

ORIGINAL RESEARCH

Open access

Analysis of Energy Efficiency and Savings in Petrochemical Processes Using Convolutional Neural Network-Based Transfer Learning with Multirate Sampling

Hassan Rahman^{1*}, Tariq Mahmood¹, Ali Raza²

Abstract

Improving energy efficiency in the petrochemical sector is vital for reducing energy use and mitigating environmental harm. A reliable model for energy efficiency can provide important guidance for decision-making on operational process adjustments. In real-world settings, however, the varying sampling frequencies common in petrochemical facilities make conventional methods for estimating energy efficiency less dependable and more challenging to apply to multirate datasets. To address this, the present study introduces a multi-channel convolutional neural network framework that leverages transfer learning of model parameters to enhance energy efficiency forecasting despite irregular sampling rates. The multi-channel design enables the extraction of distinct temporal patterns by applying convolutions across the time axis of the data. At the same time, transfer learning enables the model to adapt and capture novel input behaviors after initial training. The effectiveness of this approach for both predicting energy efficiency and evaluating potential savings was demonstrated through its implementation in a vinyl chloride monomer manufacturing scenario. Results indicate that the suggested model surpasses standard benchmark approaches and conventional CNN architectures in predictive precision and consistency, achieving an R^2 value of 0.97. Incorporating transfer learning helps maintain robust performance without major declines and improves the model's flexibility for continuous, real-time energy monitoring. In addition, an analysis of energy discrepancies identified substantial conservation opportunities, resulting in an average annual reduction in energy use of 7.25% and a corresponding decrease of 5,709 tons in CO₂ emissions.

Keywords Transfer learning, Energy efficiency prediction, Petrochemical process, Multirate prediction, Convolutional neural network

*Correspondence:

Hassan Rahman

hassan.rahman@gmail.com

¹ Department of Intelligent Grid Systems, Faculty of Engineering, University of Karachi, Karachi, Pakistan

² Department of Energy Analytics, Faculty of Engineering, National University of Sciences and Technology, Islamabad, Pakistan

Introduction

Since the 1800s, worldwide energy demand has increased steadily [1], and industrial activities account for 24.2% of total global greenhouse gas emissions [2]. Data from the Energy Information Administration's industrial energy overview highlight that the chemical processing sector ranks among the most energy-demanding fields [3], accounting for 37.77% of overall energy consumption in 2021 and generating roughly 500 million metric tons of output annually, as illustrated in **Figure 1**. Within this sector, expenses related to energy and raw inputs represent about two-thirds of total production costs [4]. As a key segment of the broader chemical industry, petrochemical operations face increasing requirements to support sustainable development objectives and move toward net-zero carbon emissions. Consequently, efforts have

intensified to forecast energy performance at large-scale facilities and identify energy conservation opportunities to cut operational expenses and reduce ecological footprints [5].

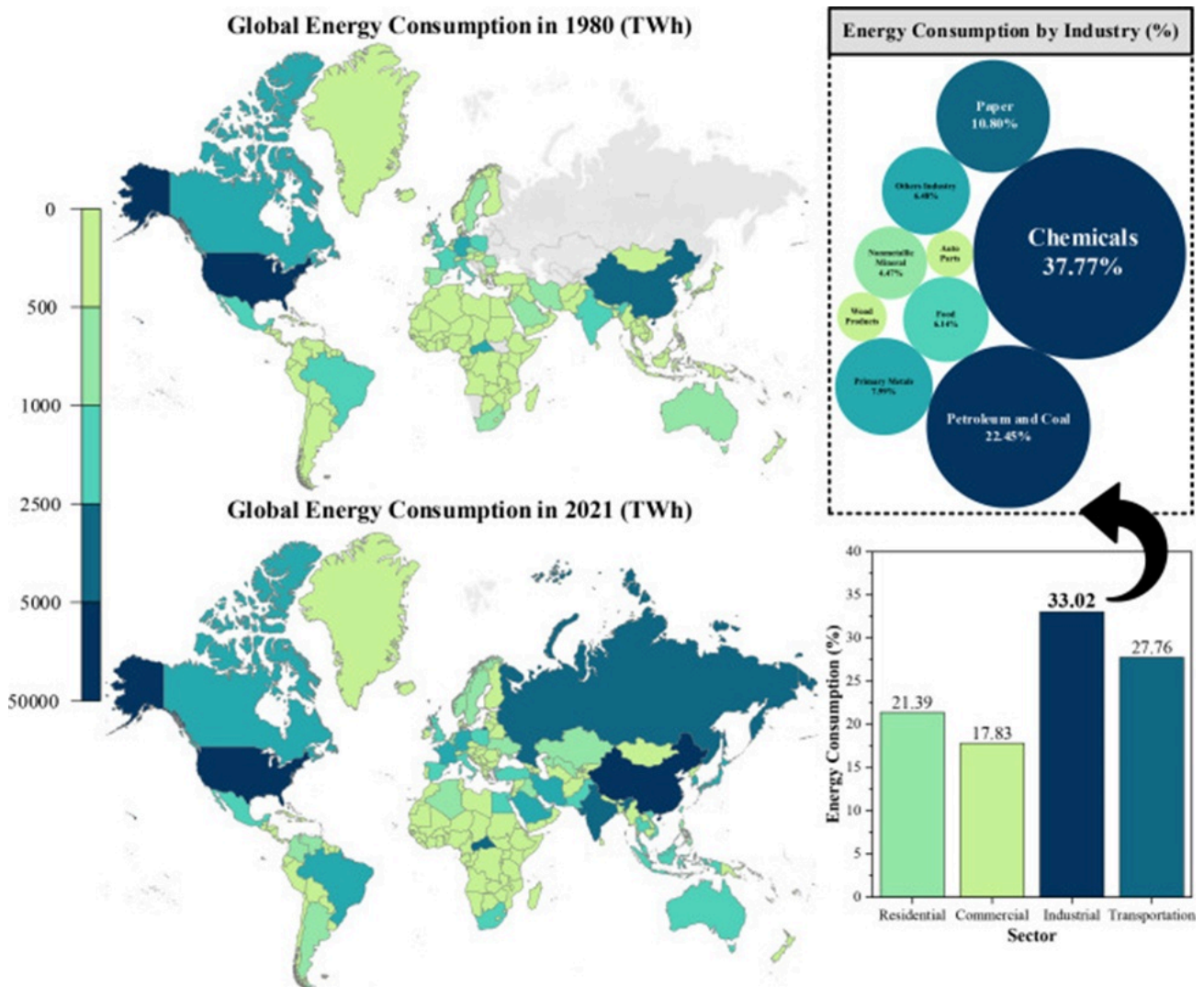


Figure 1. illustrates national, sectoral, and industry-level energy usage patterns.

In recent years, the rapid expansion of knowledge derived from large-scale data has led numerous scholarly efforts to assess energy use and potential reductions. Traditional mathematical modeling approaches, which depend on clearly defined variable interdependencies, remain sparsely validated in full [6]. While principal component analysis (PCA) offers a means to forecast energy performance, it struggles to identify an adequately comprehensive representation space [7]. Data envelopment analysis enables monitoring of energy performance and supports some degree of optimization [8], yet it faces challenges in managing parameter variability amid uncertain inputs [9]. Certain studies have used artificial neural networks (ANNs) to handle complex datasets, benefiting from their adaptable training approaches, ease of deployment, and strong performance in chemical-related applications [10]. Sauer and Walser employed convolutional neural networks (CNN) for near-term load predictions within industrial settings [11]. Machalek and colleagues introduced a gated recurrent unit (GRU) framework featuring an encoder-decoder design for horizon-based forecasting of energy in boiler steam systems [12]. Liu *et al.* introduced a GRU enhanced with an attention mechanism, allowing the system to emphasize particular segments of incoming data [13]. Prediction accuracy benefits from residual networks (RESNET), which incorporate skip connections within residual blocks to enable learning from both raw inputs and mid-level features. However, they remain susceptible to vanishing gradient difficulties [14]. The long short-term memory (LSTM) architecture,

thanks to its gating structure, effectively mitigates gradient-related issues [15]. LSTM configurations have proven capable of modeling dynamic shifts in disturbance patterns [16] and handling parameter variability [17]. Nevertheless, as time horizons lengthen, LSTM's capacity to represent extended dependencies weakens, reducing overall model dependability.

Transfer learning has gained substantial traction as a strategy to boost neural network performance in recognition tasks [18] while decreasing the volume of necessary training examples [19]. This technique facilitates the reuse of pre-trained models for novel applications or environments by adjusting model weights using limited target data [20], and it enhances model flexibility when input distributions diverge from those seen during initial training [21]. That said, broad system scopes frequently introduce high-dimensional challenges stemming from comprehensive plant-level data [22]. Convolutional neural networks (CNNs) excel at processing such multidimensional inputs owing to their robust feature-extraction capabilities [23]. Convolutional kernels scan different areas of the input, and the network's localized connections identify localized structures, which are then integrated into more abstract features [24]. Employing multi-channel CNN architectures can outperform conventional single-channel versions in pattern detection by leveraging richer information from diverse input streams [25], though these setups typically lack strong learning adaptability. Furthermore, artificial intelligence (AI) techniques are widely adopted for constructing energy-related models across diverse applications, including optimization, policy evaluation, and predictive maintenance to minimize unnecessary energy expenditure [26]. Still, due to specific data requirements and properties, relatively few instances are suitable for chemical engineering contexts [27].

Safety protocols, constraints on sensor placement, and inherent process traits often hinder uniform collection of measurable variables in industrial environments [28]. Data from chemical operations tends to be distinctive, scarce, and subject to uncertainty [29], rendering conventional machine learning methods less effective in chemical engineering scenarios [30]. In petrochemical settings, mixtures of qualitative and quantitative factors commonly necessitate offline evaluations to derive or verify additional metrics. Consequently, recorded variables exhibit uniqueness, and collection frequencies vary [31]. From the reviewed works, the majority of energy studies rely on consistent sampling cadences and fully populated datasets, leaving the issue of differing rates largely unaddressed [32]. Application of density-based spatial clustering of applications with noise (DBSCAN) can convert sparse datasets into evenly spaced ones [33]. However, this approach substantially diminishes usable data volume, leading to constrained and inadequate sample sizes. Thus, the actual effectiveness of AI-driven, facility-wide energy performance and conservation evaluations that account for multi-time-scale data issues has received limited exploration [34], thereby motivating the current investigation.

Accordingly, the present work seeks to bridge the identified shortfall by creating a real-time predictive system for energy performance, pinpointing conservation opportunities, and tackling challenges posed by predictors operating across varied time scales through the development of a multi-channel transfer learning convolutional neural network (TLCNN) architecture. Conventional energy performance evaluation typically requires synchronized, complete input sets when sampling rates differ, precluding genuine real-time capability. The TLCNN approach demonstrated herein enables the AI framework to accommodate shifts in process conditions or data characteristics post-training for petrochemical energy performance forecasting. Moreover, the method operates effectively without preliminary data treatment and preserves the full extent of available observations. Validation and reproducibility are demonstrated using the vinyl chloride monomer (VCM) production process as a representative petrochemical example. In these circumstances, the work provides fresh insights into forecasting and framework design for AI practitioners aiming to advance energy performance monitoring in sophisticated petrochemical settings by overcoming inconsistencies in sampling frequencies. The primary advancements are outlined as follows:

1. Introduce an innovative technique for forecasting energy performance amid varying sampling intervals (such as 1, 5, and 10 minutes) in a petrochemical application, along with a multi-channel CNN design that allows individual channels to discern distinct data patterns.
2. Determine key variables driving energy consumption that reflect overall plant energy generation dynamics in the VCM process. A combined strategy incorporating tree-based selection and forward stepwise procedures is applied to pinpoint

highly relevant inputs and eliminate redundant or irrelevant associations during model development.

3. Develop a TLCNN framework that enhances adaptability to post-training alterations in sampling frequency that deviate markedly from the original training conditions. Findings indicate that transfer learning markedly boosts consistency and repeatability in multi-scale forecasting contexts.

4. Evaluate discrepancies between predicted and actual energy levels to quantify opportunities for conservation, supporting more environmentally sustainable and efficient operations.

Framework for Model Construction

Figure 2 depicts the overall model construction process, encompassing variable selection, dataset splitting, model architecture definition, error assessment, and various validation scenarios. Each component is elaborated in the subsections below.

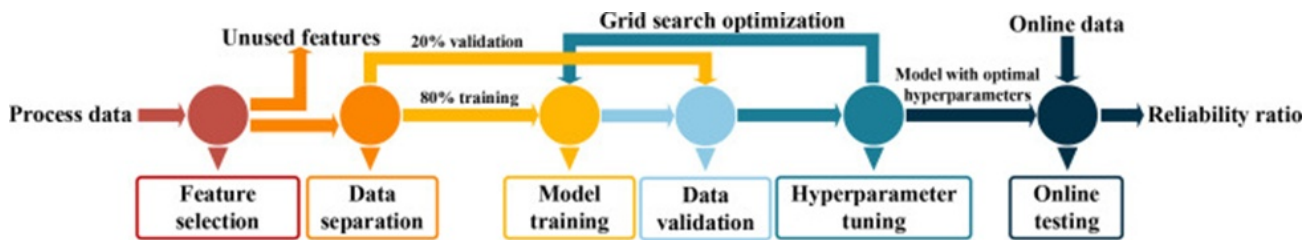


Figure 2. Overview of the model construction process in this research.

Variable selection

Forecasting energy performance in extensive industrial operations requires sophisticated variable selection strategies to identify variables with strong correlations and substantial potential for energy conservation [35]. This investigation adopts a dual-method variable selection strategy. The initial method relies on tree-based models to estimate predictor relevance, as decision trees—a form of model-dependent selection—excel at pinpointing essential variables [36]. The predictor importance score reflects the comparative significance of input variables in estimating the target output, namely energy efficiency here. Feature importance (FI) is determined by aggregating the node improvements from splits for each predictor, then normalizing by the total number of splits across the trees. Predictors exhibiting negligible importance are discarded. The computation of feature importance follows **Eqs. 1 and 2**.

$$NI_j = w_j e_j - (w \cdot e)_{child,j} \quad (1)$$

$$FI_i = \frac{\sum_{j: \text{Node } j \text{ split on feature } i} NI_j}{\sum_{k \in \text{all nodes}} NI_k} \quad (2)$$

Where w_j denotes the weighted number of samples arriving at node j , e_j represents the error at node j , NI_j indicates the importance score for node j , FI_i signifies the feature importance score for feature i , and k encompasses all tree nodes. In essence, a higher FI score signifies a stronger association between the given variable and energy efficiency.

The secondary method employs forward stepwise variable selection, beginning by training the model using the predictor with the most prominent FI score from the initial phase. Subsequently, the predictor with the next-highest FI score is incorporated in the following iteration. This process continues iteratively until all ranked predictors have been included.

Ultimately, the optimal subset of predictors that yields the maximum adjusted R^2 is selected as the definitive input set for model comparison. **Figure 3** depicts the structure of the variable selection approach implemented in this work.

The secondary method employs forward stepwise variable selection, beginning by training the model using the predictor with the most prominent FI score from the initial phase. Subsequently, the predictor with the next-highest FI score is incorporated in the following iteration. This process continues iteratively until all ranked predictors have been included. Ultimately, the optimal subset of predictors that yields the maximum adjusted R^2 is selected as the definitive input set for model comparison. **Figure 3** depicts the structure of the variable selection approach implemented in this work.

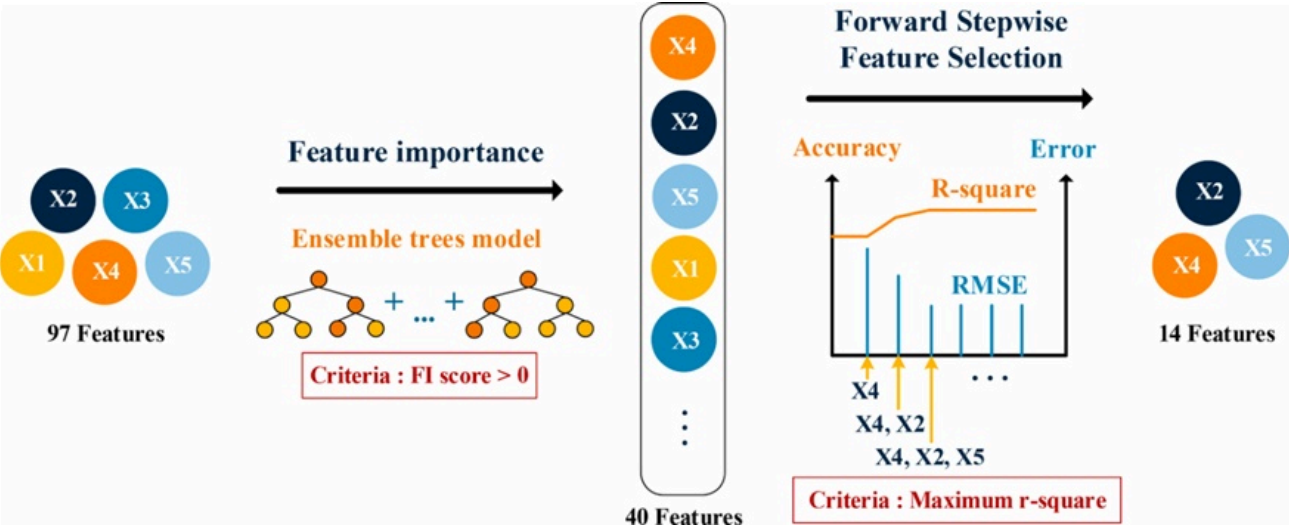


Figure 3. Structure of the variable selection process along with the elimination standards applied at every stage.

Multi-channel transfer learning convolutional neural network (TLCNN)

In the present research, the TLCNN architecture comprises five primary components: an input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer. The overall configuration of the TLCNN is illustrated in **Figure 4**, and the trainable parameters of the suggested model are detailed in **Table 1**. A description of each layer of the proposed architecture is provided in the following sections.

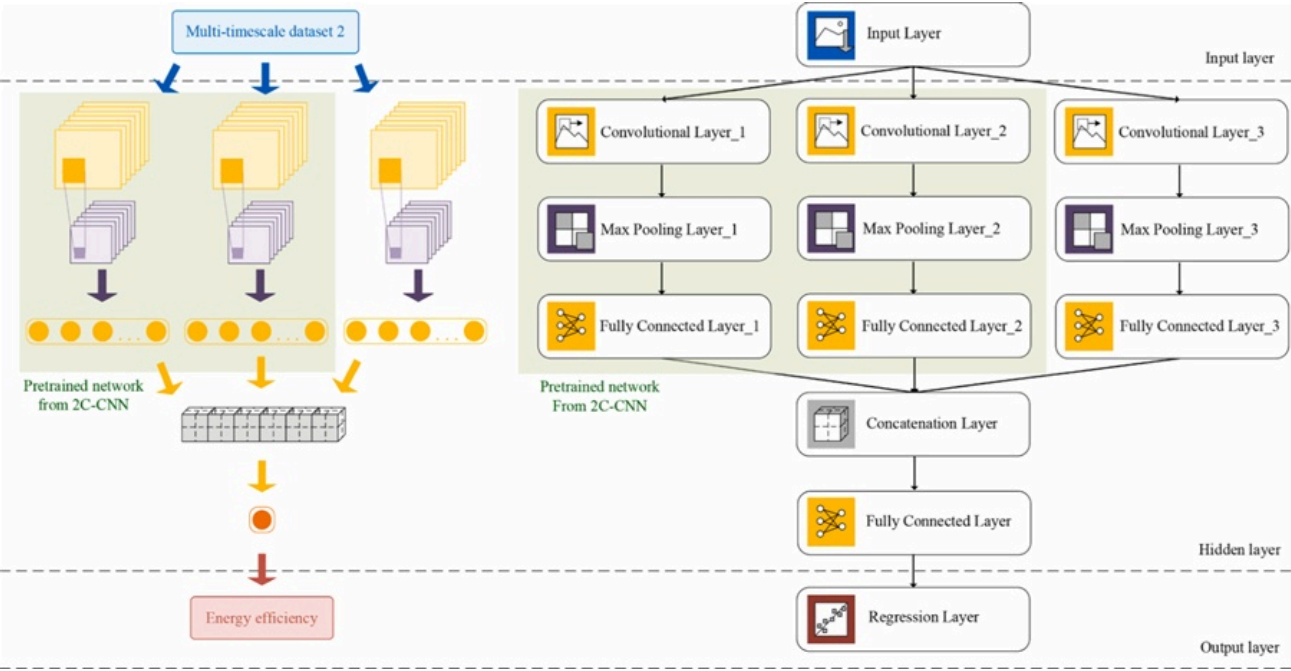


Figure 4. TLCNN model structure.

Table 1. The characteristics of the TLCNN model.

Name	Type	Activations	Learnable parameters	
Input Layer	Sequence Input	$14 \times 1 \times 1$	—	
$14 \times 1 \times 1$ with 'z-score' normalization				
Convolutional Layer_1	Convolution	$14 \times 1 \times N$	Weights	$10 \times 10 \times 1 \times N$
50 $10 \times 10 \times 1$ convolutions			Bias	$1 \times 1 \times N$
Max Pooling Layer_1	Max Pooling	$14 \times 1 \times N$	—	
3×3 max pooling				
Fully Connected Layer_1	Fully Connected	$1 \times 1 \times N$	Weights	$N \times 700$
50 fully connected layers			Bias	$N \times 1$
Convolutional Layer_2	Convolution	$14 \times 1 \times N$	Weights	$10 \times 10 \times 1 \times N$
50 $10 \times 10 \times 1$ convolutions			Bias	$1 \times 1 \times N$
Max Pooling Layer_2	Max Pooling	$14 \times 1 \times N$	—	
3×3 max pooling				
Fully Connected Layer_2	Fully Connected	$1 \times 1 \times 1$	Weights	$N \times 700$
1 fully connected layer			Bias	$N \times 1$
Convolutional Layer_3	Convolution	$14 \times 1 \times N$	Weights	$15 \times 15 \times 1 \times N$

50 $15 \times 15 \times 1$ convolutions			Bias	$1 \times 1 \times N$
Max Pooling Layer_3	Max Pooling	$14 \times 1 \times N$	–	
5×5 max pooling				
Fully Connected Layer_3	Fully Connected	$1 \times 1 \times N$	Weights	$N \times 700$
40 fully connected layers			Bias	$N \times 1$
Concatenation Layer	Concatenation	$1 \times 1 \times 3$	–	
Concatenate of 3 channels along dimension t				
Fully Connected Layer	Fully Connected	$1 \times 1 \times 1$	Weights	$1 \times N$
1 fully connected layer			Bias	1×1
Regression Layer	Regression Output	$1 \times 1 \times 1$	–	

*N is the final hyperparameter.

Input layer

The input layer of the network does not incorporate any weighting coefficients or bias terms. Its key benefit lies in enabling data normalization directly at this stage. In this work, z-score standardization is applied in the input layer to rescale all features to a consistent range, thereby enhancing training stability. This transformation is performed according to [Eqs. 3–5](#).

$$z_i^j = \frac{x_i^j - \mu^j}{\sigma^j} \quad (3)$$

$$\mu^j = \frac{\sum_{i=1}^n x_i^j}{n} \quad (4)$$

$$\sigma^j = \sqrt{\frac{\sum_{i=1}^n (x_i^j - \mu^j)^2}{n-1}} \quad (5)$$

Where x_i^j represents the original value of the i-th observation for the j-th input feature, μ^j denotes the mean of the j-th input feature, and σ^j indicates the standard deviation of the j-th input feature.

Convolutional layer

The convolutional layer contains trainable parameters that extract relevant features by applying convolutional operations to the incoming data. Convolution, as a mathematical operation, integrates the input with a set of learnable filters (also called kernels). For one-dimensional time-series inputs, the operation slides along the temporal axis. Here, the primary regression component employs convolutional filters to generate feature maps that autonomously identify critical patterns in an unsupervised manner. The resulting convolutional output is computed as shown in [Eq. 6](#).

$$y_j = f(x_i * w_k + b_k) \quad (6)$$

Where f represents the chosen activation function, w_k denotes the weight coefficients of the filter, and b_k is the corresponding bias term for the convolution kernel.

Pooling layer

Following the convolution step, the pooling layer reduces the spatial (or temporal) dimensionality of the feature maps by aggregating information from clusters of neurons into a single representative value in the subsequent layer. This layer operates without any learnable weights or biases. By downsampling each feature map, pooling substantially reduces the number of parameters to train in later stages, thereby shortening training time and mitigating the risk of overfitting. The two most widely used pooling operations are max pooling, which retains the maximum value within each defined window, and average pooling, which computes the mean across the window. The sequential processing flow through the convolution and pooling stages is depicted in **Figure 5**.

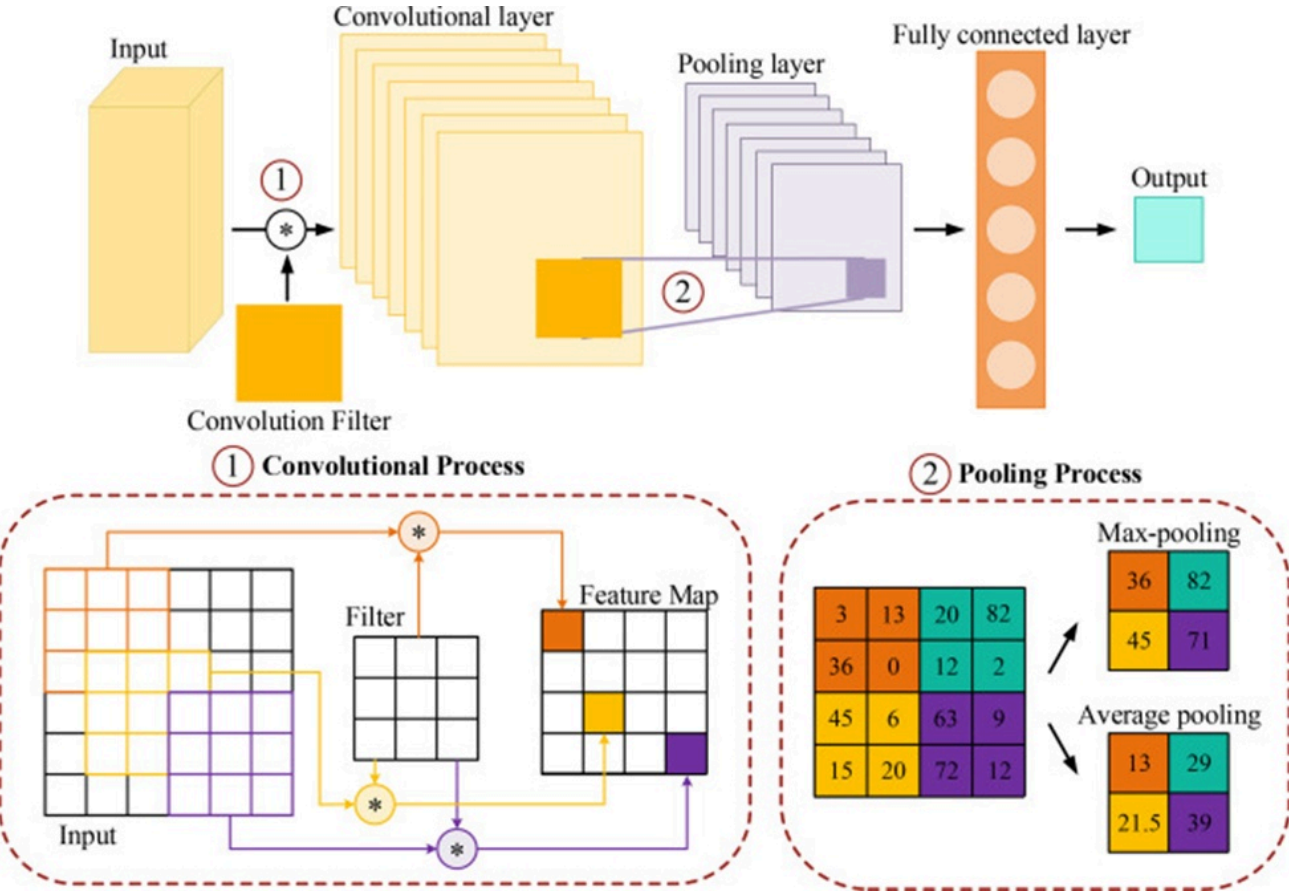


Figure 5. Illustration of the core principles underlying a convolutional neural network.

Fully connected layer

After downsampling by convolutional and pooling layers, the architecture incorporates one or more fully connected (dense) layers. These layers perform a linear transformation by multiplying the incoming flattened feature vector with a learnable weight matrix and then adding a bias vector. Additionally, the fully connected layer reshapes the output to match the dimensionality required for the final prediction.

Output layer

The output layer generates the final predicted value from the processed inputs and facilitates backward error propagation through the network during training. This backpropagation adjusts the model's learnable weights and biases to minimize

the chosen loss metric for the regression problem. In this work, a half-mean-squared error serves as the objective function guiding model optimization, computed according to Eq. 7. To identify the most effective hyperparameters for each layer, a grid search is performed, with validation R^2 as the performance criterion. The specific ranges and values explored during hyperparameter tuning are outlined in Table 2.

$$HMSE = \frac{1}{2N} \sum_{t=1}^N (y_{actual,t} - y_{predict,t})^2 \quad (7)$$

Table 2. Searching domains for hyperparameter tuning.

Layers	Hyperparameters	Values
Convolutional layer	Size of filters	[5, 20]
	Number of filters	[10, 60]
Pooling layer	Size of pooling regions	[1, 6]
	Pooling types	[Average, Max]
Fully connected layer	Number of nodes	[1, 60]
Training options	L2 Regularization	[0, 1]
	Learning rate	[0.25, 1]
	Learning rate drop period	[20, 100]
	Learning rate drop factor	[0.5, 1]
	Max epochs	[100, 500]
	Validation frequency	10
	Mini batch size	120
	Wight update optimizer	[Adam, SGD]

Transfer learning procedure

Transfer learning is an effective strategy to mitigate issues arising from random weight initialization and to enhance the consistency and reproducibility of model performance, particularly when processing high-dimensional datasets [37]. In this investigation, the parameter-based form of transfer learning was adopted, as it minimizes the requirement for extensive additional training data and significantly reduces overall computation time [38].

For the purpose of forecasting energy efficiency, a total of eight modeling approaches were assessed and compared: regression tree (RT), feed-forward neural network (FFNN), recurrent neural network (RNN), long short-term memory (LSTM), standard single-channel CNN, two-channel multi-channel CNN (2C-CNN), three-channel multi-channel CNN without transfer learning (3C-CNN), and the proposed TLCNN architecture. As baseline alternatives, data reconstruction methods based on linear interpolation and zero-order hold were also examined. Among the simplest reconstruction approaches, zero-order hold preserves the most recent observed value until a new measurement becomes available. At the same time, linear interpolation assumes a straight-line relationship between consecutive known points and estimates intermediate values accordingly. Nevertheless, both techniques extend the preprocessing time and make real-time application impractical, especially when future neighboring data points are not yet available.

The 2C-CNN configuration incorporates two parallel input channels, whereas the 3C-CNN employs three. In both cases, the initial channel mirrors the architecture of the baseline single-channel CNN. The hyperparameters for the second channel in 2C-CNN and the third channel in 3CCNN were determined through an extended grid search with additional optimization iterations.

In contrast, the proposed TLCNN leverages the pre-trained weights from the 2CCNN model for its first two channels, while dedicating a separate training phase to optimizing the third channel independently. This design promotes greater training stability by distributing the learning burden across multiple channels, enabling each to specialize in capturing distinct patterns present in the multi-rate input data. To facilitate effective adaptation, partial freezing is applied to the pre-existing portions of the network, compelling the newly added third channel to extract novel features from the incoming data. Concurrently, the parameters of the previously trained channels remain frozen to retain the knowledge acquired under the original sampling regime. At the same time, fine-tuning is restricted to the additional channel to adapt to the new, differing sampling frequencies. The detailed architecture and the stepwise transfer learning workflow are depicted in **Figure 6**.

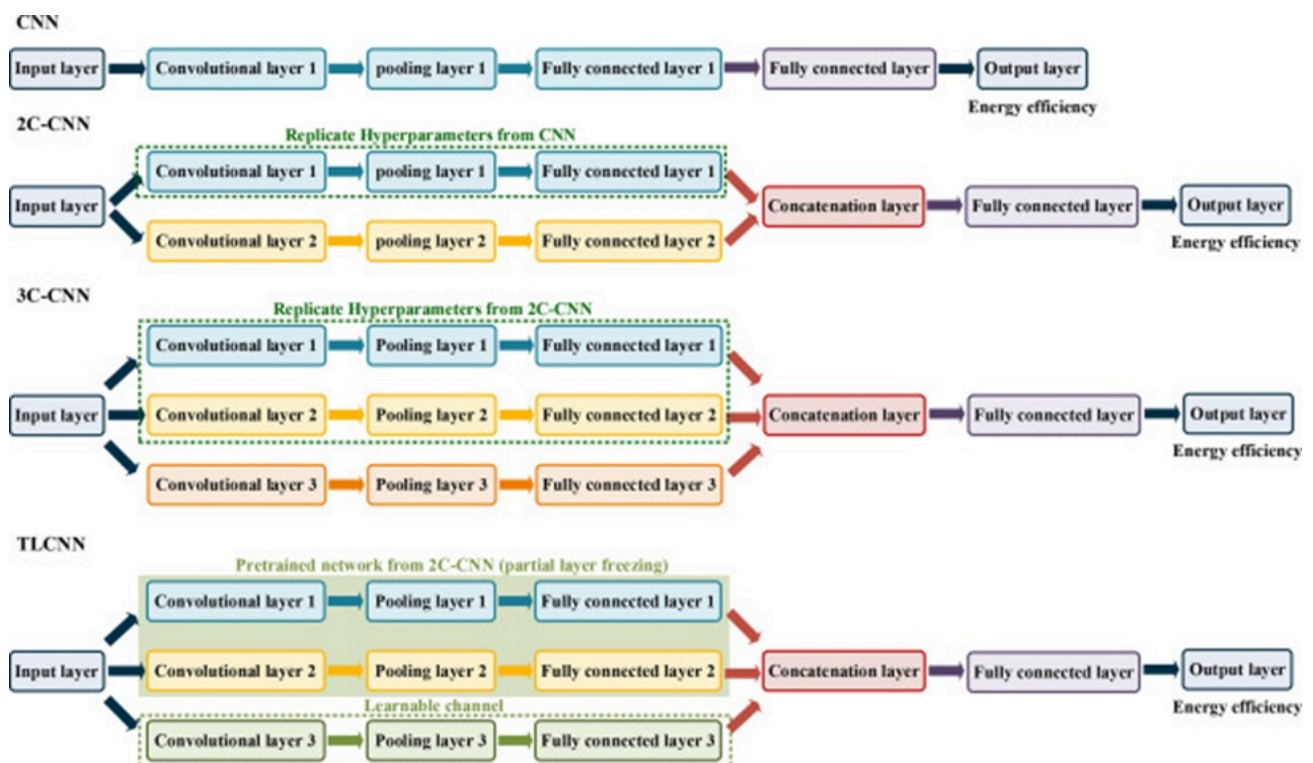


Figure 6. Architecture of the CNN model along with the transfer learning workflow.

Model evaluation

In the final phase, all developed models undergo a comprehensive assessment tailored for real-world deployment. This includes software-in-the-loop simulation to evaluate real-time performance, a reproducibility assessment to verify model consistency and compute a reliability metric, and a sliding window evaluation to simulate shifts in time-scale dynamics after initial training. Model performance is quantified using three key metrics: the coefficient of determination (R^2), root mean squared error (RMSE), and mean absolute percentage error (MAPE), as defined in **Eqs. 8–10**, respectively. Further elaboration on the specific testing protocols is provided in the results and discussion section.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (8)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y})^2} \quad (9)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}}{y_i} \right| \times 100\% \quad (10)$$

where y_i denotes the actual value for the i -th sample, $\hat{y}_i$ is the corresponding predicted value, and $\bar{y}$ represents the mean of the observed outputs. The R^2 statistic quantifies how well the model fits the data and reflects the fraction of variance in energy efficiency that can be accounted for by the selected input variables. In addition, RMSE captures the absolute magnitude of prediction errors and offers a more complete perspective on model accuracy than R^2 alone. In contrast, MAPE reports errors as percentages, making it less influenced by outliers than RMSE.

Vinyl Chloride Monomer (VCM) Production Process

The VCM manufacturing process comprises five primary units: direct chlorination, oxychlorination, ethylene dichloride (EDC) purification, EDC pyrolysis, and VCM purification. A schematic representation of the complete VCM process and its individual sections is presented in [Figure 7](#).

The production sequence starts with the introduction of ethylene and chlorine into the direct chlorination unit, where they undergo an addition reaction to yield EDC. In the oxychlorination unit, ethylene is reacted with oxygen and hydrogen chloride in the vapor phase to produce additional EDC; the resulting stream is then treated with sodium hydroxide in a caustic scrubber for neutralization. The treated mixture is then fed to a distillation column that separates EDC from water.

Purified EDC streams from three different origins—direct chlorination, oxychlorination, and recycled unreacted EDC from the downstream VCM purification—are combined and directed to the EDC purification section. Here, residual impurities such as water, ethyl chloride, and trichloroethane (TCE) are removed. The first distillation column removes light-end components, low-boiling impurities, and water, while the second column strips out heavier by-products, such as TCE.

The refined EDC is subsequently fed into a thermal cracking furnace equipped with extended tubular reactors, operating at temperatures between 480 and 530 °C and pressures ranging from 607.95 kPa to 3546.37 kPa. This high-temperature pyrolysis converts EDC into VCM along with various by-products. The hot effluent gases are cooled through a series of heat exchangers and condensers before entering the hydrogen chloride (HCl) separation column, which recovers and recycles unreacted HCl back to the oxychlorination unit as a feedstock for EDC synthesis. A subsequent distillation step isolates the VCM product from residual EDC, with the unreacted EDC returned to the purification section for further processing.

In summary, VCM production involves a series of critical transformations—including chlorination, thermal cracking, and multi-stage separations—all conducted under elevated temperature and pressure conditions. These operations require substantial energy input to sustain reaction conditions and support auxiliary utilities, making the VCM industry one of the most energy-intensive sectors of petrochemical manufacturing.

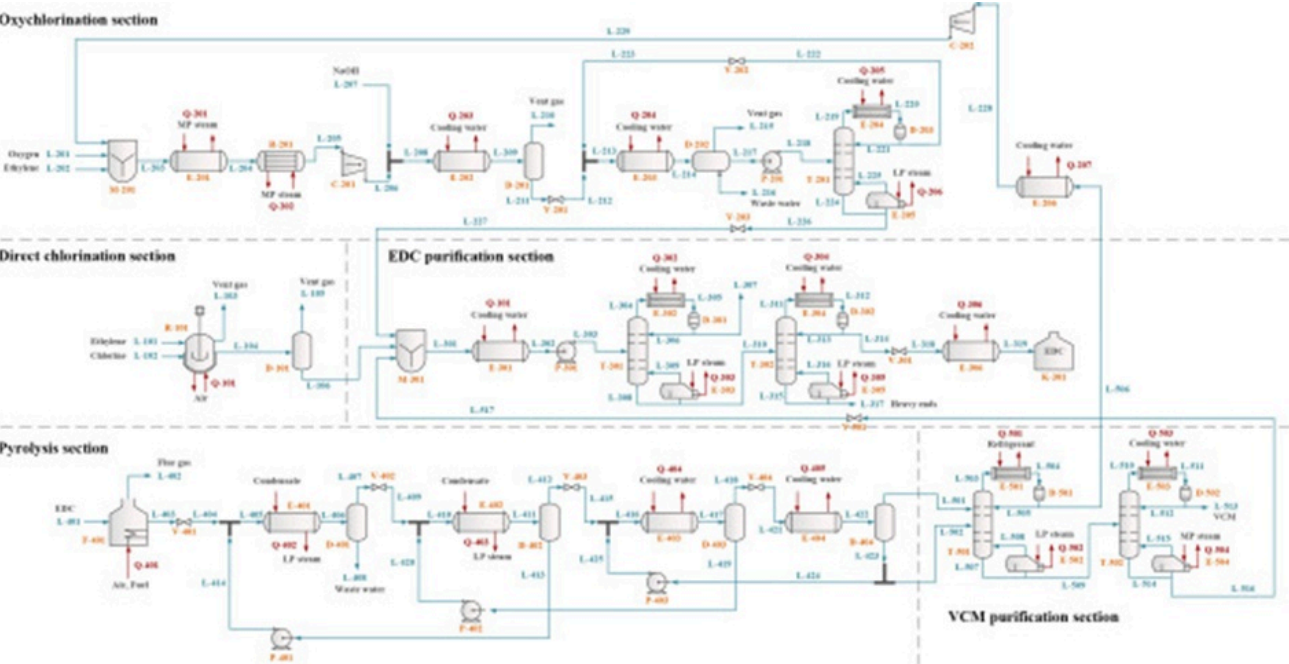


Figure 7. Comprehensive schematic of the VCM manufacturing process.

Multi-rate data characteristics

The dataset used in this investigation was generated using a co-simulation framework integrating MATLAB with UniSim Design Suite, drawing on operating parameters documented in the existing literature [39]. Within the scope of this work, the ethylene feed rate was varied from 235,000 to 410,000 kg/day (with the direct chlorination section contributing a minimum of 160,000 kg/day and the oxychlorination section contributing a minimum of 75,000 kg/day). Concurrently, the fresh EDC input to the pyrolysis unit was adjusted between 750,000 and 1,800,000 kg/day, with a random variation of ± 10 % around the specified values. A total of 2000 data points were simulated in the UniSim platform using a uniform 1-minute sampling frequency. Subsequently, these high-resolution data were resampled in MATLAB to create a multi-time-scale dataset as specified in Table 3, featuring constant sampling intervals of 1 minute, 5 minutes, and 10 minutes during the training phase.

The VCM process comprises 97 observable process variables. An illustrative example of the resulting multi-rate data structure is shown in Figure 8a. Notably, the lower-sampling-frequency variables exhibit gaps between consecutive measurements, leading to the loss of high-frequency dynamic information and making it difficult to capture and model the underlying behavioral patterns effectively. To accommodate these differing sampling rates, three distinct alignment scenarios were constructed from the available measurements: purely 1-minute data, a combination of 1-minute and 5-minute rates, and a full integration of 1-minute, 5-minute, and 10-minute rates. These alignment configurations are depicted schematically in Figure 8b.

Table 3. Description of process variables in the VCM process and sampling intervals.

No	Process variables	Interval	No	Process variables	Interval
1	Temperature of EDC at the bottom of D-101	1 min	50	CW feed rate of E-304	1 min
2	Mass flow rate at the bottom of R-101	1 min	51	CW feed rate of E-301	5 min
3	Mass flow rate at top D-101	5 min	52	CW feed rate of E-306	10 min

4	Mass flow rate of EDC at bottom D-101	1 min	53	LP steam feed rate of E-303	1 min
5	Mass flow rate of chlorine feed to R-101	1 min	54	LP steam feed rate of E-305	1 min
6	Mass flow rate of ethylene feed to R-101	1 min	55	Temperature of EDC feed to F-401 (1)	1 min
7	Air feed rate of R-101	5 min	56	Temperature of EDC feed to F-401 (2)	1 min
8	LP steam feed rate of R-101	1 min	57	Temperature at I-403	1 min
9	Temperature of recycled HCl at L-228	5 min	58	Temperature of recycled EDC D-402	1 min
10	Temperature of recycled HCl at L-229	1 min	59	Temperature at D-401 inlet	1 min
11	Temperature at C-201 outlet	10 min	60	Temperature at top D-401	1 min
12	Temperature at P-201 outlet	5 min	61	Temperature of recycled EDC D-403	1 min
13	Temperature at bottom D-201	10 min	62	Temperature at D-402 inlet	1 min
14	Temperature of recycled water at L-223	10 min	63	Temperature of recycled from D-404	1 min
15	Temperature of decanter D-202	1 min	64	Temperature at D-403 inlet	1 min
16	Temperature at top T-201	1 min	65	Temperature at D-404 inlet	1 min
17	Temperature at bottom T-201	1 min	66	Mass flow rate of EDC feed to F-401	1 min
18	Mass flow rate of ethylene feed to M-201	1 min	67	Mass flow rate at top D-401	1 min
19	Mass flow rate of oxygen feed to M-201	1 min	68	Mass rate of drainage at bottom D-401	10 min
20	Mass flow rate of EDC at bottom T-201	1 min	69	Mass flow rate at top D-402	5 min
21	Mass flow rate at L-208	10 min	70	Mass rate of recycling at bottom D-402	5 min
22	Mass flow rate of recycled HCl at L-229	1 min	71	Mass flow rate at top D-403	1 min
23	Mass flow rate at D-202 inlet	5 min	72	Mass flow rate at top D-404	5 min
24	Mass flow rate of NaOH at L-207	1 min	73	Mass rate of recycling at the bottom D-403	5 min
25	Mass flow rate at top D-201	5 min	74	Mass rate of recycled at bottom D-404	1 min
26	Mass flow rate at bottom D-202	5 min	75	Mass flow rate at P-403 outlet	1 min
27	Mass flow rate at P-201 inlet	1 min	76	Mass flow rate at D-402 inlet	5 min
28	Mass flow rate at top T-201	1 min	77	Mass flow rate at D-401 inlet	1 min
29	CW feed rate of E-202	1 min	78	CW feed rate of E-403	10 min
30	CW feed rate of E-203	5 min	79	CW feed rate of E-404	1 min
31	CW feed rate of E-204	1 min	80	Fuel feed rate of F-401	1 min
32	LP steam feed rate of E-205	1 min	81	LP steam feed rate of E-401	1 min
33	MP steam feed rate of E-201	1 min	82	LP steam feed rate of E-402	5 min
34	MP steam feed rate of R-201	5 min	83	Temperature at top T-501	1 min
35	CW feed rate of E-206	1 min	84	Temperature at bottom T-501	1 min
36	Power input of C-202	5 min	85	Temperature at top T-502	1 min

37	Temperature of EDC feed to E-301	10 min	86	Temperature at bottom T-502	1 min
38	Temperature of EDC feed to T-301	5 min	87	Temperature of recycled EDC at L-517	1 min
39	Temperature at top T-301	1 min	88	Mass flow rate at L-501	1 min
40	Temperature at bottom T-301	1 min	89	Mass flow rate at L-502	1 min
41	Temperature at top T-302	1 min	90	Mass flow rate at T-502 inlet	1 min
42	Temperature at bottom T-302	1 min	91	VCM mass flow rate	1 min
43	Temperature of purified EDC at K-301	10 min	92	Mass rate of recycling at bottom T-502	1 min
44	Mass flow rate of EDC feed to E-301	1 min	93	Mass rate of recycling at top T-501	1 min
45	Mass flow rate at top T-301	1 min	94	CW feed rate of E-503	1 min
46	Mass flow rate at bottom T-301	1 min	95	LP steam feed rate of E-502	1 min
47	Mass flow rate at top T-302	1 min	96	MP steam feed rate of E-504	1 min
48	Mass flow rate at bottom T-302	1 min	97	Refrigerant feed rate of E-501	1 min
49	CW feed rate of E-302	1 min		–	

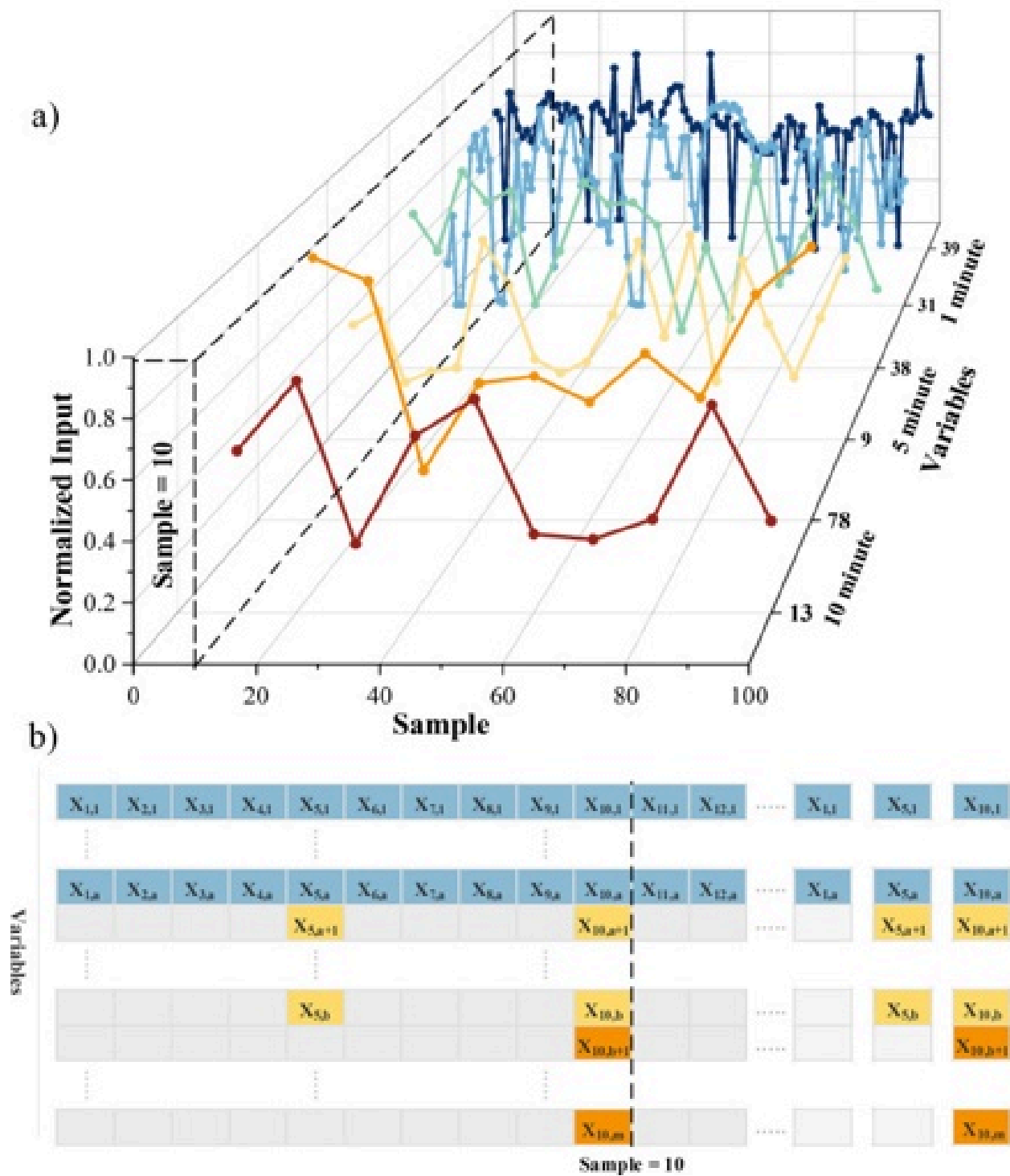


Figure 8. Illustration of multi-rate sampling: a) representative example of the collected data and b) configurations for combining inputs across different sampling intervals.

To promote both the precision and broad applicability of the deep learning architecture, the complete dataset is partitioned into two distinct portions: a training subset comprising 80% of the total observations and a validation subset accounting for the remaining 20%. Throughout the training phase, holdout validation is implemented with a validation check

performed every two training iterations. This mechanism continuously monitors model performance on unseen validation data throughout the learning process, thereby helping detect and mitigate potential overfitting.

Specific energy consumption

Metrics such as overall energy demand, thermal conversion efficiency, and energy intensity are commonly employed to characterize energy performance [34]. However, these indicators pose difficulties when comparing energy efficiency across varying production throughput levels, as they focus solely on total energy use and conversion efficiency without accounting for output scale [40]. An alternative and particularly useful metric is specific energy consumption (SEC) [41]. SEC quantifies the amount of energy consumed per unit of product generated and is computed as shown in Eq. 11.

$$SEC = \frac{\text{EnergyUsed}}{\text{AmountofProductProduced}} \quad (11)$$

By definition, the SEC standardizes energy requirements relative to the production output, enabling fair comparisons of energy performance regardless of differences in plant capacity. Moreover, the SEC serves as a versatile benchmarking tool applicable at various scales—from individual process units to regional and global assessments—and facilitates cross-industry evaluations for facilities manufacturing comparable products.

Results and Discussion

Feature importance ranking

To identify the most influential process variables, the full set of 97 measured parameters was evaluated and ranked by feature importance (FI) scores. Variables assigned an FI score of zero were immediately excluded, leaving 40 candidates for subsequent refinement. These remaining variables were then subjected to forward stepwise selection guided by the initial FI-based ranking. The outcomes of this iterative procedure during model training are presented in Figure 9. Analysis of the training performance reveals that the highest R^2 value and the lowest RMSE are achieved when exactly 14 predictor variables are included. The identities of these 14 most impactful variables, together with their corresponding FI scores, are summarized in Table 4. The forward stepwise results closely align with the original importance ranking, confirming that the selected subset comprises the variables with the strongest predictive relevance, as determined by the tree-based FI assessment.

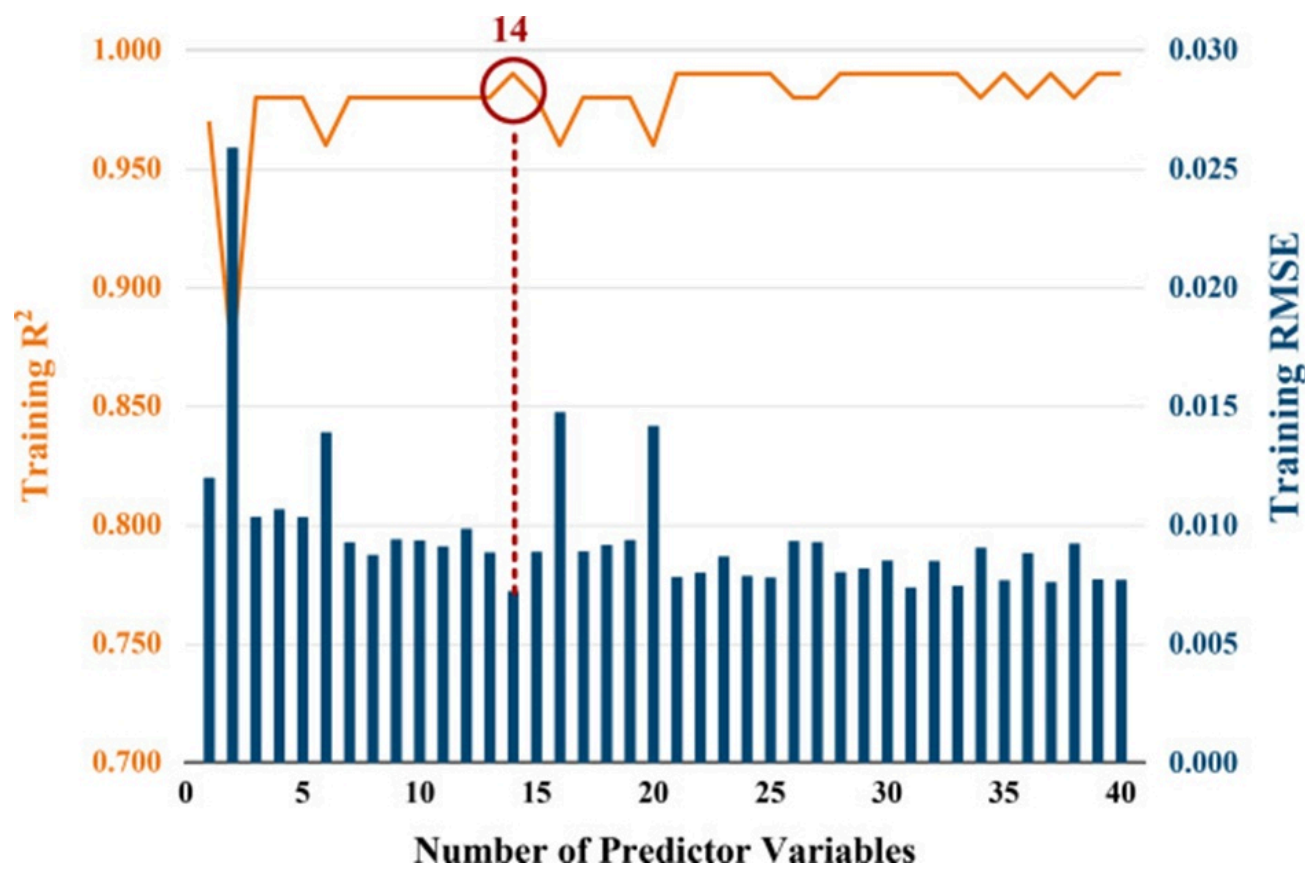


Figure 9. R2 and RMSE from model training of forward stepwise feature selection.

Table 4. Significant input variables for model development.

No.	Process variables	Interval	FI scores
63	Temperature of recycled EDC from D-404	1 min	15.15%
1	Temperature of EDC at the bottom of D-101	1 min	14.61%
84	Temperature at bottom T-501	1 min	10.15%
23	Mass flow rate at D-202 inlet	5 min	9.51%
9	Temperature of recycled HCl at L-228	5 min	8.31%
47	Mass flow rate at top T-302	1 min	8.29%
46	Mass flow rate at bottom T-301	1 min	7.17%
78	CW feed rate of E-403	10 min	7.03%
31	CW feed rate of E-204	1 min	6.39%
85	Temperature at top T-502	1 min	3.05%
10	Temperature of recycled HCl at L-229	1 min	2.53%
13	Temperature at bottom D-201	10 min	1.94%
39	Temperature at top T-301	1 min	1.78%

38	Temperature of EDC feed to T-301	5 min	1.04%
–	Other variables (total 83 variables)	–	3.05%

Hyperparameters and computational performance

Table 5 presents the optimal set of hyperparameters identified via grid search that yields the highest training R^2 across all convolutional neural network variants examined in this study. The findings indicate that both the TLCNN and 3C-CNN architectures require tuning of 14 hyperparameters. The primary distinction between these two models lies in the transfer mechanism: TLCNN inherits both the learned parameters (weights) and the hyperparameter configuration from the preceding multi-channel model, whereas 3C-CNN replicates only the hyperparameter settings without transferring the trained weights.

A comparison of the balance between training duration and predictive accuracy for the various CNN-based models is illustrated in **Figure 10**. All evaluated models demonstrate suitability for real-time industrial deployment, as their prediction times remain below one second—well within the timeframe of the most frequent sampling interval typically encountered in process measurements. The proposed TLCNN model requires approximately 28 seconds for complete training. Given that sampling intervals are on the order of minutes, this training duration ensures that model retraining can be completed before the next data point arrives.

In comparison, the 2C-CNN and 3C-CNN configurations demand longer training times than the standard single-channel CNN. Although the TLCNN has the highest training time among the tested approaches, it achieves the lowest training RMSE. Furthermore, despite sharing the same number of channels as the 3C-CNN, the TLCNN delivers faster inference speeds for new samples, highlighting an advantage in operational efficiency.

Table 5. Optimal hyperparameters for CNN models.

Tuning	Hyperparameters	CNN	2CCNN	3CCNN	TLCNN
1	Size of filters_1	10	10	10	10
	Number of filters_1	50	50	50	50
	Size of PR_1	3	3	3	3
	Number of FC nodes_1	50	50	50	50
2	Size of filters_2	–	10	10	10
	Number of filters_2	–	50	50	50
	Size of PR_2	–	3	3	3
	Number of FC nodes_2	–	1	1	1
3	Size of filters_3	–	–	15	15
	Number of filters_3	–	–	10	50
	Size of PR_3	–	–	5	5
	Number of FC nodes_3	–	–	1	40
Global	Learning rate	0.75	0.75	0.75	0.75

Note: PR = pooling regions, FC = fully connected.

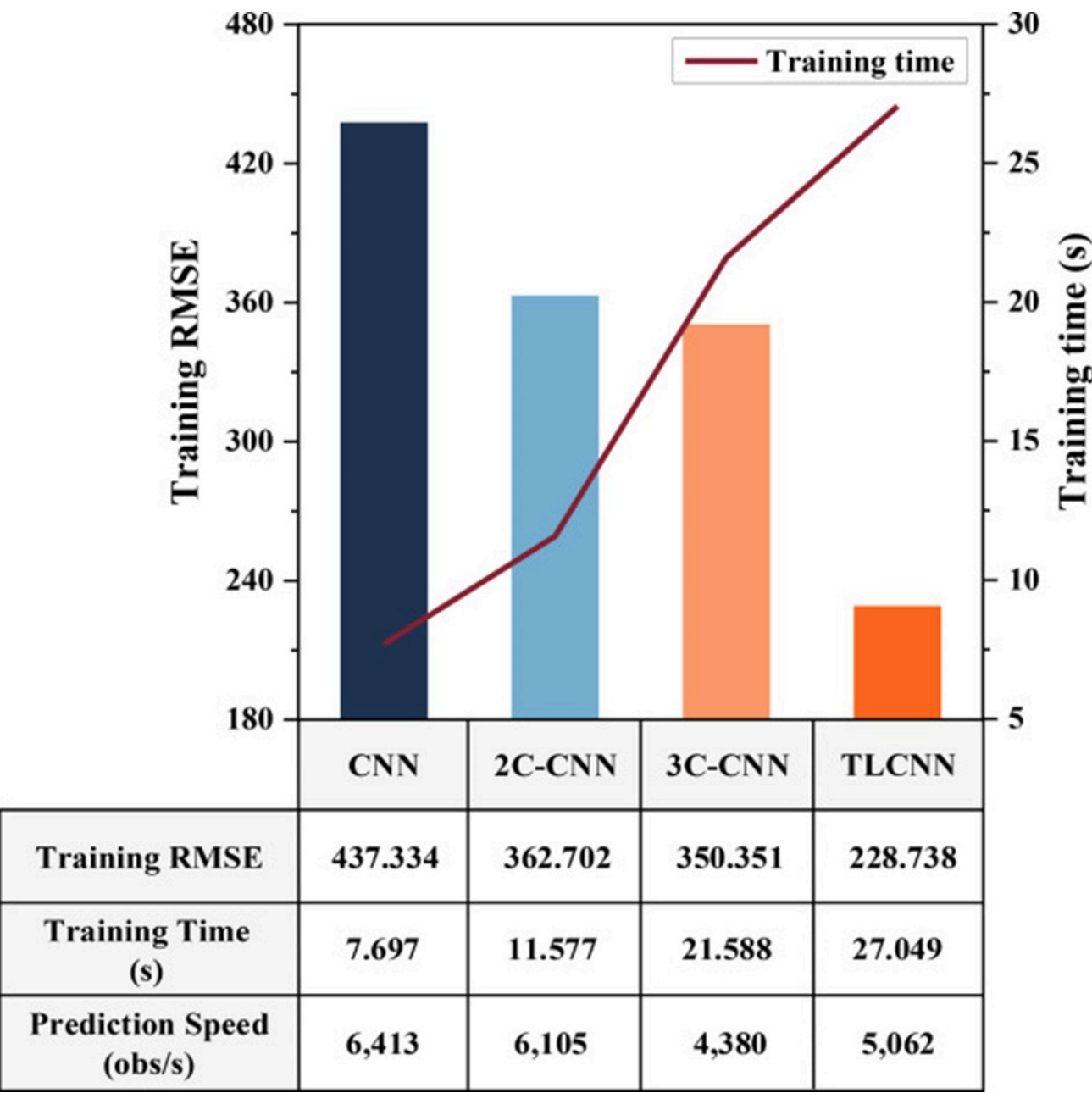


Figure 10. Comparison of Training RMSE and Time across Various CNN Models.

Performance of energy efficiency prediction models

This section presents the results of online testing for each model used for energy efficiency prediction. The models were deployed within the UniSim Design environment to evaluate their real-time performance. Each model underwent 20 iterations, with 400 samples per iteration, to assess the average performance, as summarized in [Table 6](#).

When comparing the performance of CNN, LSTM, RNN, and FFNN models, CNN demonstrated superior accuracy across multiple time scales. As a result, CNN was chosen as the preferred predictive model for the post-reconstruction data, employing zero-order hold and interpolation techniques. The baseline CNN model achieved an accuracy of 0.929. By incorporating zero-order hold and interpolation, the accuracies were slightly reduced to 0.907 and 0.915, respectively.

Among all the models tested, TLCNN (Transfer Learning CNN) provided the best performance, with an R^2 of 0.970, RMSE of 232.862, and MAPE of 0.784. The decrease in accuracy relative to the baseline CNN is attributed to the inherent assumptions of data reconstruction methods, such as zero-order hold and interpolation. The zero-order hold assumes constant data between samples, whereas interpolation presumes that missing values can be derived using a single relationship between two data points. These assumptions can lead to information loss and interpolation errors, thereby negatively affecting neural network prediction accuracy.

The proposed model improved the tracking capability of the SEC (Specific Energy Consumption) by 43.13%, surpassing the improvements from 2D-CNN (17.06%) and 3D-CNN (19.88%) compared to the simple CNN model. These findings suggest that increasing the channels in a model leads to diminishing returns in performance and greater susceptibility to overfitting. Transfer learning addresses this issue by freezing the weights and biases of the pre-trained channels, allowing the model to focus on learning from the untrained channels. This approach significantly enhances the model's performance and generalization capabilities.

Table 6. Software-in-loop testing results.

Dataset	Model	R^2	RMSE	MAPE
Training	FFNN	0.741	899.359	2.546
	RNN	0.891	433.708	1.392
	LSTM	0.904	498.773	1.511
	CNN	0.929	437.334	1.482
	CNN (Zero-orderhold)	0.851	483.642	1.667
	CNN (Interpolation)	0.871	452.174	1.551
	2CCNN	0.951	362.702	1.227
	3CCNN	0.956	350.351	1.183
	TLCNN	0.971	228.738	0.760
Validation	FFNN	0.832	848.018	1.846
	RNN	0.813	828.486	1.992
	LSTM	0.921	457.401	1.538
	CNN	0.927	436.898	1.474
	CNN (Zero-order hold)	0.868	487.621	1.678
	CNN (Interpolation)	0.872	475.678	1.631
	2CCNN	0.942	388.743	1.288
	3CCNN	0.954	355.058	1.185
	TLCNN	0.962	262.518	0.803
Testing	FFNN	0.771	869.828	2.315
	RNN	0.617	1205.701	3.501
	LSTM	0.919	462.329	1.650

CNN	0.929	392.484	1.262
CNN (Zero-order hold)	0.907	471.861	1.686
CNN (Interpolation)	0.915	443.578	1.566
2CCNN	0.945	319.277	1.094
3CCNN	0.948	311.666	1.039
TLCNN	0.970	232.862	0.784

Figure 11 compares the true SEC values with those estimated by various CNN-based models in the context of the VCM process. The findings indicate that TLCNN performs reliably across a broad range of SEC values. In contrast, the standard CNN, 2C-CNN, and 3C-CNN models produce accurate predictions only when the true SEC lies roughly within the 22,000–24,000 kJ/kg range. By employing transfer learning, the approach effectively balances bias and variance—a challenge that conventional models frequently struggle to overcome in real-time applications. In this architecture, the pre-trained channel functions primarily as a feature extractor, while the subsequent channel is dedicated to task-specific fine-tuning, thereby significantly expanding the model's effective operating range.

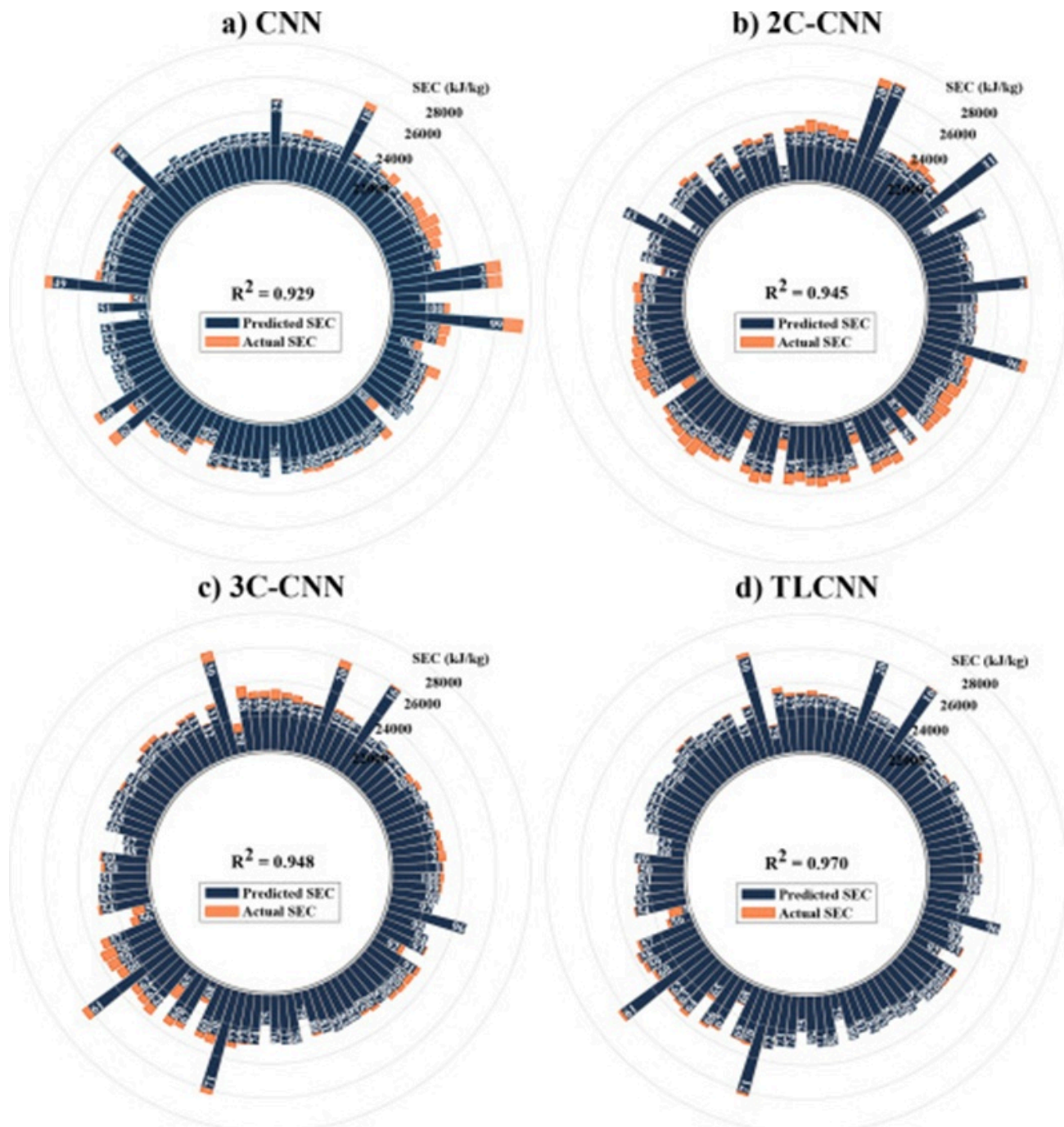


Figure 11. An example comparing the actual specific energy consumption (SEC) values against those forecasted by the various CNN-based models.

Reproducibility analysis

Beyond mere predictive accuracy, the consistency and repeatability of artificial intelligence models are critical factors in their application to innovative forecasting tasks. This section outlines the methodology and outcomes of a dedicated reproducibility evaluation to confirm that the obtained results can be reliably replicated. Each model was retrained 200 times to evaluate its variability. Fresh datasets were created via random sampling and used for training with identical optimal hyperparameters across varying random seeds.

Figure 12 displays the outcomes from the testing phase of this reproducibility procedure. In contrast to models trained only once, adaptive architectures like TLCNN exhibit superior consistency across repetitions. The analysis reveals that the test R^2 for TLCNN remains within 0.929 to 0.992, whereas the plain CNN model ranges from 0.874 to 0.933 (after outlier removal). Notably, incorporating multiple channels into 2CCNN and 3CCNN increases the peak achievable test R^2 ; however, their lowest performance levels decline markedly to 0.854 and 0.869, respectively. This heightens the risk of unsuccessful training runs during model development.

Increasing the channel count in the network architecture tends to complicate the optimization process and convergence to a robust solution. Here, parameter-based transfer learning proves particularly valuable in mitigating challenges associated with weight initialization. By providing a pre-trained model as a stable initial configuration, it guides optimization toward appropriate regions of the parameter space, constrains the number of trainable parameters, and thereby enhances overall training stability and reliability.

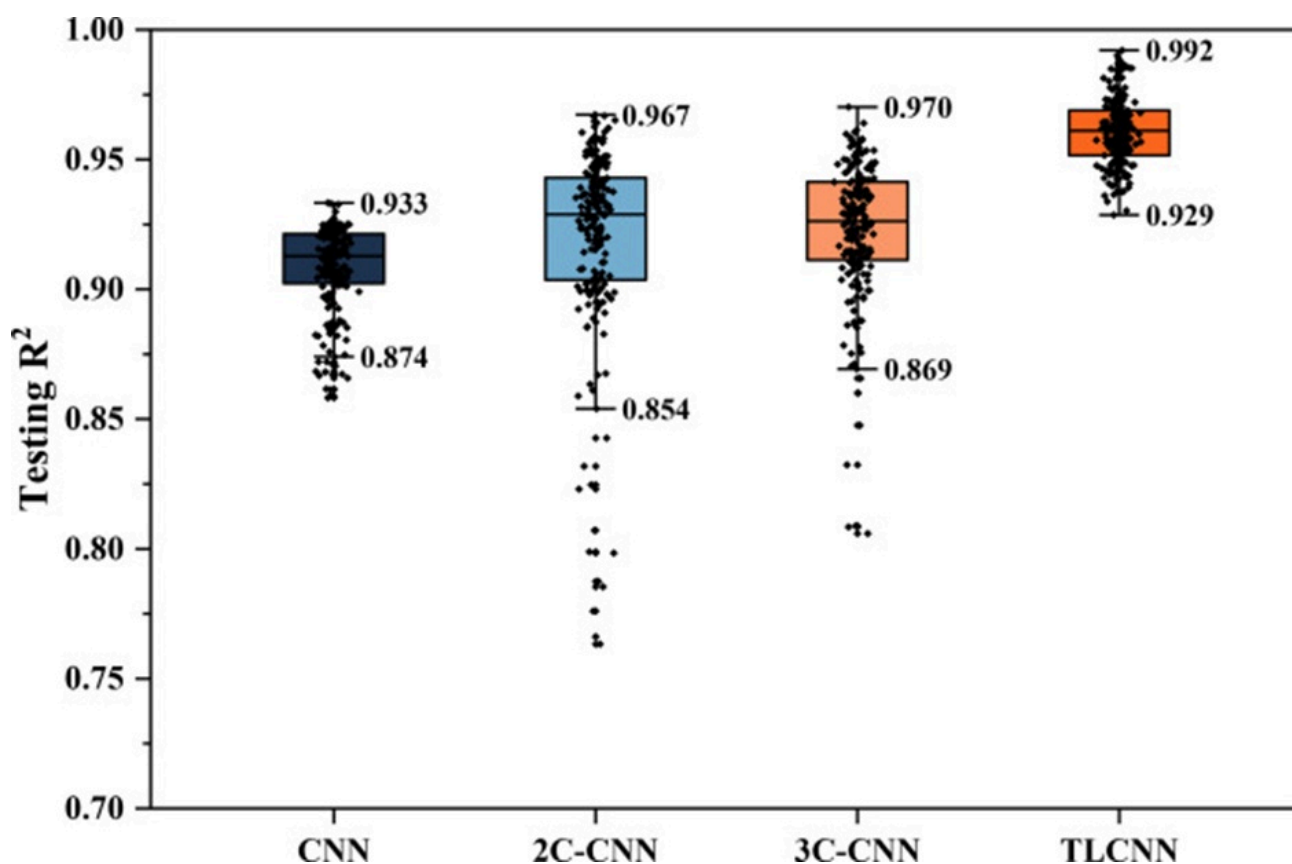


Figure 12. Boxplot displaying the R^2 values obtained during testing of the CNN models.

Sliding window evaluation (or shifting test)

In real-world applications, after completing model training, the system is put into operation to gather live performance metrics using incoming fresh data. The conventional evaluation approach on the primary test set involves making predictions using input sequences that match the structure used in training (such as durations of 1, 5, or 10 minutes). However, in multi-rate forecasting scenarios, it is essential to examine how well the model handles changes in temporal resolution or sampling intervals, as this reveals its robustness to data patterns that deviate from those seen during training.

Figure 13 depicts various scenarios in which the time alignment or sampling period shifts in industrial sensor data collection. Scenarios Testing 1 and 2 demonstrate the influence of system latency or timing offsets, resulting in shifted

sampling windows—either aligned according to the least common multiple (for example, 1–6–12 minutes) or misaligned without such synchronization (for example, 1–6–11 minutes). Scenarios Testing 3 and 4 highlight cases where the sampling frequency of specific variables increases (becoming more frequent) or decreases (potentially interrupted or halted during periods of equipment servicing or maintenance).

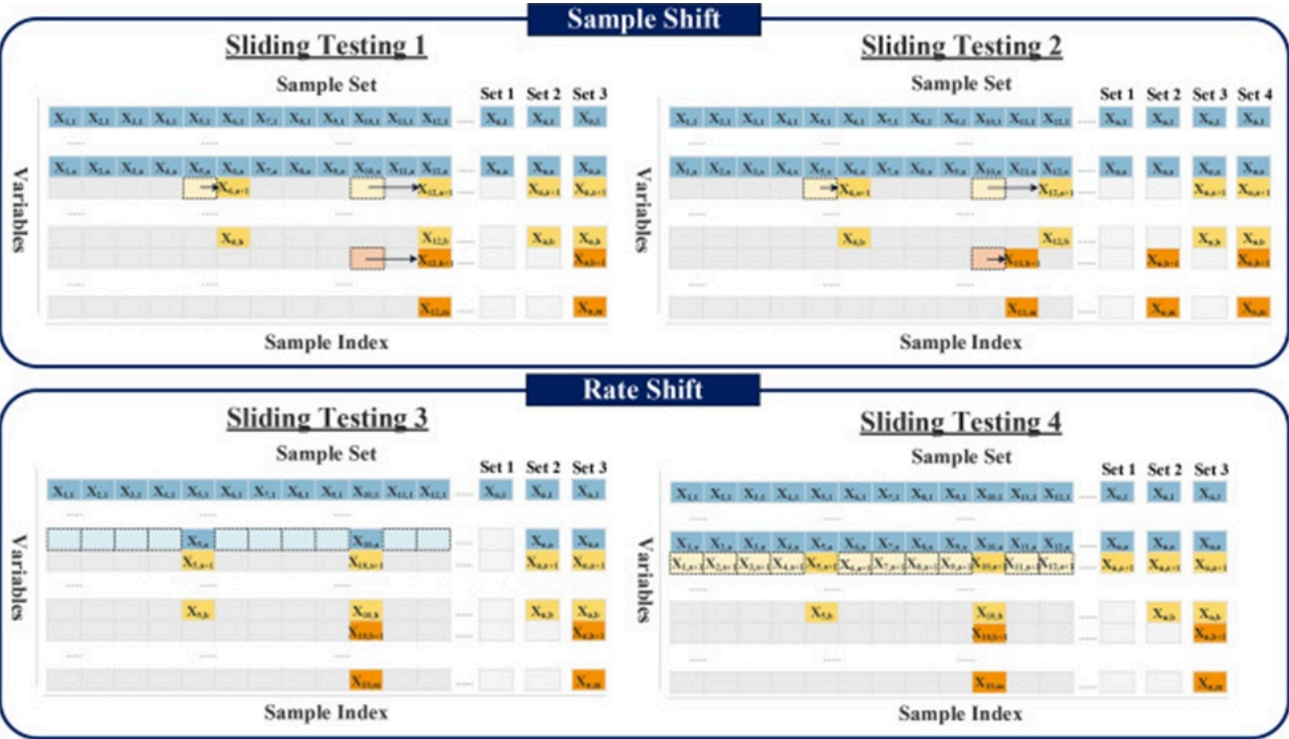


Figure 13. Depiction of the sliding evaluation scenarios.

Table 7 summarizes the performance metrics of multiple models across modified temporal scales, assessed using a sliding-window test with 400 samples. The outcomes demonstrate that all models maintain robust accuracy only in Testing 1. In this specific scenario, the three sampling configurations present in the test data (1 min, 1–6 min, 1–6–12 min) precisely correspond to those included in the training data (1 min, 1–5 min, 1–5–10 min). Conversely, as soon as an unseen sampling combination emerges during the testing phase—one that did not exist in the training phase—the forecasting accuracy of every model declines substantially.

The application of transfer learning emerges as a key factor in improving resilience under these conditions. By preserving the trained weights of earlier layers (freezing them) and fine-tuning only the last processing channel, the model gains markedly better capacity to handle novel situations. The frozen pre-trained sections preserve the valuable feature representations learned from the original time patterns. At the same time, the adaptable final channel specializes in capturing the unique properties and variations introduced by the altered sampling rates.

Table 7. Result of the CNN-based model on a sliding test.

Scenario	Dataset	Model	Testing R ²	RMSE	MAPE
Sample Shift	Test 1	CNN	0.935	334.688	1.152
		2CCNN	0.931	344.871	1.187
		3CCNN	0.933	340.815	1.181

Variable Shift	Test 2	TLCNN	0.954	271.384	0.919
		CNN	0.752	820.529	1.652
		2CCNN	0.662	918.777	2.091
		3CCNN	0.779	653.652	1.779
		TLCNN	0.938	337.797	1.071
	Test 3	CNN	0.392	2137.721	4.163
		2CCNN	0.398	2077.245	4.048
		3CCNN	0.464	1767.844	3.755
		TLCNN	0.678	985.981	2.671
	Test 4	CNN	0.471	2678.653	9.565
		2CCNN	0.462	2598.033	9.157
		3CCNN	0.505	2257.201	8.161
		TLCNN	0.815	595.259	2.024

Figure 14 illustrates how each filter in the TLCNN model assigns weight to the input and highlights the importance of network connections in the final SEC prediction, using the concept of feature attribution from weights [42]. The results demonstrate that the three channels of the CNN model learn distinct patterns from the input data. The first channel primarily focuses on inputs with 1- and 10-minute sampling intervals, contributing 53.44% of the total input weight and explaining approximately 70.32% of the SEC prediction. The second channel captures patterns from inputs with 5- and 10-minute sampling intervals, making up 65.58% of the total input weight. Lastly, the third channel in the TLCNN mainly processes inputs with a 1-minute sampling interval. Despite this channel handling the highest-frequency input, the analysis shows that it accounts for only 28.41% of the absolute SEC prediction values.

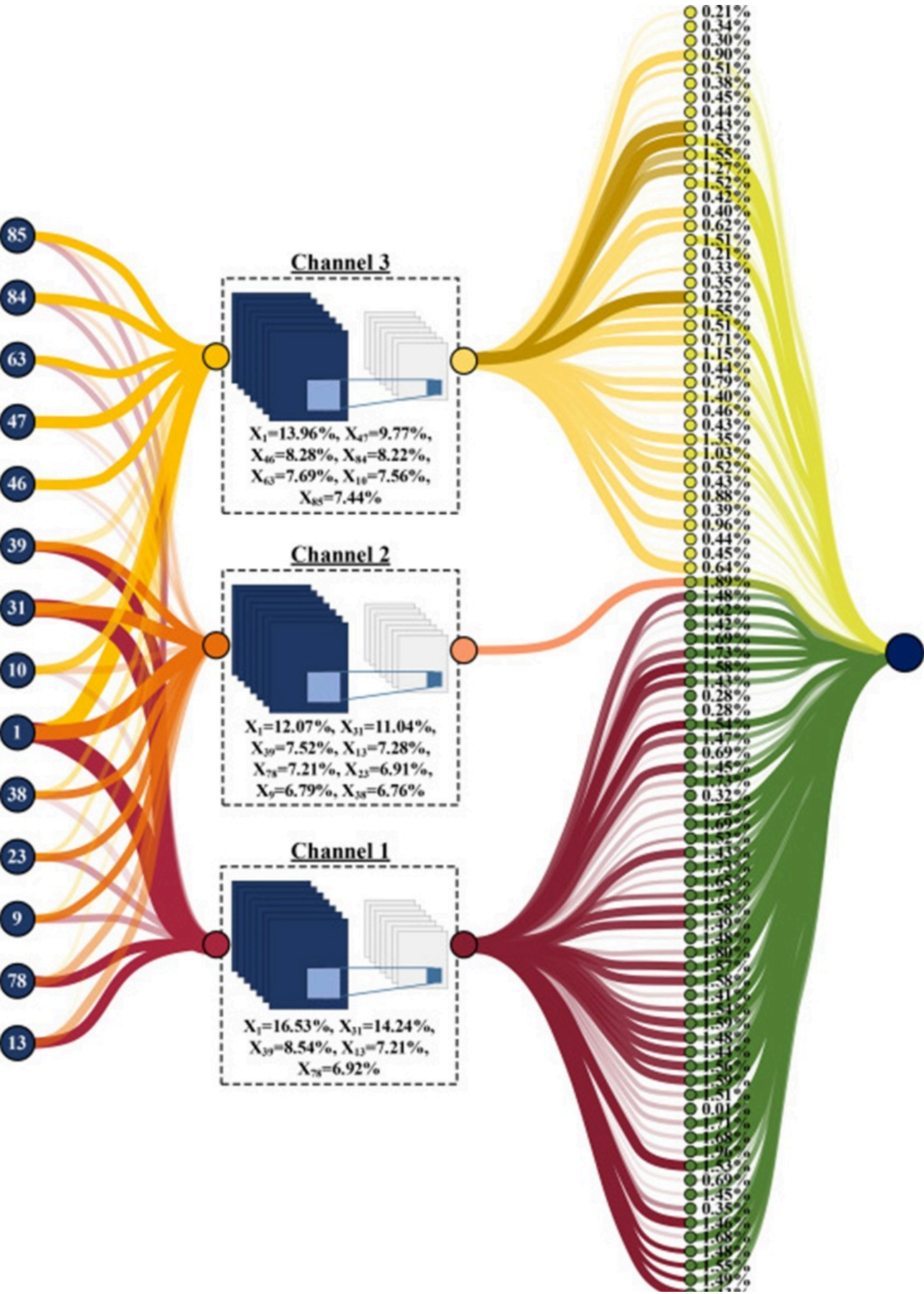


Figure 14. Results of the network analysis performed on the TLCCNN architecture.

Analysis of energy savings

A model that lacks sufficient tracking capability and produces inaccurate predictions is not appropriate for reliably identifying patterns in specific energy consumption (SEC). The residual error is the difference between the observed (actual) SEC and the predicted SEC, with the predicted SEC serving as the reference point. Consequently, a large positive residual indicates periods where energy usage is elevated relative to the expected level for the given production output—in other words, high consumption paired with relatively low production efficiency.

When applying the proposed model, a clear and substantial discrepancy (an energy gap) emerges between the measured SEC values and the corresponding predictions. As illustrated in **Figure 15**, some data points exhibit residuals that exceed the 95th percentile threshold relative to their neighbors. These outlier points can be grouped into two distinct clusters based on the ethylene plant's operating load: one cluster corresponds to a production rate of approximately 55,000 kg/h (Cluster 1). At the same time, the other aligns with roughly 52,300 kg/h (Cluster 2).

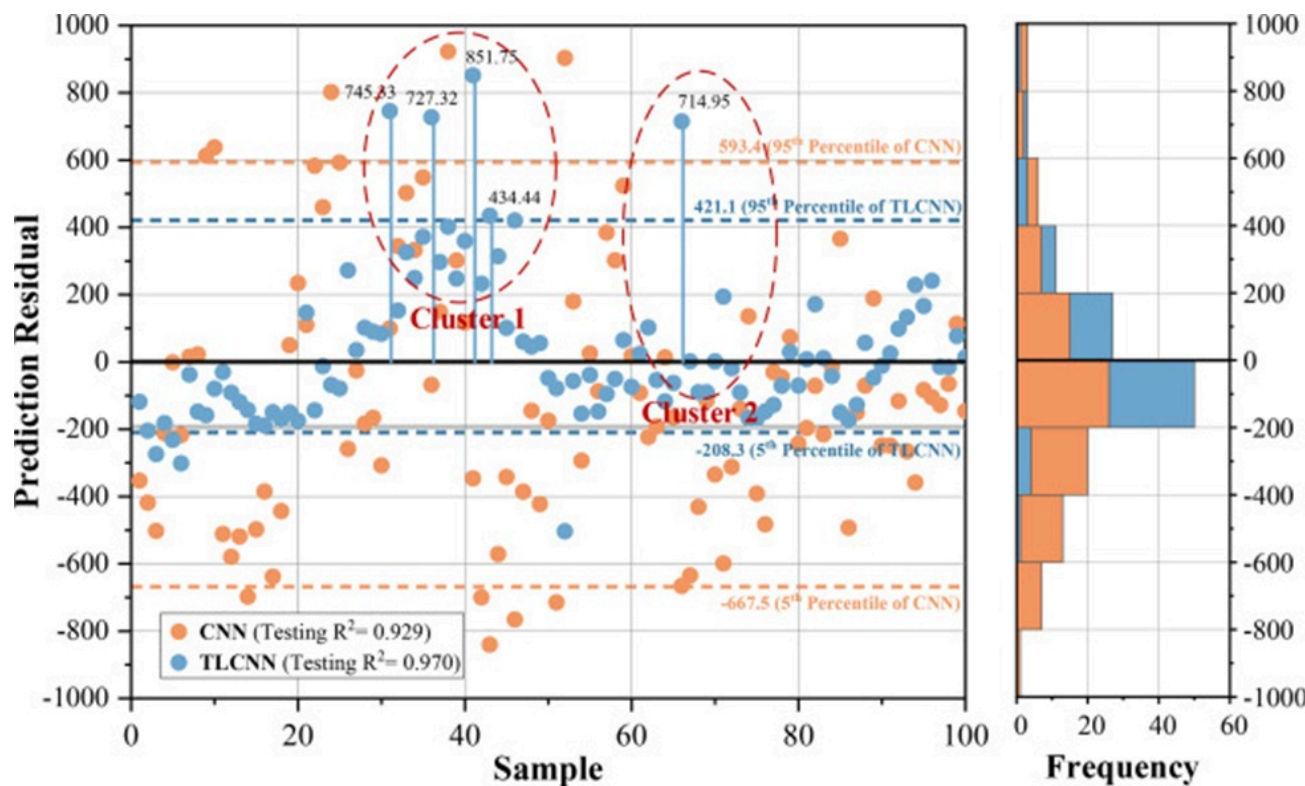


Figure 15. Plot of residuals between actual specific energy consumption (SEC) and the benchmark SEC values generated by the models.

Figure 16a delineates the range of energy-efficient operating conditions for Cluster 1, aligned with the key influential inputs listed in **Table 4**. For the outlier in Cluster 1 (specifically the 31st sample), the observed SEC reaches 21,413 kJ/kg, whereas the corresponding benchmark under efficient conditions is approximately 20,668 kJ/kg. When this 31st sample's operating parameters are compared against nearby data points that fall below the 95th percentile residual threshold and share the same ethylene production load, it becomes evident that this point exhibits notably higher energy use relative to its immediate neighbors within the feasible operating region (represented by the shaded area). Within this solution space, SEC values typically span from 20,184 to 20,764 kJ/kg. By appropriately adjusting or optimizing the control variables to move toward this efficient zone, it is possible to reduce the SEC by an estimated 649–1,229 kJ/kg.

A parallel analysis applied to Cluster 2 (represented by the 43rd sample) in **Figure 16b** yields similar insights. Here, the SEC for the 43rd sample is approximately 24,063 MJ/ton, while the identified energy-efficient solution space ranges from 22,611 to 23,654 kJ/kg. This indicates a potential SEC reduction from 409 to 1,451 kJ/kg, translating to annual utility savings of roughly 3.5% to 11%.

By integrating this analytical approach into production scheduling and real-time process optimization, the proposed framework uncovers a substantial yearly cost-saving potential of approximately 299,614 USD. This figure is derived from energy price estimates based on the natural gas spot price reported by the U.S. Energy Information Administration [43]. In environmental terms, realizing these energy reductions would avoid the release of about 5,709.59 tons of carbon dioxide equivalent (CO₂e) annually. This emission avoidance calculation relies on standard greenhouse gas reporting factors published by the U.S. Environmental Protection Agency [44], with the following emission factors applied per TJ of energy: CO₂ (1 GWP) at 50,291.18 kg/TJ, CH₄ (25 GWP) at 947.82 kg/TJ, and N₂O (298 GWP) at 94.78 kg/TJ.

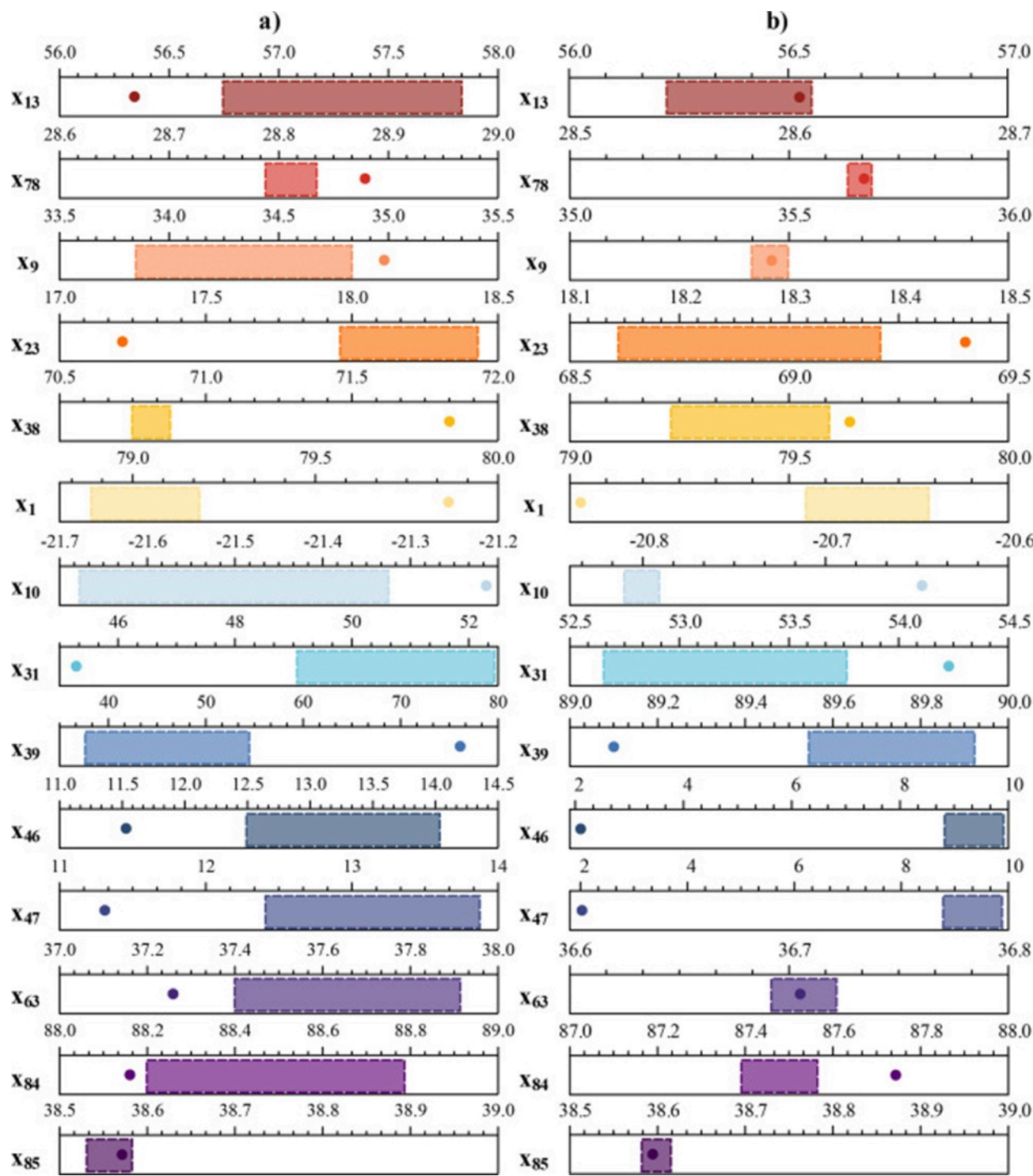


Figure 16. Benchmarking of energy-efficient operating ranges: (a) for Cluster 1 and (b) for Cluster 2.

Conclusion

In this study, TLCNN models were developed to estimate energy efficiency and support energy-saving assessments for the vinyl chloride monomer (VCM) production process, particularly under conditions involving multi-rate sampling intervals in the process measurements. The multi-channel design enhanced the effectiveness of the energy analysis by enabling

each channel to specialize in learning distinct patterns in the input data, thereby expanding the network's overall learning capability. The incorporation of transfer learning markedly improved the model's ability to adapt to previously unseen datasets, effectively addressed challenges arising from shifts in temporal alignment after full training, and mitigated issues related to random weight initialization—factors that collectively led to substantially greater reproducibility.

The performance of the proposed model was rigorously validated through real-time evaluations, reproducibility assessments, and sliding-window (time-shifted) testing scenarios. Subsequently, an energy-saving investigation was conducted by leveraging the model's predictions in combination with exploratory data analysis. The main findings can be summarized as follows:

1. The TLCNN architecture successfully manages multi-rate data streams and provides accurate real-time tracking of specific energy consumption (SEC) in the VCM process. Although its training time is approximately 1.25 times longer than that of the 3CCNN baseline, the TLCNN delivers predictions 1.15 times faster and achieves a 19% improvement in overall accuracy. Compared with conventional learning approaches, the proposed model exhibits superior performance across key metrics: $R^2 = 0.97$, $RMSE = 232.862$, and $MAPE = 0.784\%$.
2. The training process for TLCNN demonstrates notably higher reproducibility and greater robustness against weight initialization variability. Through a parameter-based transfer learning strategy that transfers all weights and biases from a pre-trained model, TLCNN consistently attains test-set accuracies between 0.929 and 0.992 across multiple runs, with no instances of training divergence or failure.
3. Once trained, the model maintains reliable predictive capability when the combinations of input variables (sampling patterns) in new data remain identical to those encountered during training. When novel combinations arise, TLCNN maintains strong performance by freezing the pre-trained core network and fine-tuning only the additional final channel to capture the characteristics of the new temporal patterns.
4. Analysis of the energy gaps between actual SEC measurements and the efficient benchmark values predicted by TLCNN identifies practical opportunities to reduce energy utilization by 3.5% to 11%. This corresponds to an estimated annual cost saving of approximately 300,000 USD in production management expenses, alongside an environmental benefit of preventing roughly 5,709 tons of CO₂ equivalent emissions per year.

In summary, the proposed TLCNN model offers excellent predictive accuracy, strong reproducibility, and greater interpretability than standard machine learning methods. It performs robustly even when input sampling intervals are inconsistent, although its effectiveness diminishes when entirely new combinations of predictor variables appear during deployment. Looking ahead, integrating self-adaptive learning mechanisms into the architecture could further enhance long-term performance. Such an adaptive network would automatically detect declines in prediction quality during online operation and recalibrate relevant parameters accordingly, ensuring sustained reliability as the distribution of input variable combinations evolves.

Acknowledgements

The author would like to acknowledge the support of the Faculty of Engineering, Kasetsart University (Grant No.66/04/CHEM/D.Eng), the Center for Advanced Studies in Industrial Technology, and the Center of Excellence on Petrochemical and Materials Technology. We would also like to acknowledge Honeywell and GRD Tech Co., Ltd. for providing the UniSim Design Suite simulation software used in the study.

Conflict of interest

None

Financial support

None

Ethics statement

None

Received: 07 Oct 2024

Revised: 13 Nov 2024

Accepted: 22 Dec 2024

Published online: 20 March 2025

Rights and permissions

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

References

- Ritchie H, Roser M, Rosado P. Energy. Our World in Data; 2022. <https://ourworldindata.org/energy> [accessed July 8, 2022].
- Patterson M, Singh P, Cho H. The current state of the industrial energy assessment and its impacts on the manufacturing industry. *Energy Rep.* 2022;8:7297-311.
- Energy Information Administration. Manufacturing Energy Consumption Survey (MECS) 2021. <https://www.eia.gov/consumption/manufacturing/data/archive/> [accessed July 8, 2022].
- Menghi R, Papetti A, Germani M, Marconi M. Energy efficiency of manufacturing systems: a review of energy assessment methods and tools. *J Clean Prod.* 2019;240:118276.
- Golmohamadi H. Demand-side management in industrial sector: a review of heavy industries. *Renew Sustain Energy Rev.* 2022;156:111963.
- Gao H, Li Z, Qiu S, Yang B, Li S, Wen Y. Energy exchange efficiency prediction from non-linear regression for membrane-based energy-recovery ventilator cores. *Appl Therm Eng.* 2021;197:117353.
- Zhu L, Chen J. Development of energy efficiency principal component analysis model for factor extraction and efficiency evaluation in large-scale chemical processes. *Int J Energy Res.* 2019;43:814-28.
- Cattani G. Combining data envelopment analysis and Random Forest for selecting optimal locations of solar PV plants. *Energy AI.* 2023;11:100222.
- Yang J, Chen B. Energy efficiency evaluation of wastewater treatment plants (WWTPs) based on data envelopment analysis. *Appl Energy.* 2021;289:116680.
- Vieira LW, Marques AD, Schneider PS, José da Silva Neto A, Viana FAC, Abdeljawad M, et al. Methodology for ranking controllable parameters to enhance operation of a steam generator with a combined Artificial Neural Network and Design of Experiments approach. *Energy AI.* 2021;3:100040.

Walser T, Sauer A. Typical load profile-supported convolutional neural network for short-term load forecasting in the industrial sector. *Energy AI*. 2021;5:100104.

Machalek D, Tuttle J, Andersson K, Powell KM. Dynamic energy system modeling using hybrid physics-based and machine learning encoder–decoder models. *Energy AI*. 2022;9:100172.

Liu J, Chen L, Xu W, Feng M, Han Y, Xia T, et al. Novel production prediction model of gasoline production processes for energy saving and economic increasing based on AM-GRU integrating the UMAP algorithm. *Energy*. 2023;262:125536.

Han Y, Cao L, Geng Z, Ping W, Zuo X, Fan J, et al. Novel economy and carbon emissions prediction model of different countries or regions in the world for energy optimization using improved residual neural network. *Sci Total Environ*. 2023;860:160410.

Han Y, Du Z, Geng Z, Fan J, Wang Y. Novel long short-term memory neural network considering virtual data generation for production prediction and energy structure optimization of ethylene production processes. *Chem Eng Sci*. 2023;267:118372.

Wang S, Fan Y, Jin S, Takyi-Aninakwa P, Fernandez C. Improved anti-noise adaptive long short-term memory neural network modeling for the robust remaining useful life prediction of lithium-ion batteries. *Reliab Eng Syst Saf*. 2023;230:108920.

Wang S, Takyi-Aninakwa P, Jin S, Yu C, Fernandez C, Stroe D-I. An improved feedforward-long short-term memory modeling method for the whole-life-cycle state of charge prediction of lithium-ion batteries considering current-voltage-temperature variation. *Energy*. 2022;254:124224.

Li J, Lin M, Li Y, Wang X. Transfer learning network for nuclear power plant fault diagnosis with unlabeled data under varying operating conditions. *Energy*. 2022;254:124358.

Liu J, Hou G-Y, Shao W, Chen J. A supervised functional Bayesian inference model with transfer-learning for performance enhancement of monitoring target batches with limited data. *Process Saf Environ Prot*. 2023;170:670-84.

Mondal S, Chattopadhyay A, Mukhopadhyay A, Ray A. Transfer learning of deep neural networks for predicting thermoacoustic instabilities in combustion systems. *Energy AI*. 2021;5:100085.

Peirelinck T, Kazmi H, Mbuwir BV, Hermans C, Spiessens F, Suykens J, et al. Transfer learning in demand response: a review of algorithms for data-efficient modelling and control. *Energy AI*. 2022;7:100126.

Panjanornpon C, Bardeeniz S, Hussain MA, Chomchai P. Explainable deep transfer learning for energy efficiency prediction based on uncertainty detection and identification. *Energy AI*. 2023;12:100224.

Wang S, Ren P, Takyi-Aninakwa P, Jin S, Fernandez C. A critical review of improved deep convolutional neural network for multi-timescale state prediction of lithium-ion batteries. *Energies*. 2022;15:5053.

Heo J, Song K, Han S, Lee D-E. Multi-channel convolutional neural network for integration of meteorological and geographical features in solar power forecasting. *Appl Energy*. 2021;295:117083.

Chiney A, Paduri AR, Darapaneni N, Kulkarni S, Kadam M, Kohli I, et al. Handwritten data digitization using an anchor based multi-channel CNN (MCCNN) trained on a hybrid dataset (h-EH). *Procedia Comput Sci*. 2021;189:175-82.

Quest H, Cauz M, Heymann F, Rod C, Perret L, Ballif C, et al. A 3D indicator for guiding AI applications in the energy sector. *Energy AI*. 2022;9:100167.

Panjanornpon C, Bardeeniz S, Hussain MA. Improving energy efficiency prediction under aberrant measurement using deep compensation networks: a case study of petrochemical process. *Energy*. 2022;125837.

Huang K, Wu S, Li Y, Yang C, Gui W. A multi-rate sampling data fusion method for fault diagnosis and its industrial applications. *J Process Control*. 2021;104:54-61.

Panjabornpon C, Bardeeniz S, Hussain MA. Deep learning approach for energy efficiency prediction with signal monitoring reliability for a vinyl chloride monomer process. *Reliab Eng Syst Saf*. 2023;231:109008.

Thebelt A, Wiebe J, Kronqvist J, Tsay C, Misener R. Maximizing information from chemical engineering data sets: applications to machine learning. *Chem Eng Sci*. 2022;252:117469.

Liu H, Han L, Cao Y. Improving transmission efficiency and reducing energy consumption with automotive continuously variable transmission: a model prediction comprehensive optimization approach. *Appl Energy*. 2020;274:115303.

Tien PW, Wei S, Darkwa J, Wood C, Calautit JK. Machine learning and deep learning methods for enhancing building energy efficiency and indoor environmental quality – a review. *Energy AI*. 2022;10:100198.

Moghadasi M, Ozgoli HA, Farhani F. Steam consumption prediction of a gas sweetening process with methyldiethanolamine solvent using machine learning approaches. *Int J Energy Res*. 2021;45:879-93.

Han Y, Wu H, Geng Z, Zhu Q, Gu X, Yu B. Review: energy efficiency evaluation of complex petrochemical industries. *Energy*. 2020;203:117893.

Müller IM. Feature selection for energy system modeling: identification of relevant time series information. *Energy AI*. 2021;4:100057.

Li L, Wang J. Research on feature importance evaluation of wireless signal recognition based on decision tree algorithm in cognitive computing. *Cogn Syst Res*. 2018;52:882-90.

Himeur Y, Elnour M, Fadli F, Meskin N, Petri I, Rezgui Y, et al. Next-generation energy systems for sustainable smart cities: roles of transfer learning. *Sustain Cities Soc*. 2022;85:104059.

Pinto G, Wang Z, Roy A, Hong T, Capozzoli A. Transfer learning for smart buildings: a critical review of algorithms, applications, and future perspectives. *Adv Appl Energy*. 2022;5:100084.

Chinprasit J, Panjabornpon C. Model predictive control of vinyl chloride monomer process by Aspen Plus Dynamics and MATLAB/Simulink co-simulation approach. *IOP Conf Ser Mater Sci Eng*. 2020;778:012080.

Zhang X, Hou L, Liu J, Yang K, Chai C, Li Y, et al. Energy consumption prediction for crude oil pipelines based on integrating mechanism analysis and data mining. *Energy*. 2022;254:124382.

Lawrence A, Thollander P, Andrei M, Karlsson M. Specific Energy Consumption/Use (SEC) in energy management for improving energy efficiency in industry: meaning, usage and differences. *Energies*. 2019;12:247.

Machlev R, Heistrene L, Perl M, Levy KY, Belikov J, Mannor S, et al. Explainable Artificial Intelligence (XAI) techniques for energy and power systems: review, challenges and opportunities. *Energy AI*. 2022;9:100169.

Energy Information Administration. Henry Hub natural gas spot price (Dollars per Million Btu) 2022. <https://www.eia.gov/dnav/ng/hist/rngwhhdm.htm> [accessed July 8, 2022].

United States Environmental Protection Agency. GHG emission factors hub 2015. <https://www.epa.gov/climateleadership/ghg-emission-factors-hub> [accessed July 8, 2022].