual production, particularly during periods of high solar activity. However, there are slight deviations, especially during the peaks and troughs, which suggest opportunities for further refinement in the model. Overall, the model performs well in forecasting PV production, as evidenced by the close overlap between actual and predicted values.

In addition, a summary of the overall predictive capabilities and accuracy levels of our transformer model for both energy consumption and production in the residential building is presented in Table 7. Our model has performed notably well in predicting both consumption and production, with relatively low MSE and RMSE values. The R^2 values indicate that the model explains a significant portion of the variance in both consumption and production data, with an especially strong performance in predicting production.

3.3. Household consumers

We have assessed the prediction accuracy of our hybrid transformer model for households' energy consumption in different regions (i.e., UK and Portugal) and climatic factors to see the influence on the model performance and generalization capabilities. London households exhibit greater variability in energy usage, particularly due to seasonal effects, which poses a challenge for the predictive model. On the other hand, the Portugal households display more consistent energy consumption patterns, especially during the summer months.

3.3.1. London households

We selected a subset of 100 London households (i.e. ids: MAC000001-MAC000100) on a standard flat-rate tariff from the smart meter energy consumption dataset [57] provided by UK Power

Table 3

A subset of features used for hybrid transformer model training for buildings 1F and 2B.

Category	Basic features	Derived features	Interaction features
Weather data	Outside Temperature, Humidity	N/A	N/A
Temporal features	Hour of Day, Day of Week, Week of Year, Day of Year		
Lag features	Previous Day, Previous Hour, Lag 2 h, Lag 3 h, Lag 4h		
Statistical features	N/A	24-Hour Moving Median, Exponentially Weighted Moving Average (24-Hour), 24-Hour Rolling Standard Deviation	Rolling Mean Interaction, Rolling Standard Deviation Interaction
Trend features		Daily Consumption Trend, Energy Consumption Gradient	Rolling Average Difference
Anomaly detection		N/A	Spike Indicator
Frequency features		Dominant Frequencies from Fourier Transform (Frequency 2, Frequency 3, Frequency 5)	N/A

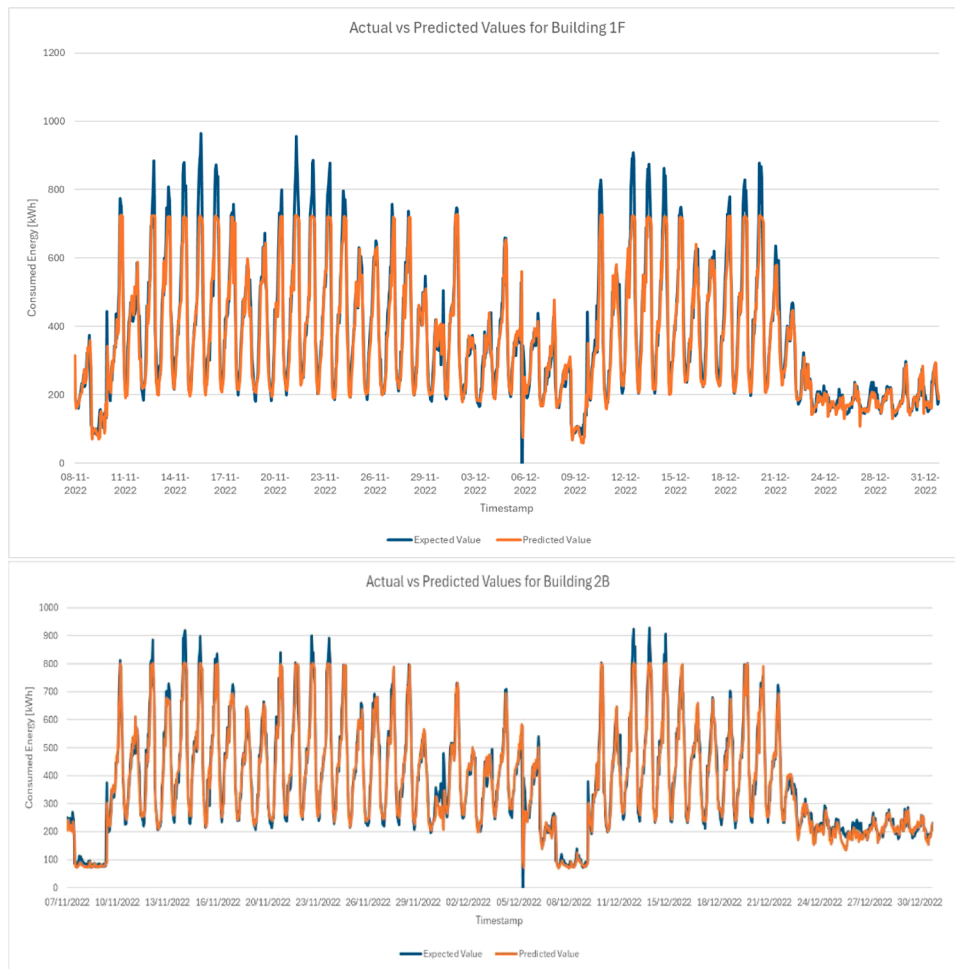


Fig. 3. Hybrid transformer model energy prediction results for apartments' building.

Table 4

Average energy prediction accuracy for apartments' building case.

Energy consumption profile	sMAPE	R ²	MAE
Building 2B	8.52 %	0.94	33.87
Building 1F	10.75 %	0.91	38.63

Networks. We have analyzed the energy consumption patterns of the selected households to understand the data distribution variability, outliers and variability (see Table 8). There is a notable difference between the mean and median suggesting a skewed distribution, characterized by a long tail of higher consumption values. The maximum recorded energy consumption is 10.009, while the minimum is 0. This wide range reflects significant variability among households, which could negatively affect the accuracy of predictive models.

For feature selection, the Firefly algorithm produced different sets of relevant features for the selected households. However, there are common features selected across all households, which are reported in Table 9.

Afterwards we evaluate the predictive performance of our proposed hybrid transformer model using several metrics. Fig. 7 presents box plots for sMAPE and R² metrics computed for all households providing information about the accuracy and reliability of the predictions. The median sMAPE is approximately 20 %, and the interquartile range spans from about 15 % to 25 %, showing that the central 50 % of the predictions have errors within this range. This indicates a moderately good level of accuracy. There are several outliers with sMAPE values

exceeding 40 %, and a few reaching up to 60 % where the models struggled to make accurate predictions, likely due to the high variability and presence of extreme values in the energy consumption data, as highlighted by the previous analysis. The box plot for R² shows the distribution of the goodness-of-fit measure for the predictive models. The median R² value is approximately 0.5, indicating that the models explain about 50 % of the variance in the observed energy consumption data. The presence of outliers with R² values close to 0.2 and up to 0.8 indicates that some models perform substantially better or worse than the majority, suggesting that there are specific conditions under which the models excel or fail.

Next, we selected 10 households with the best energy prediction outcomes and analyzed the results more in-depth by looking at the accuracy distribution during the day (see Fig. 8).

The analysis of the hourly sMAPE plot reveals significant variations in the model's predictive accuracy across different times of the day, highlighting specific periods where the model excels and others where it struggles. In the early hours of the day, particularly between midnight and 5:00 AM, the model shows relatively low sMAPE values, with the error decreasing to as low as 20–25 %. This suggests that during these hours, energy consumption is more consistent and predictable, likely due to the stable and minimal usage patterns associated with night-time inactivity. However, there is a noticeable spike in sMAPE at 6:00 AM, where the error reaches over 40 %. This increase coincides with the beginning of the day for many households, where energy usage can vary significantly depending on the activities initiated, such as preparing breakfast, heating, and other morning routines. As the day progresses, from 7:00 AM to around 10:00 AM, the sMAPE gradually increases,

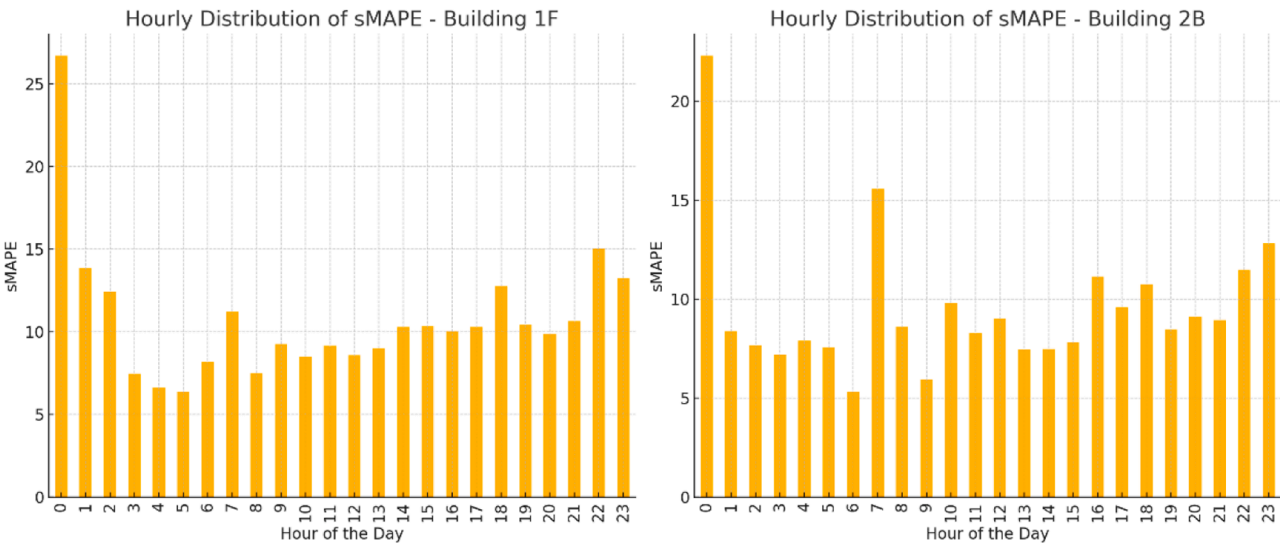


Fig. 4. Hourly energy prediction accuracy distribution over a day.

Table 5
Features used for hybrid transformer model training for the residential building consumption.

Category	Basic features	Derived features
Weather data	Temperature at 2 m, Wind Speed at 10 m, Direct Radiation, Direct Normal Irradiance	N/A
Temporal features	Hour of Day, Day of Week, Month of Year	
Lag features	Previous Hour, Lag 2 h, Lag 5 h	
Statistical features	N/A	Rolling Skewness, Z-score
Frequency features		Dominant Frequencies from Fourier Transform (Frequency 1, Frequency 3, Frequency 5)

peaking again close to 10:00 AM. This period might reflect the transitional phase between morning routines and the start of work or school, where energy usage can be sporadic. The continued high error rate suggests that the model struggles to capture these changes, which may be due to overlapping activities that introduce a level of complexity not adequately addressed by the current model features. From 11:00 AM

Table 6
Features used for model training for residential building production.

Category	Basic features	Derived features
Weather data	Cloud Cover, Wind Speed at 100 m, Shortwave Radiation, Diffuse Radiation, Direct Normal Irradiance, Terrestrial Radiation	N/A
Temporal features	Hour of Day, Day of Week	
Lag features	Previous Hour, Lag 3 h, Lag 6 h	
Statistical features	N/A	Rolling Skewness, Rolling Kurtosis, Z-score, Rolling Average Difference
Frequency features		Dominant Frequencies from Fourier Transform (Frequency 2, Frequency 3, Frequency 4, Frequency 5)

onwards, the sMAPE shows a slight decline but remains elevated until early afternoon, hovering around 35–40 %. The model’s performance improves significantly from 4:00 PM onwards, with the sMAPE dropping

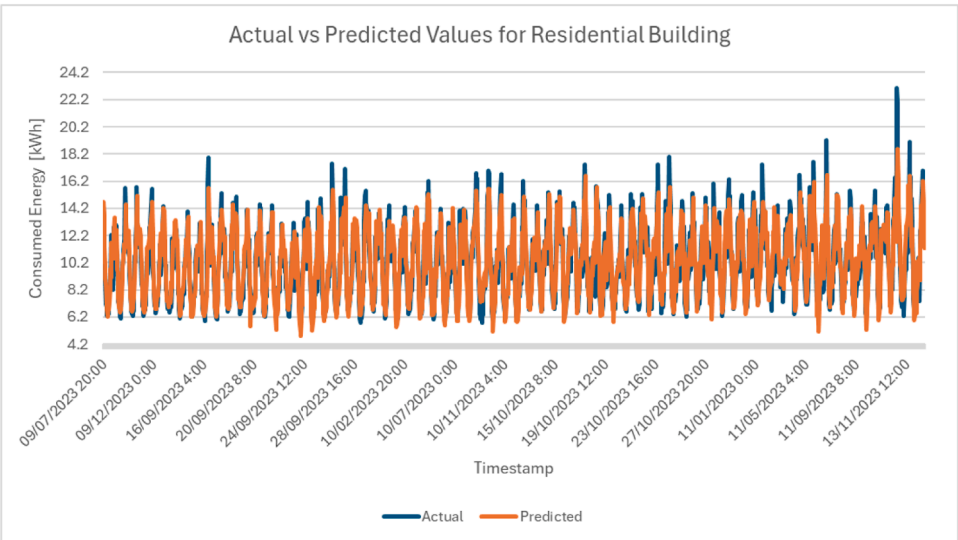


Fig. 5. Transformer model energy consumption prediction results.

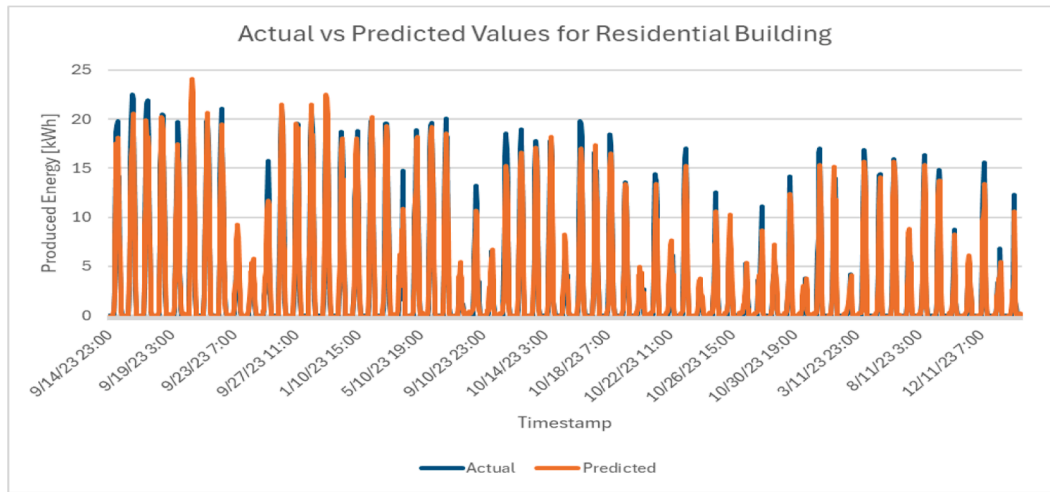


Fig. 6. Performance of the hybrid transformer model for energy production.

Table 7

Hybrid transformer model overall prediction accuracy.

Energy profile	sMAPE	R ²	MAE
Building consumption	9.06 %	0.77	0.94
Building production	–	0.94	0.66

steadily, reaching its lowest point in the late evening around 8:00 PM. This suggests that during this period, energy consumption becomes more predictable, likely due to consistent evening routines such as cooking dinner, heating, and other household activities. However, there is another spike in sMAPE at 9:00 PM, where the error increases sharply. This could be due to the variability in late-evening activities. The hourly R² analysis highlights the model's varying performance throughout the day, with better performance observed during periods of consistent, predictable energy usage, such as early morning and late evening. However, the model's struggles can be seen during the afternoon and evening hours. Finally, Fig. 9 shows the energy prediction results for one of the households showing good accuracy results.

3.3.2. Portugal households

We have selected 100 households (ids: MT002-MT100) from a dataset that contains the electricity consumption of 370 clients from Portugal [58]. The data, measured in kilowatts (kW), time-stamped according to Portuguese local time, including adjustments for daylight saving changes, was converted into kilowatts-hour (kWh) to better fit the architecture of our hybrid transformer model and the experiment definition. Like the case of London households, we have performed an analysis of energy data distribution and variability the results being reported in Table 10.

We employed the Firefly algorithm for feature selection which are listed in Table 11.

The average value for sMAPE is 9.26 %, and for R² is 0.91, indicating good prediction accuracy. Fig. 10 presents box plots for the sMAPE and R² scores determined for all selected households. The median sMAPE is around 3–4 %, indicating low prediction error, and the median R² is close to 0.8, reflecting that the model explains approximately 80 % of the variance in the energy data. The relatively short interquartile ranges for both sMAPE and R² suggest consistent performance across most

households in the datasets. However, the presence of outliers, particularly in sMAPE values exceeding 10 % and some R² scores dropping below 0.5, highlights occasional challenges.

The hybrid transformer model shows good predictive accuracy, but encounters difficulties during periods of higher variability, particularly in the late afternoon and late evening (see Fig. 11). The lower accuracy rates in the morning could be attributed to the irregular levels of energy consumption typical for this time of day, where usage patterns may vary significantly. As the day progresses into the morning and early afternoon, there is an improvement in the model's accuracy, reflected in a downward trend in sMAPE. This could be linked to the onset of daily routines, where energy consumption becomes more predictable and consistent. During the afternoon and early evening, from around 14:00 to 20:00, the sMAPE continues to decline, reaching its lowest point at approximately 8 % between 17:00 and 19:00. However, after 20:00, the plot reveals a sharp increase in sMAPE, with error rates rising back to around 11 % by 23:00.

The R² scores fluctuate but generally remain strong, reflecting the model's ability to capture the underlying patterns despite high variability in energy consumption profiles. However, a decrease in R² occurs in the late afternoon, suggesting that the hybrid transformer model struggles with the increasing variability in energy use during the evening. The R² scores recover in the early evening, demonstrating the model's improved performance for more stable energy consumption patterns.

Fig. 12 shows the energy prediction results for one of the households

Table 9

Common features selected for model training.

Category	Basic features	Derived features
Weather data	Outside temperature, Humidity, Wind speed	N/A
Lag features	N/A	Previous day value, Previous week value
Statistical features		Trend, Residuals (from seasonal decomposition), Time since last significant change, EWM for the last 24 h
Frequency features		Dominant Frequencies from Fourier Transform (Frequency 4)

Table 8

Households' energy profile statistics.

Metric	Mean	Median	Maximum	Minimum	Standard Deviation	Variance
Value	0.452307	0.281	10.009	0.0	0.551803	0.304987

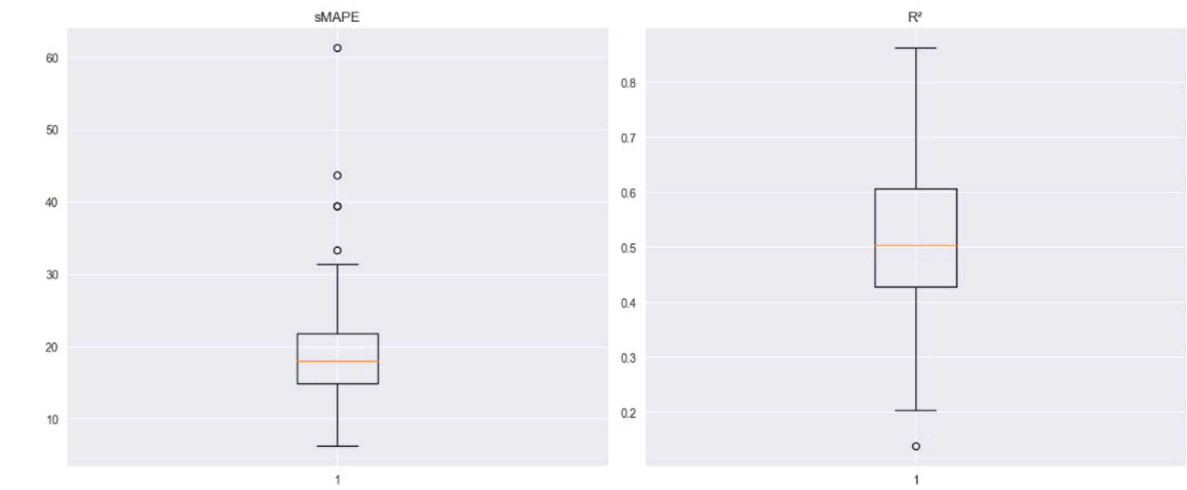


Fig. 7. Hybrid transformer model accuracy using sMAPE and R².

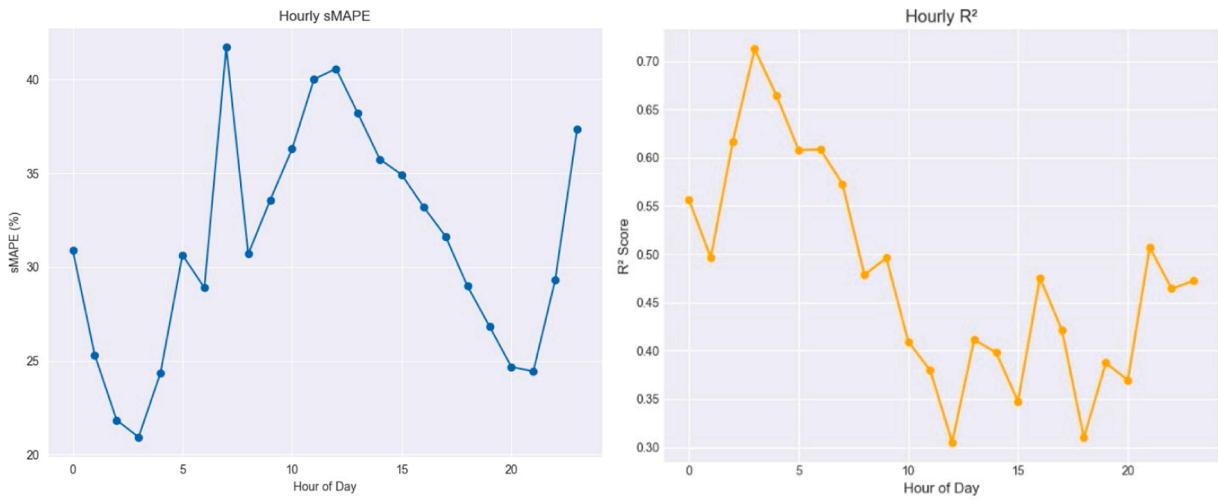


Fig. 8. Average sMAPE and R² scores during the day.

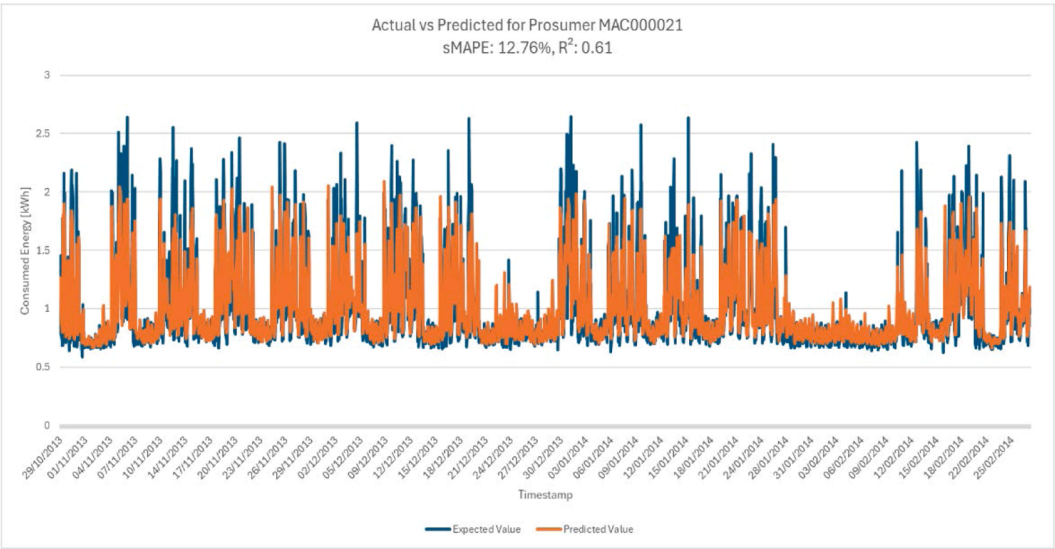


Fig. 9. Actual vs predicted energy profiles for household with id MAC000021.

Table 10
Households' energy profile statistics.

Metric	Mean	Median	Maximum	Minimum	Standard Deviation	Variance
Value	128.86	54.61	3860.35	0.0	260.844	68,040.08

Table 11
Common features selected in model training.

Category	Features
Temporal features	Hour of day
Lag features	Previous day value, Previous week value
Statistical features	Residuals, Spike Indicator, rolling skew, moving average for 6h
Frequency features	Dominant Frequencies from Fourier Transform (Frequency 2, 3 and 5)

showing a very good prediction accuracy with an sMAPE of 4.36 %.

3.4. Commercial building

The Building Data Genome Project 2 (BDG2) is an open dataset comprising energy consumption data from 3053 m across 1636 non-

residential buildings [59]. The dataset covers two full years, 2016 and 2017, with hourly measurements capturing electricity, heating and cooling water, steam, and irrigation usage. For our analysis, we utilized only the electricity consumption data from the dataset, focusing specifically on patterns and insights derived from this subset of the information provided. The experiments were conducted considering data from 3 commercial buildings, delimited by their identifier Robin (i.e., Robin Education Zenia, Robin Lodging Dorthy, Robin Public Carolina), corresponding to locations from University College of London. The buildings were selected based on having the fewest null and zero electricity observations. The models were trained using the same hyperparameter configuration. The Firefly algorithm was employed for selecting the best subset of features, which are listed in Table 12.

Across the three buildings, the trends that emerge in our hybrid model's performance are similar, and a consistent characteristic is that the model performs best during periods with stable energy consumption that exhibit lower variability (see Table 13). Moreover, the model



Fig. 10. Hybrid transformer model accuracy using sMAPE and R^2 .

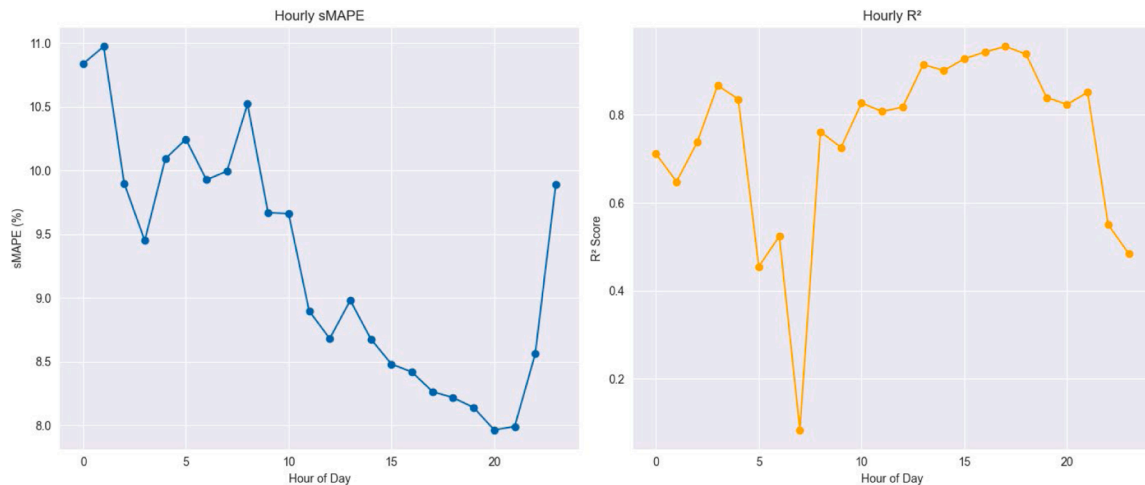


Fig. 11. Average sMAPE and R^2 scores during the day.

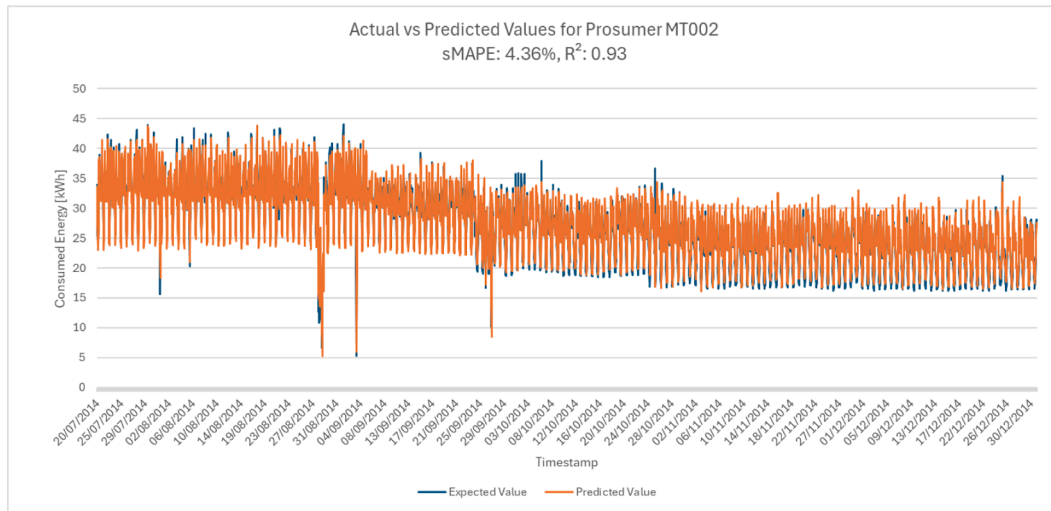


Fig. 12. Actual vs predicted energy consumption profiles for household with id MT002.

continues to show good predictive power during periods with moderate variability. Conversely, the proposed model struggles across all cases during transitional periods, particularly in the early morning stage or during abrupt spikes in energy consumption. The disadvantages of the model are clearly shown during timestamps where consumption reaches extreme values, highlighting its inability to predict sudden changes.

For the *Robin_Education_Zenia*, the model struggles the most during the early morning stage, where fluctuations in energy consumption are not well-captured. However, it performs best during the evening, reflecting stable energy use patterns during less variable periods (see Fig. 13). In contrast, for *Robin_Lodging_Dorothy*, the model excels in the afternoon, likely corresponding to consistent operational activities. It performs worse during the morning hours, which coincide with irregular energy demands such as operational surges in lodging facilities. For *Robin_Public_Carolina*, the model achieves its best performance during the evening, where energy consumption stabilizes. However, it exhibits the highest errors during the late-night and early morning hours.

The error metrics showcase a good overall accuracy and strong predictive power, showcased by 5.39 % average sMAPE and 0.92 average correlation score. The model explains the variance present in the data accurately, especially when general and strong trends are present. However, it also highlights the model's difficulty in effectively predicting energy consumption during transitional periods and high variability in energy datasets.

4. Discussion

In this section, we discuss the fundamental differences between the main types of models used for building energy prediction such as the classical transformer, LTSM and ANN, and our hybrid model (see Table 14), comparing prediction accuracy and training time across various cases.

ANNs rely on relatively simple solutions for modeling relationships

Table 12

Common features selected in model training for the commercial buildings.

Category	Features
Temporal features	Hour of day
Lag features	Previous day value, Previous week value, Lag 4 h, Lag 2 h
Statistical features	Trend, Seasonality, Residuals, Moving Average 6 h and 12 h, Rolling Standard Deviation 24 h
Frequency features	Dominant Frequencies from Fourier Transform (Frequency 2)

Table 13

Hybrid transformer accuracy for commercial buildings.

Name	sMAPE	MAE	R ²
Robin_education_Zenia	4.47 %	7.36	0.91
Robin_public_Carolina	5.41 %	7.40	0.94
Robin_lodging_Dorothy	6.28 %	2.15	0.91

in time-series energy data, using a straightforward structure of layers with interconnected neurons and activation functions. The models are computationally efficient but lack mechanisms to process sequential or temporal data which are rather important for building's energy predictions. Therefore, expensive data pre-processing and feature engineering must be employed to compensate. LSTMs, on the other hand, are designed with mechanisms, such as gated memory cells, which enable them to retain information about past states and to model medium-term temporal dependencies, especially for time series energy data with low variability and consistent patterns. However, the models face several challenges that limit their effectiveness in building energy forecasting, such as difficulties in sequence modeling, which often require the integration of other models like CNNs and RNNs. Additionally, the models are exposed to the vanishing gradient problem, and their performance tends to decrease when confronted with large datasets with noisy data. The classical transformer models rely on self-attention mechanisms and positional encodings to efficiently represent the relationships over long energy time series sequences. This is achieved by dynamically assigning weights to different time instances in the series, allowing the transformer model to focus on the most important and relevant elements of the input energy data sequence. Positional encodings, on the other hand, provide the model with temporal information by capturing dependencies among distant time steps, making them suitable for building energy predictions. However, they face limitations in computational efficiency, being more resource-dependent, and struggle with high energy data variability, especially in the case of small-scale energy data. Additional mechanisms are needed to better understand the nonlinearity in building's energy consumption and capture finer grain temporality more efficiently. Our hybrid transformer model addresses the challenges exposed above by integrating adaptive encoding, reservoir processing and LNNs. The adaptive encoding combines sinusoidal, learnable and temporal encodings for better representation of temporal patterns and capturing complex temporal within the building's energy data. The inclusion of LNNs introduces controlled non-linearity, enabling the model to better capture complex relationships between input elements, and to reduce the computational overhead of our transformer model. Finally, the CNN

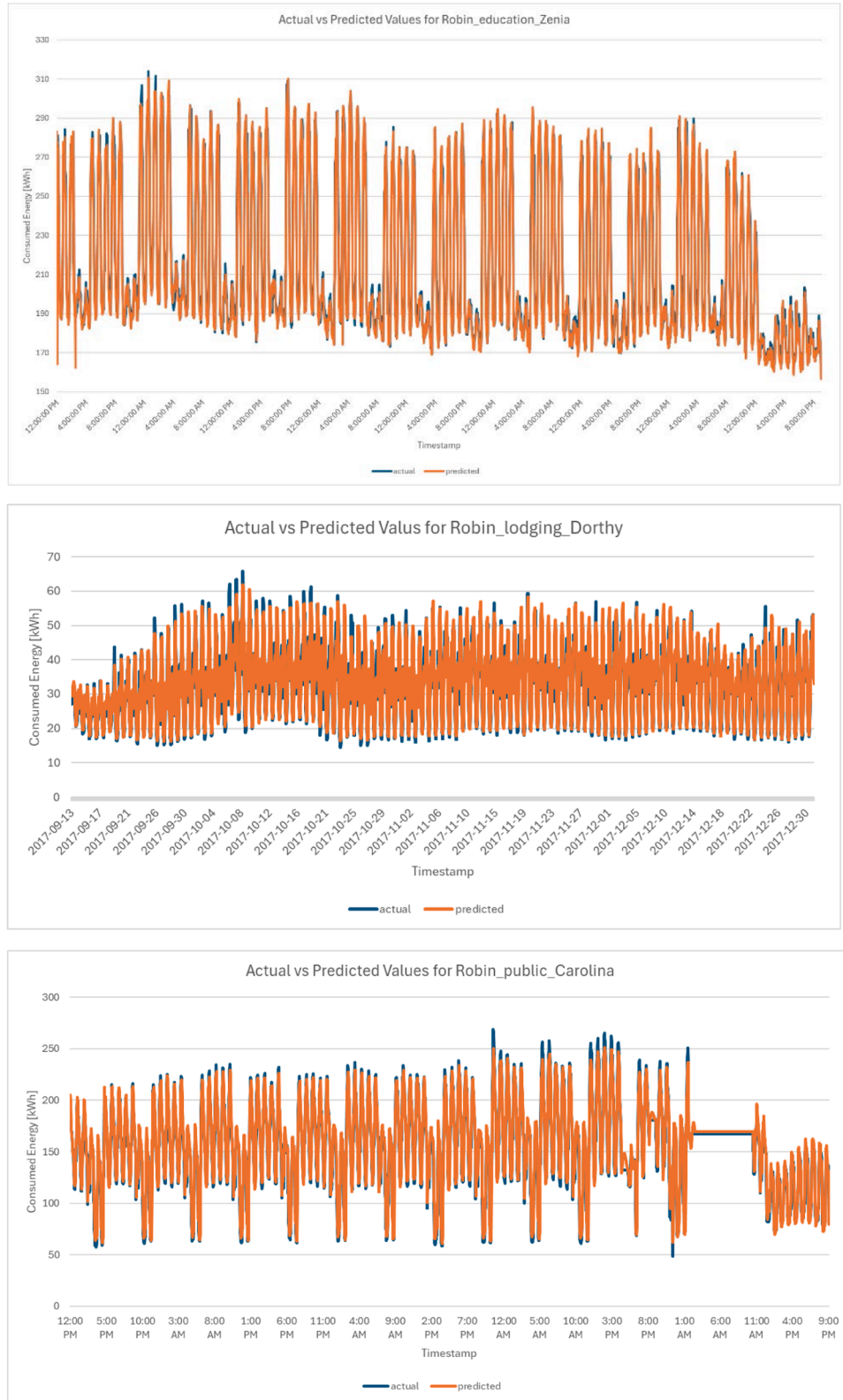


Fig. 13. Actual vs predicted energy profiles for the three commercial buildings.

encoder improves the model's capacity to process spatial patterns to improve prediction accuracy.

To benchmark the energy prediction accuracy of our hybrid transformer model, we compared it against LSTM, a basic transformer, and ANN. All models were trained using sequences of 24-hour energy

consumption data for the various building cases discussed in Section 3, and then used to predict the next 24 energy values (see Table 15). The models were optimized using techniques such as model checkpoints, learning rate reduction, and early stopping to prevent overfitting and enhance generalization. The LSTM model features three layers with

Table 14

Fundamental mechanism of prediction models for building's energy prediction.

Features		Model			
		Basic transformer	LSTM	ANN	Hybrid transformer
Core mechanisms		Self-Attention & Positional Encoding	Gated memory cells	Layers of interconnected neurons	Adaptive encoding with reservoir processing
Building energy prediction	Advantage	Captures relationships over long sequences	Memory of previous steps in a sequence.	Easier to implement	Better accuracy and training time than basic transformers
	Limitation	Memory & computational overhead	Vanishing gradient and difficulties in processing long sequences	Lack of sequence modelling	Higher Memory & computational overhead than LSTM & ANN

decreasing units (40, 20, and 10), followed by a dense layer to produce the energy prediction results. The LSTM showed a higher median sMAPE and lower R^2 compared to the proposed method, indicating less accurate and reliable predictions. The LSTM model demonstrated some capability in capturing temporal dependencies, as in the case of Portugal households that exhibit consistent energy consumption patterns. However, the LSTM models struggled with the variability and complexity inherent in energy consumption patterns. The ANN model features a structure with three Dense layers, with 120, 60 and 30 neurons, with ReLU as activation function. ANN performed well in capturing simple linear patterns and indicated a degree of computational efficiency on small datasets with lower variability, as observed in the residential buildings and households' test cases. For residential buildings, ANN prediction model resulted in an sMAPE of 10.22 %, while in households it recorded an sMAPE of 10.04 %, comparable with the other models. However, the model failed to consider the more complex patterns within the consumption data, leading to suboptimal performance in scenarios with high variability, such as in apartment buildings, where sMAPE increased to 21.42 %, in comparison to 9.64 % obtained by the hybrid transformer model.

The basic transformer uses sinusoidal non-cyclical positional encoding, struggled with prediction accuracy in several cases. This was evident for building test cases with longer input sequences due its reliance on static encodings that limited its adaptability. In the apartment buildings case, the prediction model has lower accuracy (i.e., 20.11 % sMAPE), compared to our hybrid transformer (i.e., 9.64 sMAPE). The model is less capable of identifying periods with sudden changes in energy data, as demonstrated in households cases, where MAE increased to 78.6 from 44.8 and R^2 decreased from 0.91 to 0.81. The basic transformer model could not interpret all the variance in the data, especially periods with peaks or valleys. Moreover, it achieved worse accuracy results compared to the LSTM model and, in some cases, underperformed even the ANN. Our hybrid transformer model demonstrated advantages over all above models in terms of accuracy. The integration of LNNs and learnable encodings allows for better management of complex relationships and fluctuations in building energy data,

as evidenced by its error metrics in apartment buildings (i.e., sMAPE of 9.64 %) and households (i.e., sMAPE of 9.26 %). The CNN encoder enhanced the model's ability to process the locality in the energy data inputs, improving the capturing of temporal analysis features, particularly in commercial buildings, with an improvement of approximately 25 % than the next model in terms of sMAPE. Additionally, our model surpassed the ANN and LSTM in terms of accuracy in all test cases, confirming its superiority in handling diverse load consumption scenarios, even though it struggled with predictions during periods of abrupt changes in the consumption patterns.

We compared the training times of all the considered models to evaluate the trade-offs between accuracy and time overhead for building energy prediction. The proposed hybrid transformer model is more computationally intensive than simpler architectures like LSTM and ANN, however it improves the prediction accuracy for all building cases considered. ANN is the most computationally efficient due to its simplicity leading to shorter training times, which makes them suitable for scenarios in which resources are constrained. However, it provides less accurate energy prediction results in all building cases than both LSTM and the proposed model, as it can be seen in Table 16. LSTM achieved better prediction accuracy than ANN in all test cases, particularly for apartment buildings, with a 23.81 % improvement in sMAPE, and for commercial buildings, with a 13 % improvement. However, despite their relatively fast training times, compared to transformer-based models, LSTMs are impacted by the vanishing gradient problem. Consequently, their predictive accuracy can be limited, even though their computational efficiency represents an advantage, as evidenced by the shorter training times. The basic transformer model are computationally expensive due to the quadratic complexity of their attention mechanism and the need for multiple stacked layers. Training transformer models takes significantly longer than LSTM and ANN (on average, 3.5 higher training time than LSTM and 6.5 higher training time than ANN), but generates better prediction accuracy results in few situations not sufficiently justify the need for such intensive computational needs. The hybrid Transformer model follows the trend of having longer training times than the traditional methods, but despite this, it improves significantly the energy prediction accuracy. On average, the proposed model shows an improvement of approximately 26.3 % in training time compared to the basic transformer, showcasing the advantage of the hybrid architecture in both accuracy and resource efficiency, making it relevant for buildings energy demand forecasting. This makes the hybrid transformer model suitable for applications where building energy prediction accuracy is prioritized over the training time and resources overhead. LSTM models, on the other hand, offer a balance between accuracy and computational efficiency. The ANN models, despite being the fastest, show slightly reduced precision, making them more appropriate when rapid predictions are needed and some compromise on accuracy is acceptable.

The proposed hybrid transformer model presents important generalization features across various types of buildings (i.e., apartment buildings, household prosumers, and commercial buildings), showing good energy prediction accuracy. Several measures can be implemented to enhance its generalization capabilities further and extend its applicability to other types of energy assets. To better account for the energy

Table 15

Models' average accuracy for energy demand prediction.

Building Type	Prediction Model	sMAPE (%)	R^2	MAE
Apartments Building	LSTM	16.32 %	0.81	57.57
	ANN	21.42 %	0.69	71.66
	Basic Transformer (BT)	20.11 %	0.86	62.49
	Hybrid Transformer (HT)	9.64 %	0.92	36.25
Residential Building	LSTM	9.08 %	0.74	0.98
	ANN	10.22 %	0.72	1.08
	Basic Transformer (BT)	11.94 %	0.64	1.27
	Hybrid Transformer	9.06 %	0.77	0.94
Households (Portugal)	LSTM	9.63 %	0.90	68.74
	ANN	10.04 %	0.83	70.29
	Basic Transformer (BT)	12.6 %	0.81	78.6
	Hybrid Transformer	9.26 %	0.91	44.8
Commercial Building	LSTM	7.25 %	0.84	7.93
	ANN	8.34 %	0.78	10.24
	Basic Transformer (BT)	7.70 %	0.80	9.37
	Hybrid Transformer	5.39 %	0.92	6.64

Table 16

Models training time for energy demand prediction in seconds.

Apartments building				Residential building				Households (Portugal)				Commercial building			
LSTM	ANN	BT	HT	LSTM	ANN	BT	HT	LSTM	ANN	BT	HT	LSTM	ANN	BT	HT
76	48	356	272	167	58	524	402	189	114	754	416	136	93	281	243

data temporality, the adaptive encoding module can be fine-tuned for different time series granularities. In the experiments mentioned above, all building energy data had one-hour granularity. The encoding module requires adjustments to capture both medium-term and long-term dependencies. In the current version, the cyclical part of sinusoidal encoding considers mostly weekly patterns. Further improvement can be achieved by integrating domain-specific information, such as weather-based seasonality or social factors that can impact the energy consumption patterns of buildings. The integration of LNNs improves the generalization features by considering broader types of non-linearities to better identify and handle sudden changes in load consumption. Finally, the self-attention mechanism can present a dynamic scaling factor across all queries and keys. Based on the variability within the energy data, the number of attention heads can also be adjusted for each building scenario.

5. Conclusions and future work

In this work, we developed a hybrid transformer model, integrating LNN, CNN, and learnable encoding to forecast the energy consumption of various types of buildings. The CNNs were introduced for input sequences processing to enhance feature representations, for the model to better capture spatial and local dependencies, before moving to temporal analysis. Additionally, LNN and learnable encodings, provide a non-linear component into the model architecture addressing the high variability and long temporal dependencies in energy consumption data. A Reservoir Processing module implemented using Liquid Neural Networks is used to capture more effective complex non-linear energy data patterns that can be masked by noise and fluctuations.

The experimental results show that our model achieved better accuracy compared to basic transformer, LTSM and ANN and faster convergence times than basic transformer. Moreover, it shows good generation features as we have utilized the model on both pilot data and state-of-the-art datasets storing the energy consumption of different building contexts, such as large buildings of apartments, commercial building or small households with and without on-site energy production. However, the model showed some limitations during periods with abrupt changes in energy usage. These results suggest that further refinement is necessary to efficiently handle these outlier patterns. Furthermore, the hybrid transformer models require more computational resources than traditional models such as LTSM and ANN, although convergence time was reduced due to the addition of LNNs, compared to basic transformer. However, despite the need for additional resources, the overall improvement in terms of prediction accuracy metrics is significant making it a useful tool for both building and grid operators.

Regarding future work, we aim to improve the model's capacity to handle anomalies and outlier patterns by including supplementary factors such as socio-economic influences and consumer behaviors and anomaly detection techniques to manage irregularities in energy data. Methods need to be developed to tailor the input energy data to minimize the impact of outliers or better consider for anomalous energy behaviors. Data augmentation can enhance the model's ability to generalize by simulating a wider range of scenarios, including anomalies in sensor data and extreme energy usage patterns. Various simulation techniques can be employed to evaluate extreme cases that are under-represented in the dataset while adding random noise can help simulate sensor inaccuracies. Finally, the hybrid transformer's attention

mechanisms can be enhanced to more effectively handle sudden changes or anomalies in energy usage. Attention mechanisms can be combined with physics-informed models to consider physics principles and domain-specific knowledge for building energy demand prediction, could further improve the model's adaptability and generalization capacity. Furthermore, it may improve its ability to capture the relationship between physical phenomena and building energy demand.

CRediT authorship contribution statement

Gabriel Antonesi: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology. **Tudor Cioara:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Conceptualization. **Ionut Anghel:** Writing – review & editing, Writing – original draft, Supervision, Conceptualization. **Ioannis Papias:** Writing – review & editing, Writing – original draft, Software, Data curation. **Vasilis Michalakopoulos:** Writing – review & editing, Writing – original draft, Methodology. **Elissaios Sarmas:** Writing – review & editing, Writing – original draft, Project administration.

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.

Acknowledgments

This work has been conducted within the DEDALUS project grant number 101103998 funded by the European Commission as part of the Horizon Europe Framework Programme and within Ministry of Research, Innovation and Digitization, CNCS/CCCDI - UEFISCDI, project number PN-IV-P8-8.1-PRE-HE-ORG-2023-0111, within PNCDI IV.

Data availability

The link to the LCL dataset is within the manuscript and the other datasets utilized are described in detail.

References

- [1] Pop C, Cioara T, Antal M, Anghel I, Salomie I, Bertoncini M. Blockchain based decentralized management of demand response programs in smart energy grids. *Sensors* 2018;18.
- [2] Skaloumpakas P, Sarmas E, Rachmanidis M, Marinakis V. Reshaping the energy landscape of Crete through renewable energy valleys. *Sci Rep* 2024;14(1):8038.
- [3] Amin A, Oudom K, Gallegos P, Chervet P, Ksontini F, Mourshed M. Demand response in buildings: unlocking energy flexibility through district-level electro-thermal simulation. *Appl Energy* 2022;305.
- [4] Todorean L, Cioara T, Anghel I, Sarmas E, Michalakopoulos V, Marinakis V. Demand response optimization for smart grid integrated buildings: review of technology enablers landscape and innovation challenges. *Energy Build* 2024; 115067.
- [5] Wang R, Bai L, Rayhana R, Liu Z. Personalized federated learning for buildings energy consumption forecasting. *Energy Build* 2024;323.
- [6] Marzak Z, Benabbou R, Mouatassim S, Benhra J. Forecasting seasonal and trend-driven data: a comparative analysis of classical techniques. *J Optim Ind Eng* 2023; 16(2):49–62.
- [7] Zhang PG. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing* 2003;50:159–75.
- [8] Zhao X, Zhao P, Zhu L, Zhang G. A comparison of multivariate and univariate time series models applied in tree sap flux analyses. *Forest Sci* 2023;68(5–6):473–86.

- [9] Maleki N, Lundström O, Musaddiq A, Jeansson J, Olsson T, Ahlgren F. Future energy insights: time-series and deep learning models for city load forecasting. *Appl Energy* 2024;374.
- [10] Hao W, Fang Q, Xi Z. User-side net load forecasting based on wavelet packet decomposition and least squares support vector machine. *Mod Electr Power* 2023; 40(2):192–200.
- [11] Tan Z, Zhang J, He Y, Zhang Y, Xiong G, Liu Y. Short-term load forecasting based on integration of SVR and stacking. *IEEE Access* 2020;8:227719–28.
- [12] Nie P, Rocotelli M, Fanti MP, Ming Z, Li Z. Prediction of home energy consumption based on gradient boosting regression tree. *Energy Rep* 2021;7: 1246–55.
- [13] Tang H-S, Wu L-A. Short-term load forecasting based on improved stochastic forest. *Short-term load forecasting based on improved stochastic forest*. In: 2020 International Conference on Virtual Reality and Intelligent Systems (ICVRIS); 2020.
- [14] Wu X, He J, Zhang P, Hu J. Power system short-term load forecasting based on improved random forest with grey relation projection. *Dianli Xitong Zidonghua/Autom Electr Power Syst* 2015;39:50–5.
- [15] Casolaro A, Capone V, Iannuzzo G, Camastra F. Deep learning for time series forecasting: advances and open problems. *Information* 2023;14.
- [16] Sarker I. Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. *SN Comput Sci* 2021;2.
- [17] Noh S-H. Analysis of gradient vanishing of RNNs and performance comparison. *Information* 2021;12(11):442.
- [18] Sarmas E, Strompolas S, Marinakis V, Santori F, Bucarelli MA, Doukas H. An incremental learning framework for photovoltaic production and load forecasting in energy microgrids. *Electronics (Basel)* 2022;11(23):3962.
- [19] Liu C, Jin Z, Gu J, Qiu C. Short-term load forecasting using a long short-term memory network. In: 2017 IEEE PES Innovative Smart Grid Technologies Conference Europe, ISGT-Europe 2017 - Proceedings; 2017.
- [20] Bouktif S, Fiaz A, Ouni A, Serhani MA. Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: comparison with machine learning approaches. *Energies (Basel)* 2018;7(7):11.
- [21] Sarmas E, Spiliotis E, Stamatopoulos E, Marinakis V, Doukas H. Short-term photovoltaic power forecasting using meta-learning and numerical weather prediction independent Long Short-Term Memory models. *Renew Energy* 2023; 216.
- [22] Michalakopoulos V, Sarantinopoulos E, Sarmas E, Marinakis V. Empowering federated learning techniques for privacy-preserving PV forecasting. *Energy Rep* 2024;12:2244–56.
- [23] Hu Z, Gao Y, Sun L, Mae M, Imaizumi T. Self-learning dynamic graph neural network with self-attention based on historical data and future data for multi-task multivariate residential air conditioning forecasting. *Appl Energy* 2024;364.
- [24] Sreekumar G, Martin JP, Raghavan S, Joseph CT, Raja SP. Transformer-based forecasting for sustainable energy consumption toward improving socioeconomic living: AI-enabled energy consumption forecasting. *IEEE Syst Man Cybern Mag* 2024;10:52–60.
- [25] Wang C, Wang Y, Ding Z, Zheng T, Hu J, Zhang K. A transformer-based method of multienergy load forecasting in integrated energy system. *IEEE Trans Smart Grid* 2022;13:2703–14.
- [26] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Polosukhin I. Attention is all you need. *Adv Neural Inf Process Syst* 2017.
- [27] Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H, Zhang W. Informer: beyond efficient transformer for long sequence time-series forecasting. In: AAAI Conference on Artificial Intelligence; 2021.
- [28] Oliveira H, Oliveira H. Transformers for energy forecast. *Sensors* 2023;23.
- [29] Sarkar MR, Anavatti SG, Dam T, Pratama M, Kindhi BA. Enhancing wind power forecast precision via multi-head attention transformer: an investigation on single-step and multi-step forecasting. In: 2023 International Joint Conference on Neural Networks (IJCNN); 2023. p. 1–8.
- [30] Wen Q, Zhou T, Zhang C, Chen W, Ma Z, Yan J, Sun L. Transformers in time series: a survey. In: Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23); 2023.
- [31] Zhang H, Li B, Su S-F, Yang W, Xie L. A novel hybrid transformer-based framework for solar irradiance forecasting under incomplete data scenarios. *IEEE Trans Industr Inform* 2024;20.
- [32] Saeed F, Aldera S. Adaptive renewable energy forecasting utilizing a data-driven PCA-transformer architecture. *IEEE Access* 2024;12.
- [33] Xu C, Chen G. Interpretable transformer-based model for probabilistic short-term forecasting of residential net load. *Int J Electr Power Energy Syst* 2024;155.
- [34] Li N, Dong J, Liu L, Li H, Yan J. A novel EMD and causal convolutional network integrated with Transformer for ultra short-term wind power forecasting. *Int J Electr Power Energy Syst* 2023;154.
- [35] Harrou F, Dairi A, Dorbane A, Sun Y. Enhancing wind power prediction with self-attentive variational autoencoders: a comparative study. *Results Eng* 2024;2.
- [36] Yu H, Chen S, Chu Y, Li M, Ding Y, Cui R, Zhao X. Self-attention mechanism to enhance the generalizability of data-driven time-series prediction: a case study of intra-hour power forecasting of urban distributed photovoltaic systems. *Appl Energy* 2024;374.
- [37] Mo S, Wang H, Li B, Xue Z, Fan S, Liu X. Powerformer: a temporal-based transformer model for wind power forecasting. *Energy Rep* 2024;11:736–44.
- [38] Chen, F., Yan, J., Bertling, L., Song, D., Yan, Y., & Liu, Y. (2023). Medium-term wind power forecasting based on dynamic self-attention mechanism. 2023 IEEE BelgradePowerTech, PowerTech 2023.
- [39] Fan J, Zhuang W, Xia M, Fang W, Liu J. Optimizing attention in a transformer for multihorizon, multienergy load forecasting in integrated energy systems. *IEEE Trans Industr Inform* 2024;20:10238–48.
- [40] Qu K, Si G, Shan Z, Kong X, Yang X. Short-term forecasting for multiple wind farms based on transformer model. In: 2021 The 2nd International Conference on Power Engineering (ICPE 2021); 2021.
- [41] Dai X, Liu G-P, Hu W. An online-learning-enabled self-attention-based model for ultra-short-term wind power forecasting. *Energy* 2023;272.
- [42] Wu H, Xu J, Wang J, Long M. Autoformer: decomposition transformers with auto-correlation for long-term series forecasting. In: Proceedings of the 35th International Conference on Neural Information Processing Systems; 2021.
- [43] Tian R, Li X, Ma Z, Liu Y, Wang J, Wang C. LDformer: a parallel neural network model for long-term power forecasting. *Front Inf Technol Electron Eng* 2023;24(9).
- [44] Fu H, Zhang J, Xie S. A novel improved variational mode decomposition-temporal convolutional network-gated recurrent unit with multi-head attention mechanism for enhanced photovoltaic power forecasting. *Electronics (Basel)* 2024;13.
- [45] Bharti S, Saini VK, Kumar R, Vijayvargiya A. Attention mechanism in deep learning for wind power forecasting. In: 2022 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES); 2022.
- [46] Huang J, Kaewunruen S. Forecasting energy consumption of a public building using transformer and support vector regression. *Energies (Basel)* 2023;16.
- [47] Song Y, Tang D, Yu J, Yu Z, Li X. Short-term forecasting based on graph convolution networks and multiresolution convolution neural networks for wind power. *IEEE Trans Industr Inform* 2023;19(2).
- [48] Liu J, Fu Y. Renewable energy forecasting: a self-supervised learning-based transformer variant. *Energy* 2023;284.
- [49] Xiang L, Liu J, Yang X, Hu A, Su H. Ultra-short term wind power prediction applying a novel model named SATCN-LSTM. *Energy Convers Manag* 2021.
- [50] Sheng H, Chang Y, Yinpeng Q. Deep learning model-transformer based wind power forecasting approach. *Front Energy Res* 2023;10.
- [51] Ye H, Zhu Q, Zhang X. Short-term load forecasting for residential buildings based on multivariate variational mode decomposition and temporal fusion transformer. *Energies (Basel)* 2024;17.
- [52] Pan C, Zhang C, Ngai EC, Liu J, Li B. HALO: HVAC load forecasting with industrial IoT and local-global-scale transformer. *IEEE Internet Things J* 2024;11.
- [53] Padilha GA, Ko J, Jung JJ, Neto PS. Transformer-based hybrid forecasting model for multivariate renewable energy. *Appl Sci* 2022;12(21).
- [54] Wu X, Yang L, Xinhao F, Zhifang S, Qingchang W. Short-term wind power forecasting based on dual attention mechanism and gated recurrent unit neural network. *Front Energy Res* 2024;12.
- [55] Wang Q, Wang Y, Zhang K, Liu Y, Qiang W, Wen QH. Artificial intelligent power forecasting for wind farm based on multi-source data fusion. *Processes* 2023;11(5).
- [56] Li Zhang KM. Feature selection using firefly optimization for classification and regression models. *Decis Support Syst* 2018;106(0167–9236):64–85.
- [57] London datasets. (2015). <https://data.london.gov.uk/dataset/smartmeter-energy-use-data-in-london-households>.
- [58] Portugal households dataset. (2015). <https://archive.ics.uci.edu/dataset/321/electricityloaddiagrams20112014>.
- [59] Clayton Miller AK. The building data genome project 2, energy meter data from the ASHRAE great energy predictor III competition. *Sci Data* 2020.