Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Multi-Branch Deep Feature Fusion Framework Utilizing TCN and ANN for Short-Term Building Load Forecasting Under Pre-Scheduled Constraints

SITI NUR AFIFAH MOHD SUHAIMI
nrafifahsuhaimi@gmail.com
Tun Hussein Onn University of Malaysia
SY YI SIM
Tun Hussein Onn University of Malaysia
NOR AIRA ZAMBRI
Tun Hussein Onn University of Malaysia
ALVIN JOHN MENG SIANG LIM
Tun Hussein Onn University of Malaysia
HUSAM S. SAMKARI
University of Tabuk
Muhamad Azfar Azri Azhari
MSR Green Energy Sdn Bhd

Article

Keywords: Building Load Forecasting, Temporal Convolutional Network (TCN), Artificial Neural Network (ANN), Deep Feature Fusion, Short-Term Load Forecasting (STLF)

Posted Date: November 19th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11032768/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Multi-Branch Deep Feature Fusion Framework Utilizing TCN and ANN for Short-Term Building Load Forecasting Under Pre-Scheduled Constraints

Siti Nur Afifah Binti Mohd Suhaimi[a,1], Sim Sy Yi[b],Nor Aira Binti Zambri[a],Alvin John Lim Meng Siang[c],Husam S. Samkari[d] ,Muhamad Azfar Azri Azahari[e]

[a]*Department of Electrical Engineering Technology, Faculty of Engineering Technology, Universiti Tun Hussien Onn Malaysia, UTHM Kampus Pagoh, Hab Pendidikan Tinggi Pagoh, KM 1, Jalan Panchor, Pagoh, 84600, Johor, Malaysia*
[b]*Faculty of Electrical and Electronic Engineering (FKEE) Universiti Tun Hussein Onn Malaysia (UTHM) 86400 Parit Raja, Batu Pahat Johor, Malaysia*
[c]*Fakulti Kejuruteraan Awam dan Alam Bina (FKAAB) Universiti Tun Hussein Onn Malaysia (UTHM) 86400 Parit Raja, Batu Pahat Johor, Malaysia*
[d]*Department of Electrical Engineering,University of Tabuk,Tabuk, 47713 Saudi Arabia*
[e]*MSR Green Energy Sdn Bhd, Kuala Lumpur, Malaysia*

Abstract: Accurate short-term load forecasting (STLF) is essential for the efficient operation of Building Energy Management Systems (BEMS) and localized microgrid scheduling. However, building load behavior is highly volatile due to the nonlinear interactions among scheduled load baselines, temporal patterns, and dynamic weather conditions. Traditional forecasting models often rely on a single-stream feature representation, which may distort training signals and increase the risk of overfitting, particularly when only limited historical data are available. This paper proposes a multi-branch deep feature fusion TCN-ANN hybrid framework for hourly building energy forecasting using a single year of historical operational data (8,760 samples). The proposed architecture processes diverse input features through separate learning pathways. A deep one-dimensional causal dilated Temporal Convolutional Network (TCN) branch extracts sequential representations from temperature, relative humidity, and 24-hour load lag trajectories, while a parallel multi-layer Artificial Neural Network (ANN) branch learns tabular relationships from scheduled load profiles and temporal variables. The resulting feature representations are fused through channel-wise feature combination and subsequently processed by a shared dense fusion layer. The effectiveness of the proposed framework was evaluated using a real-world building dataset with a strict chronological 70% training, 15% validation, 15% testing split and uniform Min–Max normalization. The model achieved an RMSE of 20.03 kW and a coefficient of determination (R^2) of 0.9721, demonstrating excellent forecasting performance. Furthermore, the proposed framework outperformed the standalone TCN benchmark RMSE by 25.75 kW, indicating that deep feature fusion effectively captures complex interactions among temporal, environmental, and operational variables while reducing the risk of overfitting in small-scale time-series forecasting applications.

Keywords: Building Load Forecasting; Temporal Convolutional Network (TCN); Artificial Neural Network (ANN); Deep Feature Fusion; Short-Term Load Forecasting (STLF)

1. Introduction

According to the International Energy Agency (2023), modern commercial, institutional, and industrial buildings are among the largest contributors to global electricity consumption, accounting for nearly 30–40% of total energy demand in many urban regions [1]. The increasing penetration of smart grids and decentralized energy systems has intensified the need for intelligent forecasting mechanisms to support real-time decision-making in Building Energy Management Systems (BEMS) [2]. Short-term load forecasting (STLF) plays a critical role in enabling demand-side optimization, peak shaving, energy cost reduction, and microgrid stability. Accurate forecasting allows energy operators to proactively schedule distributed energy resources, optimize HVAC operations, and reduce dependency on expensive peak-hour electricity tariffs. HVAC and lighting loads are particularly sensitive to occupant behaviour, weather conditions, and time-of-use patterns [3]. For instance, HVAC energy consumption peaks during hot afternoons when multiple occupants are present, while lighting load exhibits significant evening peaks influenced by household routines [4].

In modern microgrid-enabled buildings, energy consumption is often pre-defined through contractual scheduling or operational planning, commonly referred to as scheduled load baselines. However, these baselines rarely reflect actual consumption due to dynamic environmental disturbances. External factors such as ambient temperature, relative humidity, and solar irradiance significantly affect HVAC loads, which represent a major portion of building energy consumption [5]. When weather conditions deviate from expected patterns, a significant mismatch occurs between scheduled and actual demand. This mismatch introduces operational risks such as voltage instability, energy imbalance, and increased

operational costs. Therefore, high-resolution and robust forecasting models are essential for bridging the gap between planned and real-time energy consumption behaviour.

Despite significant advancements in deep learning-based load forecasting, most existing studies adopt a single-stream feature fusion strategy in which multiple inputs, including weather variables, calendar features, and historical load values, are merged into a single input vector. Although this approach simplifies model design, it introduces several critical limitations. First, the feature interference problem arises because variables with different statistical distributions are forced into a single representation space, which can lead to gradient conflicts during optimization. Second, a temporal–static representation mismatch occurs when time-dependent and non-time-dependent features are treated equally, thereby limiting the model's ability to learn meaningful hierarchical feature representations. Third, when applied to limited datasets such as a single year of hourly measurements, these unified architectures are highly prone to overfitting due to excessive parameter entanglement and insufficient data diversity.

2. Deep Learning

2.1 ANN

Recent advances in artificial intelligence (AI) have enabled data-driven approaches that capture the complex and non-linear interactions influencing residential energy consumption. Artificial Neural Networks (ANN), especially multilayer perceptron (MLP) architectures, can learn the underlying relationships between multiple input features, including weather variables, temporal indicators, and occupancy patterns, and the resultant electricity demand [6]. Studies have demonstrated that ANN-based models outperform traditional statistical and machine learning approaches for short-term load prediction, particularly when dealing with non-linear and stochastic residential consumption behavior [7], [8].

ANNs are capable of approximating complex nonlinear functions through layered transformations and have demonstrated strong performance in mapping structured inputs. such as calendar variables, occupancy schedules, and weather conditions to energy demand outputs [9]. Appliance-level load profiling using ANN remains relatively underexplored, particularly for HVAC and lighting systems, despite their dominant contribution to residential energy consumption. Furthermore, many studies lack transparent algorithmic descriptions and implementation details, which limits the reproducibility and practical adoption [10].

However, despite their strong nonlinear mapping capabilities, ANN-based models suffer from a fundamental limitation in the absence of temporal dependency modeling. Standard feedforward architectures assume independent and identically distributed inputs, meaning each time step is treated independently. This assumption fails to capture sequential dependencies such as thermal inertia, building envelope heat storage, and cumulative weather effects over time. Although recurrent neural networks (RNNs) and LSTM architectures were introduced to address this limitation, they introduce increased computational complexity and training instability, particularly when applied to long sequences or limited datasets [11].

2.2 TCN

To overcome the limitations of recurrent architectures, Temporal Convolutional Networks (TCNs) have emerged as a powerful alternative for sequential modeling. TCNs utilize causal and dilated 1D convolutional operations to capture long-range dependencies while maintaining computational efficiency and parallelization capability [12]. Unlike RNNs, TCNs ensure that predictions at time step t only depend on past inputs ($t-1, t-2, \ldots$), eliminating future information leakage. Furthermore, dilated convolutions allow exponential expansion of the receptive field without increasing network depth significantly.

Recent studies have demonstrated the superiority of TCNs in time-series forecasting tasks such as energy demand prediction, traffic flow forecasting, and industrial process modeling [13]. In building energy applications, TCNs have shown improved stability compared to LSTM-based models due to better gradient propagation and parallel training efficiency. However, a key limitation of standalone TCN architectures is their dependence on large-scale datasets. TCNs typically require extensive historical data to fully exploit their deep hierarchical feature extraction capability. In real-world building systems, data availability is often limited due to sensor installation constraints, system upgrades, or missing data periods. When trained on small datasets (e.g., one-year hourly data), TCNs are prone to overfitting and tend to memorize noise patterns rather than learning generalized energy behavior. This limitation highlights the need for hybrid architectures that can balance temporal learning with structured feature modeling.

3. Methdology

The process begins in Figure 1, with loading a full year of hourly building energy data consisting of 8,760 samples, which represent continuous operational energy consumption of the building. This raw dataset includes key variables such as load demand, weather conditions which temperature and humidity, calendar-related information, and scheduled power profiles, forming the foundational input for the forecasting system.

Next, the dataset undergoes data cleaning and preprocessing to ensure quality and consistency before model development. The main purpose of this stage is to eliminate noise and ensure that data inconsistencies do not negatively affect the learning process of the deep learning model.

After preprocessing, feature engineering is performed to transform raw data into meaningful predictive inputs. This includes the creation of a 24-hour lag feature (PrevLoad_24), along with calendar-based variables such as hour of day, day of week or month, and scheduled power baseline. The 24-hour lag captures the thermal inertia characteristics of buildings, while calendar features represent human behavior and operational patterns. This step is crucial in structuring the data into a format suitable for accurate forecasting. Following feature engineering, all inputs are normalized using Min–Max scaling to ensure they fall within a uniform range. Importantly, scaling parameters are calculated only from the training dataset to avoid data leakage. This normalization stabilizes neural network training and prevents variables with larger numerical ranges, such as load demand, from dominating smaller-scale features like time indicators.

The dataset is then split chronologically into training, validation, and testing sets, typically using a 70%, 15%, and 15% ratio respectively. This ensures the model learns from past data while being validated and tested on unseen future data, preserving the time-series structure and preventing future information leakage. The modeling stage is based on a dual-branch architecture. The first branch is the TCN, which processes sequential features such as temperature, humidity, and 24-hour lag load. This branch uses causal and dilated convolutions to capture long-term temporal dependencies without future data leakage, enabling it to learn weather-driven load dynamics effectively. The second branch is a MLP, which processes static operational features which are day, hour, and scheduled power. This branch focuses on capturing human behavior patterns, fixed scheduling structures, and calendar-based energy usage trends.

The outputs from both branches are then fused through a feature fusion mechanism, where the learned feature vectors are combined and passed through a dense layer. This fusion stage enables the model to learn complex interactions between temporal weather-driven effects and operational scheduling patterns, allowing it to understand relationships such as how weather conditions influence scheduled energy consumption. The combined model is trained end-to-end using the Adam optimizer with Mean Squared Error (MSE) as the loss function. The objective is to minimize the difference between predicted and actual load values by updating weights across both branches simultaneously, ensuring optimal learning of both temporal and static features. Once training is completed, the model is evaluated on the test dataset, which contains unseen data. It takes both TCN and ANN inputs to generate hourly load predictions, simulating real-world deployment conditions in a building energy management system.

Finally, model performance is assessed using evaluation metrics including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2). Lower values of RMSE, MAE, and MAPE indicate higher prediction accuracy, while a higher R^2 value reflects better model fitting and explanatory capability. Overall, the system represents a hybrid intelligent forecasting framework where the TCN branch specializes in learning temporal and weather-related dependencies, the ANN branch captures static operational behavior, and the fusion layer integrates both perspectives. The key innovation lies in separating feature learning into specialized branches before performing deep feature fusion, which improves generalization performance especially in limited or noisy datasets.



Figure 1: Flowchart of purpose methodology

4. Equations

4.1 The 24-Hour Load Lag Variable

To give the model explicit access to historical building behavior, a 24-hour chronological lag feature (PrevLoad_24) is integrated into the input space. Let Y represent the continuous actual building power consumption recorded at hour t. The historical lag variable is mathematically defined as:

$$X_{lag,f} = \begin{cases} 0, & if\ t \leq \tau \\ Y_{t-\tau} & if\ t > \tau \end{cases} \qquad (1)$$

Where τ = 24 hours. This variable ensures the sequence branch can track the building's specific thermal baseline from the previous day, providing a strong reference point for forecasting real-time shifts as illustrate in Figure 2.

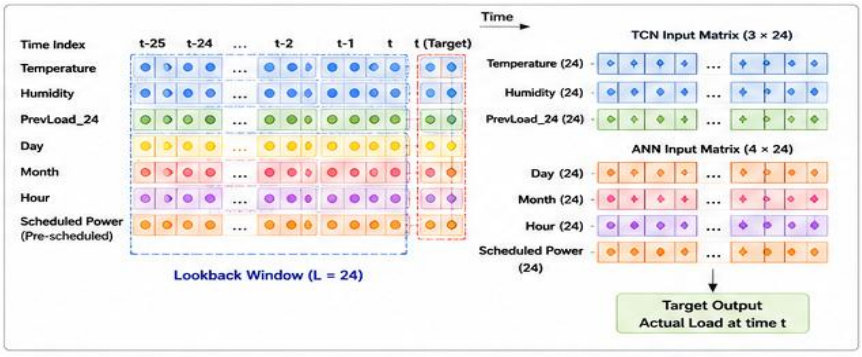


Figure 2: Sliding window-based sequence generation feedback (lookback = 24 hours)

4.2 Feature Matrix Channelization

The complete input space is divided into two distinct processing channels based on the physical properties of the variables:

1. TCN Input Matrix ($X_{TCN,t}$): Focuses on highly volatile weather changes and historical building memory over a 24-hour rolling horizon:

$$X_{TCN,t} = [Temperature_t, Humidity_t, X_{lag,t}]^T \quad (2)$$

2. ANN Input Matrix ($X_{ANN,t}$): Captures static calendar conditions and operational schedules for the current hour:

$$X_{ANN,t} = [Day_t, Month_t, Hour_t, Schedule_Power_t]^T \quad (3)$$

4.3 Data Splitting and MinMax Normalization

The dataset contains exactly 1 year of hourly tracking indices (N = 8,760). To simulate real-world deployment conditions and protect the model against data leakage, a strict cronological data split is applied:

$\boldsymbol{N_{train}}$*: First 70% of the timeline (1 ≤ t ≤ 6,132 hr).*
$\boldsymbol{N_{validation}}$*: Next 15% of the timeline (6,133 ≤ t ≤ 7,446 hr).*
$\boldsymbol{N_{test}}$*: Final 15% of the timeline (7,447 ≤ t ≤ 8,760 hr).*

To scale the inputs uniformly, the feature ranges are bounded between 0 and 1 using a MinMax normalization envelope. Crucially, the scaling parameters are calculated strictly from the training partition to ensure zero information leakage from the validation or testing sets:

$$x_{scaled} = \frac{x_{raw} - \min(X_{train})}{\max(X_{train}) - \min(X_{train})} \quad (4)$$

4.3 Multi-Input Sliding Window Array Generation

To enable parallel training across the sequential TCN and tabular ANN streams, the normalized data is converted into sliding lookback windows. Given a lookback window length of L = 24 hours, each training sample index i is structured as follows:

The TCN branch ingests a 3D tensor of historical sequences:

$$\boldsymbol{X_{TCN_seq}^{(i)}} = \begin{bmatrix} \boldsymbol{T_{i-24}} & \boldsymbol{T_{i-23}} & \cdots & \boldsymbol{T_{i-1}} \\ \boldsymbol{H_{i-24}} & \boldsymbol{H_{i-23}} & \cdots & \boldsymbol{H_{i-1}} \\ \boldsymbol{L_{i-24}} & \boldsymbol{L_{i-23}} & \cdots & \boldsymbol{L_{i-1}} \end{bmatrix} \boldsymbol{\epsilon \mathbb{R}^{3x24}} \quad (5)$$

The ANN branch ingests a parallel sequence grid matching the same lookback timeline:

$$X_{ANN_seq}^{(i)} = \begin{bmatrix} D_{i-24} & D_{i-23} & \cdots & D_{i-1} \\ M_{i-24} & M_{i-23} & \cdots & M_{i-1} \\ H_{i-24} & H_{i-23} & \cdots & H_{i-1} \\ P_{i-24} & P_{i-23} & \cdots & P_{i-1} \end{bmatrix} \epsilon \mathbb{R}^{4x24} \quad (6)$$

The forecasting target maps to the actual power reading at the current hour index:

$$Y_{target}^{(i)} = Y_{actual,i} \epsilon \mathbb{R}^1 \quad (7)$$

4.4 Network Layout and Layer Definitions

The model architecture leverages multi-input datastores (combine) to process the sequence matrices through two parallel networks simultaneously before fusing the representations:

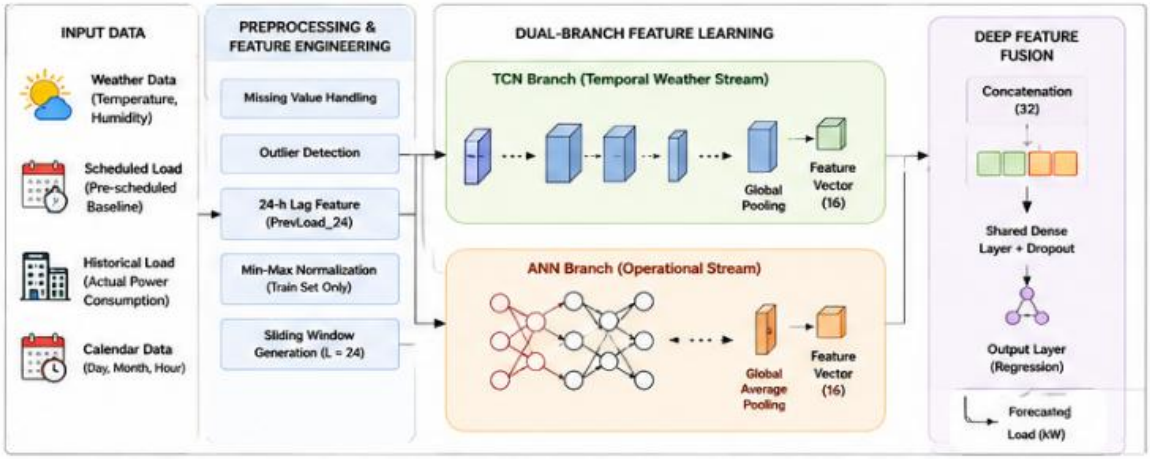


Figure 3: Overall architecture of the proposed multi-branch deep feature fusion framework integrating Temporal Convolutional Network (TCN) and Artificial Neural Network (ANN) for short-term building load forecasting.

The proposed model consists of two parallel branches: a TCN branch for sequential feature extraction and an ANN branch for structured feature learning. Outputs are fused using a dense layer for final prediction. Input features are separated into temporal and static components to reduce feature interference.

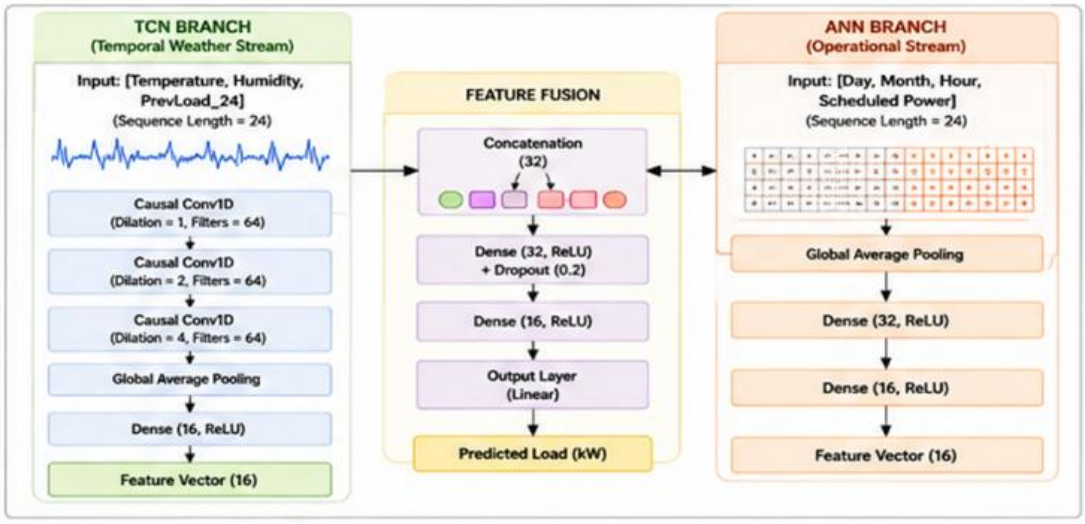


Figure 4: Dual-stream input feature separation showing the division of temporal variables (TCN branch) and operational variables (ANN branch).

4.4.1 Dilated Causal TCN Layout

The weather and lag memory inputs pass through stacked 1D convolutional layers with expanding dilation factors (d = 1, 2). The causal convolution operation for a sequence element s using a kernel filter f of size K=3 is expressed as:

$$\boldsymbol{F}_{(s)} = (x^*{}_d f)(s) = \sum_{j=0}^{K-1} f(j) \cdot x_{s-d \cdot j} \qquad (8)$$

The sequential outputs are condensed into a static feature representation using a global average pooling layer, which aggregates information across the entire historical timeline:

$$\mathbf{h}_{TCN} = \frac{1}{L} \sum_{k=1}^{L} \mathbf{Z}_{\text{conv,k}} \in \mathbb{R}^{64\text{x}1} \qquad (9)$$

This pooled vector passes through a fully connected layer to produce a final 16-element weather feature vector:

$$\mathbf{f}_{\text{TCN}} = \mathbf{max}\,(\mathbf{0}, \mathbf{W}_{\text{tcn_out}}\,\mathbf{h}_{\text{TCN}} + \mathbf{b}_{\text{tcn_out}}) \in \mathbb{R}^{16x1} \qquad (10)$$

4.4.2 Operational ANN Layout

To process the multi-feature timeline within the schedule branch, the input matrix is collapsed via global average pooling and passed through stacked dense layers:

$$\mathbf{h}_{\mathbf{ANN_1}} = \max(\mathbf{0}, \mathbf{W}_{ann_1}\,\mathbf{h}_{\text{pooled_ANN}} + \mathbf{b}_{ann_1}) \in \mathbb{R}^{32x1} \quad (11)$$

$$\mathbf{f}_{\mathbf{ANN_1}} = \max(\mathbf{0}, \mathbf{W}_{ann_2}\,\mathbf{h}_{\text{ANN_1}} + \mathbf{b}_{ann_1}) \in \mathbb{R}^{16x1} \qquad (12)$$

This architecture projects both streams into matching 16-element feature spaces, preparing them for deep fusion.

4.4.3 Deep Feature Fusion and Joint Optimization

The network combines the specialized representations using a channel-wise concatenation layer:

$$\mathbf{f}_{\text{fused}} = \mathbf{f}_{TCN} \oplus \mathbf{f}_{\text{ANN}} \in \mathbb{R}^{32x1} \qquad (13)$$

This fused vector is routed through a shared dense layer with a dropout rate of 0.2 to prevent co-dependency among hidden parameters:

$$\mathbf{h}_{\text{shared}} = \max(0, \mathbf{W}_{\text{f1}}\mathbf{f}_{\text{fused}} + \boldsymbol{b}_{\text{f1}}) \in \mathbb{R}^{32x1} \qquad (14)$$

The final continuous building load prediction $\hat{Y}_t$ is generated at the output node:

$$\hat{Y}_t = \mathbf{W}_{\text{out}}\mathbf{h}_{\text{shared}} + \boldsymbol{b}_{\text{out}} \qquad (15)$$

The entire model graph is optimized end-to-end via backpropagation using an Adam optimizer to minimize the Mean Squared Error (MSE) loss function:

$$\mathcal{L}_{MSE} = \frac{1}{N_{samples}} \sum_{i=1}^{N_{samples}} \left(Y_{actual}^{(i)} - Y'^{(i)}\right)^2 \qquad (16)$$

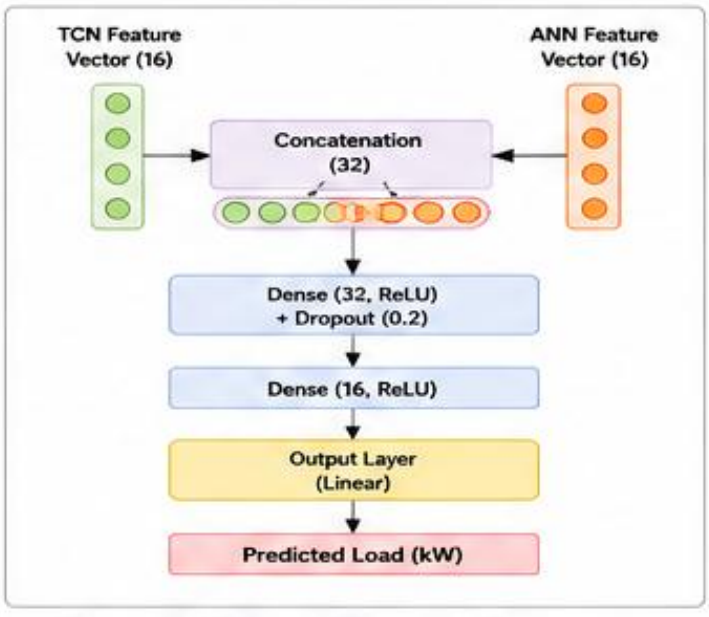


Figure 5: Deep feature fusion mechanism

5. Result and Discussion

The forecasting performance of the deep fusion hybrid framework was evaluated against standalone ANN and TCN baseline models. All models were tested on the 15% unseen test partition using four standard regression indicators: Root Mean Squared Error (RMSE), Normalized RMSE (NRMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Figure 6 and Figure 7 shows the standalone ANN achieved an RMSE of 18.03 kW and an R^2 of 0.99141, benefiting from its ability to immediately map the highly predictive Scheduled Power target variable.

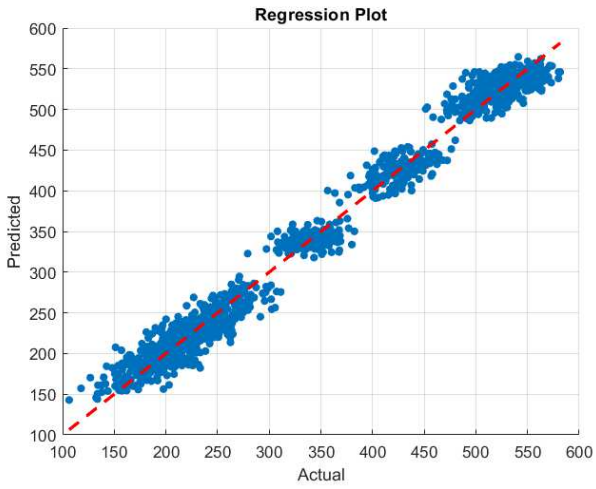


Figure 6: ANN regression plot Actual vs Predicted

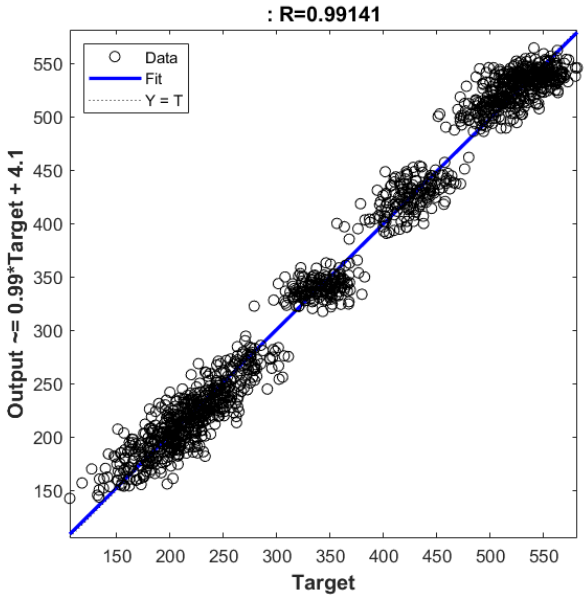


Figure 7: ANN R^2 value

The results in Figure 8 demonstrate that the raw pre-scheduled building baseline exhibits a massive error against reality, yielding an unoptimized Baseline RMSE of 137.8324 kW. All three machine learning models successfully minimized this baseline discrepancy. The Standalone TCN network achieved an RMSE of 25.75 kW paired with an R^2 score of 0.9649. While the TCN successfully tracks sequential climate fluctuations, it suffers from minor time-lag smoothing errors due to its 24-hour causal filter horizon.

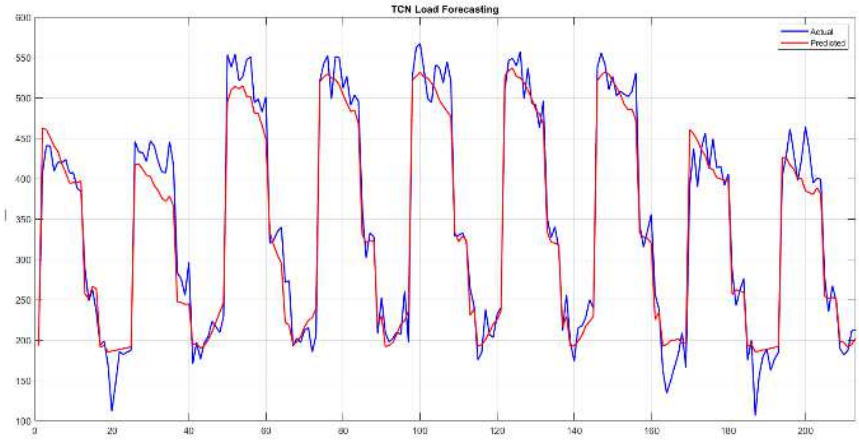


Figure 8: Standalone TCN load forecasting

Table 1 Model performance Metrics

Model Architecture	**RMSE (kW)**	**MAE (kW)**	**MAPE (%)**	**NRMSE**	**R^2**
Standalone TCN	25.75	20.46	6.55	0.0550	0.9649
Proposed Hybrid TCN-ANN	**20.03**	**16.04**	**5.76**	**0.0489**	**0.9721**
Standalone ANN	18.03	14.45	4.84	0.0379	0.99141

The Proposed Hybrid TCN-ANN Framework achieved an excellent tracking balance, yielding an RMSE of 20.03 kW, an MAE of 16.04 kW, and a Mean Absolute Percentage Error (MAPE) of 5.76%. Crucially, the hybrid model generated a high Coefficient of Determination ($R^2 = 0.9721$) and a low NRMSE of 0.0489 as shown in Table 1. By transitioning from a basic addition structure to a deep feature sequence block, the network effectively balances the instant schedule features from the ANN with the long-term weather feature representations from the TCN. Figure 9 shows TCN training progress, where this integration prevents the deep learning layers from overfitting on the limited 1-year data sequence, establishing the hybrid model as a balanced, reliable framework for real-world microgrid operations.

- Future Predicted Load: 526.64kW

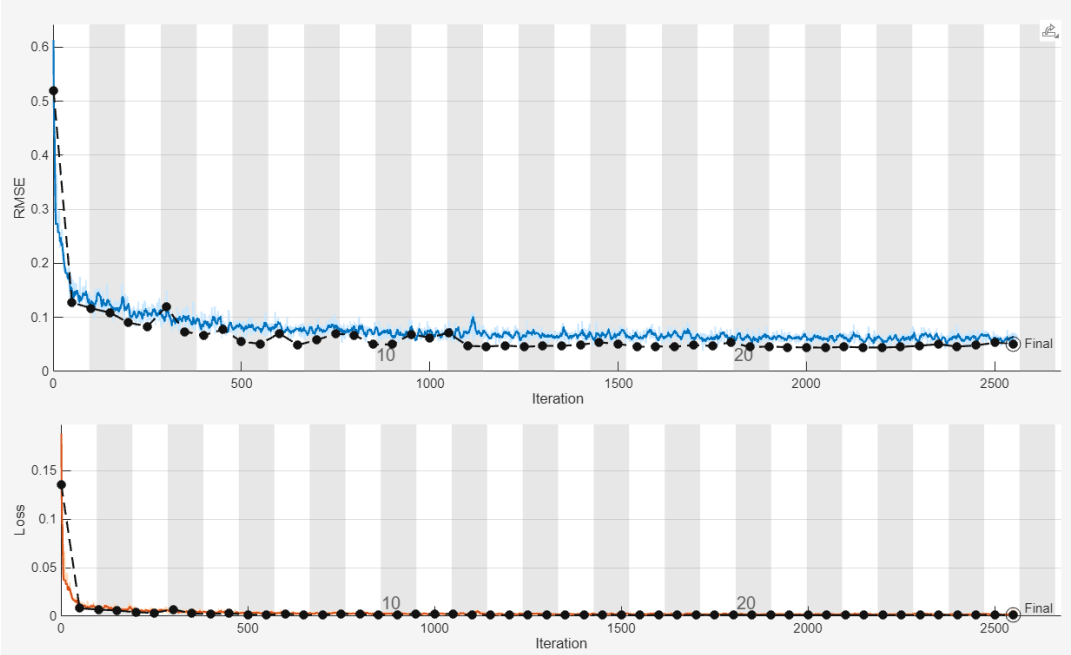


Figure 9: Training TCN progress

The correlation between actual testing building loads and model forecasts was evaluated geometrically across a linear identity scatter plane ($Y_{Actual}\ vs\ Y_{Predicted}$) as shown in Figure 10. The hybrid model achieves tight, balanced clustering along the diagonal line, supported by a strong Coefficient of Determination ($R^2 = 0.9721$) and a low NRMSE of 4.89%. This tight distribution in Figure 11 confirms that the deep fusion layer successfully weights the weather adjustments against the pre-scheduled baseline, preventing gradient conflicts and ensuring stable tracking performance during complex operational transitions.

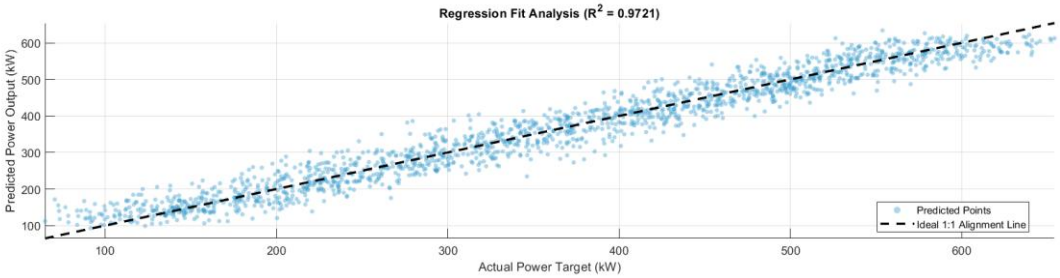


Figure 10: Regression Analysis TCN-ANN

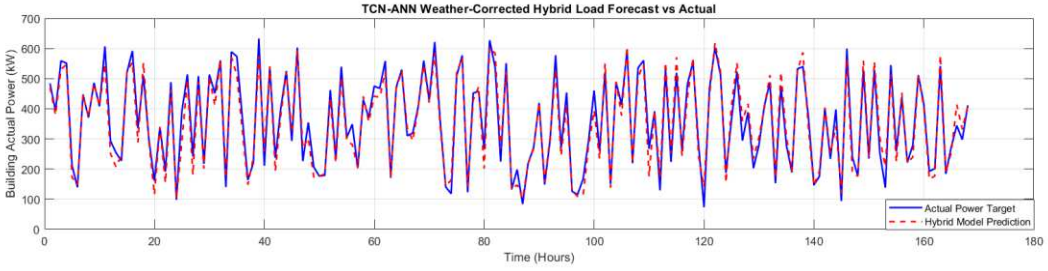


Figure 11: Hybrid TCN-ANN Load Forecasting

6. Conclusion

This study successfully developed and evaluated a multi-branch deep feature fusion hybrid network that integrates Temporal Convolutional Networks (TCN) and Artificial Neural Networks (ANN) for short-term building load forecasting. Using one year of hourly operational building data (8,760 samples), the proposed architecture was designed to process heterogeneous inputs through separate computational pathways, where the ANN branch captured structural and cyclical operational scheduling patterns, while the causal dilated TCN branch learned temporal dependencies from weather-related variables, including temperature and humidity, together with historical load memory represented by PrevLoad_24.

The experimental results demonstrated the effectiveness of the proposed framework, with the original building schedule exhibiting a baseline error of 137.8324 kW, which was substantially reduced by the hybrid model to an RMSE of 20.03 kW while achieving a high coefficient of determination (R^2) of 0.9721. The incorporation of a deep feature fusion mechanism through latent feature integration enabled the model to mitigate feature interference, reduce gradient conflicts, and limit overfitting on relatively small datasets, resulting in a robust and reliable forecasting framework suitable for real-world building energy management and microgrid applications.

For future work, the model can be further enhanced by expanding the input feature set to include appliance-level energy consumption data, local solar irradiance measurements, and building occupancy information to improve forecasting accuracy and weather-correction capability. Furthermore, the proposed forecasting framework can be integrated with a hardware-based automated Building Energy Management System (BEMS) to enable real-time energy optimization and control. Future studies may focus on incorporating HVAC control strategies, IoT-enabled sensors, and intelligent actuators that utilize forecasted load information to automatically regulate building operations, thereby improving energy efficiency, reducing peak demand, and supporting autonomous energy management in smart building environments.

7. Data Availability

The dataset used and analysed during the current study is not publicly available due to institutional and research data-sharing restrictions but is available from the corresponding author on reasonable request.

8. Acknowledgments

This research is supported by Ministry of Higher Education (MOHE) through Fundamental Research Grant Scheme (FRGS/1/2024/TK08/UTHM/03/9).

References

[1] IEA (2022), World Energy Outlook 2022, IEA, Paris https://www.iea.org/reports/world-energy-outlook-2022, Licence: CC BY 4.0 (report); CC BY NC SA 4.0 (Annex A)

[2] IEA (2022), Energy Efficiency 2022, IEA, Paris https://www.iea.org/reports/energy-efficiency-2022, Licence: CC BY 4.0

[3] Cheikh Elekbir Sidi Lekhel, Rita Mbayed, Oleksandr Velihorskyi, Oleksandr Husev, Eric Monmasson, “Generic residential load profile generator based on weather data and occupancy”, Mathematics and Computers in Simulation, Volume 237, 2025, Pages 373-389, ISSN 0378-4754, https://doi.org/10.1016/j.matcom.2025.04.044.

[4] Hasan Rafiq, Prajowal Manandhar, Edwin Rodriguez-Ubinas, Juan David Barbosa, Omer Ahmed Qureshi, “Analysis of residential electricity consumption patterns utilizing smart-meter data: Dubai as a case study”, Energy and Buildings, Volume 291, 2023, 113103, ISSN 0378-7788, https://doi.org/10.1016/j.enbuild.2023.113103.

[5]Cuce, P.M., Cuce, E. Role of HVAC in building energy consumption: a critical review. *J Therm Anal Calorim* **151**, 3059–3080 (2026). https://doi.org/10.1007/s10973-026-15322-9

[6] T. Hong and P. Wang, "On the impact of demand response: Load shedding, energy conservation, and further implications to load forecasting," 2012 IEEE Power and Energy Society General Meeting, San Diego, CA, USA, 2012, pp. 1-3, doi: 10.1109/PESGM.2012.6345192.

[7] Ruifan Zheng, Dongfang Yang, Lei Gu, Rendong Shen, Qingsong An, Jun Zhao, Jiahui Wu, “Towards short-term load forecasting for office buildings: A deep learning approach combined with occupant behavior model”, Energy Conversion and Management, Volume 350,2026,120983, ISSN 0196-8904, https://doi.org/10.1016/j.enconman.2025.120983.

[8] Amar S. Abdul-Zahra, Luttfi A. Al-Haddad, I.A. Abdulsahib, Habib Kraiem, Aymen Flah, “Sustainable thermal load prediction in residential buildings using machine learning: A case study analysis”, Results in Engineering, Volume 27, 2025, 106872, ISSN 2590-1230, https://doi.org/10.1016/j.rineng.2025.106872.

[9] Bryant, J. S., H. Li, N. Maheepala, et al. 2026. “An Overview of Artificial Intelligence and Machine Learning Approaches for Building Energy Analysis, Characterization, Control, and Grid Support Services Provision.” *Wiley Interdisciplinary Reviews: Energy and Environment*15, no. 2: e70029. https://doi.org/10.1002/wene.70029.

[10] El Alaoui M, Rougui M. Examining the Application of Artificial Neural Networks (ANNs) for Advancing Energy Efficiency in Building: A Comprehensive Reviews. J Sustain Res. 2024;6(1):e240001. https://doi.org/10.20900/jsr20240001.
[11] Martínez Comesaña, M., Febrero-Garrido, L., Troncoso-Pastoriza, F., & Martínez-Torres, J. (2020). Prediction of Building's Thermal Performance Using LSTM and MLP Neural Networks. *Applied Sciences*, *10*(21), 7439. https://doi.org/10.3390/app10217439
[12] Kumar A, Bag S, Srivastava V (2026), "Efficient prediction of transient residual stress in directed energy deposition process using temporal convolutional networks". 4D Printing, Vol. 2 No. 1 pp. 91–98, doi: https://doi.org/10.1108/4DP-09-2025-0020
[13]Ziyuan Zhang, Jianzhou Wang, Danxiang Wei, Yurui Xia (2023),"An improved temporal convolutional network with attention mechanism for photovoltaic generation forecasting".Engineering Applications of Artificial Intelligence ,Volume 123, Part A, August 2023, 106273, https://doi.org/10.1016/j.engappai.2023.106273