Dual-Framework Deep Learning Architectures for Electricity Consumption Forecasting: Hybrid LSTM-GRU and Convolutional-Temporal Networks

1st H. S. Shreyas, 2nd Dhruv Prrasana, 3rd Prajeesha, 4th Anuradha M
Dept. of Electronics & Communication Engg,
PES University
Bengaluru, India
shreyas.hosur2506@gmail.com, dhruvprrasana@gmail.com, prajeeshaemmanuel@pes.edu, anuradha@pes.edu

***Abstract*—Forecasting electricity consumption with accuracy has a highly significant impact on smart grids. With the new advanced meters, power systems will be able to manage resources and grids efficiently. This research presents two different deep learning frameworks on the same time-series datasets. One of these frameworks is a Hybrid LSTM-GRU model while the other is a Convolutional-Temporal (TCN-LSTM) Network based model. The first framework mixes numerous layers of Long Short-Term Memory (LSTM) networks in a top-down manner (in a label hierarchy) with gated recurrent units (GRU) in a sequential manner. This helps learn complex temporal dependencies and will have an effect on the successive execution of temporal dependencies learned by the LSTM layers. Because of this, the model can learn temporal dependencies at various levels of the label hierarchy, from general (top-most) to very specific (bottom-most). The hybrid LSTM-GRU model achieves superior performance with a test RMSE of 0.0076 kWh, MAE of 0.0053 kWh, MAPE of 5.78%, and R^2 score of 0.9692, demonstrating significant improvements over individual base models. The TCN-LSTM model consists of combining dilated causal convolutions and feed-forward processing with a LSTM. High-frequency patterns can be efficiently extracted through deep stacks of dilated causal convolutions.**

***Index Terms*—electricity load forecasting, hybrid deep learning, LSTM, GRU, temporal convolutional networks, smart grids, time-series prediction**

I. Introduction

The projection of electricity consumption has now become one of the essential tasks of the modern power system. It assists the utility in developing an optimal generation schedule while minimizing operational expenses and stabilizing as well as allocating resources. Deep learning has shown great potential in forecasting time series recently. It has superior capabilities than conventional methods for modeling complex and nonlinear relationships.

RNN or recurrent neural network mainly LSTM and LSTM network with gating mechanism are very powerful to learn sequential data with long-term dependency. Through their complex gating units, the information can flow through them for a long time through a sequence and learn which information to remember/forget. While RNN models learn data sequentially, the Temporal Convolutional Network (TCN) architecture of time-series forecasting tasks is very much popular for its efficiency of capturing multi-scale temporal patterns via dilated convolution and leveraging parallel computing capabilities of training.

An individual LSTM architecture and an individual GRU architecture both have benefits and drawbacks. The LSTM network can (i.e., with its three gates being the input gate, forget gate, output gate) learn long-term dependencies very efficiently. Accordingly, it is well suited for modelling long term dependencies spanninig one or multiple days, as well as periodicity exhibited by electricity consumption data. In comparison, the update and reset gates of the GRU network can regulate information.

This research outlines the construction of two new hybrid structures to predict electricity demand: (1) a sequential hybrid LSTM-GRU model which combines recurrent components and (2) a Convolutional-Temporal hybrid TCN-LSTM model which combines convolutional and recurrent components. Both frameworks are evaluated on identical preprocessing of time-series datasets spanning one calendar year with fifteen-minute sampling intervals. Both models integrate temporal and statistical features derived from electricity usage patterns and environmental data through a unified preprocessing pipeline.

This research makes the following contributions: (1) development of two distinct novel hybrid architectures (LSTM-GRU and TCN-LSTM) for electricity load forecasting demonstrating complementary approaches; (2) the application of feature engineering which combines cyclical encoding with a set of statistical lags that are applied on multiple scales; (3) a comprehensive experimental validation to find the efficacy of the proposed work; (4) demonstration of a global scaling property to train one single unified model across all consumption entities; (5) detailed explanation of methodology which includes data pre-processing pipeline from raw data to entry construction. In addition, a global scaling strategy is proposed that enables one single model training across all consumption entities with the TCN-LSTM architecture. In conclusion, we show extensive experimental validation and analysis on the algorithm robustness and generalisation performance on an industrial-scale real-world dataset, along with a detailed description of the entire pre-processing pipeline

from data to sequence.

The rest of the document will be organized as follows: Section II reviews literature pertinent to our study, encompassing deep learning techniques for electricity load forecasting. The proposed dual handling and dual architecture methodologies are explained in Section III. The experimental results and details are presented in Section IV.

II. Literature Survey

The electricity industry has seen significant improvements in time series forecasting with deep learning. Researchers have researched various neural network structures to improve their prediction accuracy.

Improvements based on attention push the field further. Bao et al. [1] in their paper suggested a short term load forecasting method using dual attention with bidirectional LSTM networks. They showed that the use of correlation based feature selection and temporal importance weighting to improve prediction quality by feature and temporal attention.

As of now, in terms of better performance hybrid and ensemble methods are known to perform well. Unlu et al. [2] performed a comparative study on integrated deep learning models for load forecasting. They compared CNN, RNN, LSTM and GRU in integrated models. However, their studies reveal that integrated models can outperform individual models as they use complementary patterns over different time scales and feature spaces. Subsequently, other hybrid combinations have been studied. Vimalraj and Jothi Swaroopan [3] analyzed stacked GRU-RNN and a bidirectional LSTM network for renewable energy forecasting in smart grids. They analyzed the effect of different combinations of architectures on prediction performance.

A powerful substitute for recurrent architectures is the Temporal Convolutional Network. With a focus on dilated convolutions for capturing multi-scale temporal dependencies in energy data, Lara-Benítez et al. [9] demonstrated the effectiveness of temporal convolutional networks on national electricity demand and electric vehicle charging data from Spain. Their work showed that causal convolutions with exponentially growing dilation factors allow long sequences to be processed in parallel while preserving temporal causality and computational efficiency, with TCN outperforming LSTM across an extensive search over more than a thousand model configurations. The application of TCN to patterns of electricity consumption that show strong periodic and cyclical characteristics at various temporal scales is validated by this work. An alternative to recurrent depth for capturing long-range dependencies without sequential computation is receptive field expansion through dilation.

By utilizing complementary strengths, hybrid architectures that combine LSTM layers and Temporal Convolutional Networks have demonstrated efficacy. TCN–LSTM hybrids were studied by Heng et al. [10], who proposed a CEEMDAN–TCN–LSTM model for power load forecasting and showed that convolutional feature extraction combined with sequential recurrent modeling captures both long-range dependencies and high-frequency variations. Their work integrates dilated convolutions with LSTM processing in a model that also introduces sample entropy to reorganize decomposed signal components, creating an architecture in which the LSTM processes multi-scale temporal patterns extracted by TCN layers acting as feature encoders to improve sequential modeling and forecasting accuracy.

Global modelling methods are proposed as a substitute to entity-specific training. Grabner et al. [11] propose a global modeling framework for load forecasting across a large number of individual consumer loads, low-voltage feeders, and transformer stations in distribution networks. They show through experiments that training a single global model across all loads helps the model learn common behaviour patterns while outperforming entity-specific local models for aggregated loads, with an additional localization mechanism needed to also achieve gains for individual, highly volatile consumers. This method helps in finding universal signatures which are useful for entities with different consumption behaviour and demand patterns. It also reduces modelling complexity as well as computational cost in comparison to having individual models for each entity.

The encoder–decoder framework utilized in time-series forecasting has facilitated advanced sequence-to-sequence modeling. Du et al. [12] proposed a sequence-to-sequence framework using stacked LSTM cells for both the encoder and decoder, applying it to multivariate air quality forecasting and showing that this architecture outperformed shallow models and baseline recurrent networks. This supports the broader case for encoder–decoder integration in sequential forecasting tasks. In this setup, TCN layers extract multi-scale features, which are then fed into LSTM layers to improve sequential modeling. This creates an implicit encoder–decoder structure, even though a direct sequence-to-sequence approach isn't used.

These investigations collectively suggest that hybrid architectures, which integrate various recurrent and convolutional neural network elements, present viable strategies for improving electricity load forecasting. This study builds upon prior research by systematically investigating two distinct, complementary hybrid frameworks: sequential LSTM-GRU integration and convolutional-temporal TCN-LSTM integration. To facilitate a fair comparison and demonstrate the versatility of hybrid methodologies, these frameworks are assessed using the same datasets.

III. Methodology

Estimating the electricity load forecasts requires systematic transformation of the raw consumption data into appropriate deep-learning representations. In this part, we first give a unified data preprocessing pipeline shared by both frameworks followed by providing detailed technical architectures for each framework.

A. Unified Data Preprocessing and Feature Engineering

Both the Python LSTM–GRU and MATLAB TCN–LSTM models employ the same preprocessing pipeline for the same dataset to ensure a fair comparison. While both frameworks share identical preprocessing pipelines and datasets, they are implemented in different environments—Python (TensorFlow/Keras) for the LSTM–GRU model and MATLAB (Deep Learning Toolbox) for the TCN–LSTM model. This reflects practical constraints of the research workflow and does not affect the preprocessing pipeline or data splits. However, it does introduce framework-level differences in default numerical precision and optimiser behaviour that should be considered when interpreting cross-framework performance comparisons.

1) Data Normalization: Raw electricity consumption data x_i and environmental features (temperature) are standardized to zero mean and unit variance using standardization:

$$x_{\text{normalized}} = \frac{x - \mu}{\sigma} \tag{1}$$

2) Temporal Feature Engineering: Circular encoding, such as sine and cosine, is implemented on hour-of-day, day-of-week, and month-of-year to retain the circular format of temporal data. A binary weekend indicator captures behavioral differences weekly.

3) Statistical Feature Generation: Rolling average features capture trends at 7-day ($k = 336$) and 30-day ($k = 1440$) intervals. Lag features capture consumption for the previous 15 minutes, 24 hours prior, and 7 days prior. Combined, these create a 14-dimensional feature vector for each timestep.

4) Sequence Construction and Data Splitting: The input sequences are created using a sliding window-based approach. The Python LSTM–GRU model uses a length of $L = 168$ (one week), while the MATLAB TCN–LSTM model uses a length of $L = 60$ (approx. 15 hours). The LSTM–GRU model uses $L = 168$ (one full week) to capture long-range weekly periodicity intrinsic to individual entity behaviour. The TCN–LSTM model uses $L = 60$ (approximately 15 hours) since dilated causal convolutions expand the effective receptive field exponentially—achieving coverage equivalent to much longer windows—while maintaining lower computational overhead for global training across 250 entities. The dataset is structured in chronological order and divided into Train (60%), Validation (20%), and Test (20%) sets.

B. Framework : Hybrid LSTM-GRU Architecture (Python Implementation)

The first framework sequentially integrates LSTM and GRU layers to extract complex long-range patterns and refine representations.

1) Training: Loss and Optimization: Loss is defined as $\mathcal{L}_{\text{total}} = \frac{1}{N}\sum(\hat{y}_i - y_i)^2 + \lambda \sum w^2$ with $\lambda = 0.001$. We utilize the Adam optimizer with gradient clipping and a ReduceLROnPlateau scheduler.

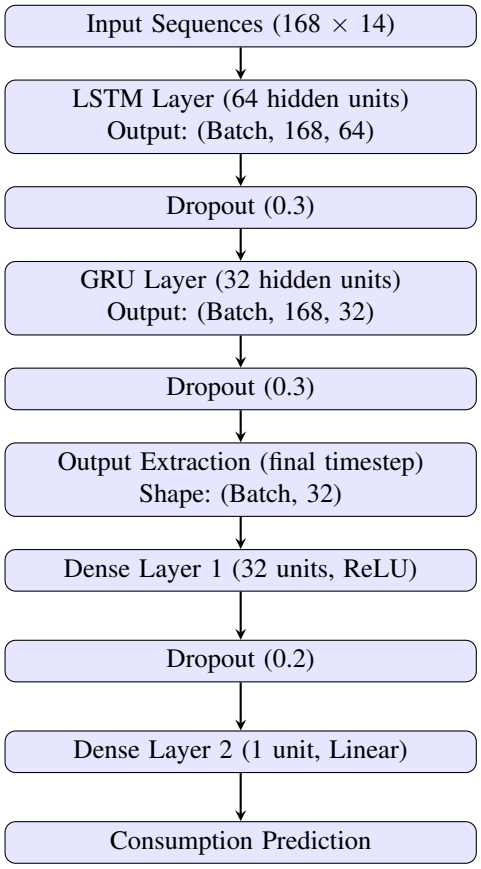

Fig. 1. Architectural flow of the Hybrid LSTM-GRU model.

TABLE I
MATHEMATICAL FORMULATIONS OF LSTM AND GRU GATING MECHANISMS

Component	Mathematical Formulation
Long Short-Term Memory (LSTM)	
Forget Gate (f_t)	$\sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$
Input Gate (i_t)	$\sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
Cell Candidate ($\tilde{C}_t$)	$\tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$
Cell State (C_t)	$f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$
Output Gate (o_t)	$\sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$
Hidden State (h_t)	$o_t \odot \tanh(C_t)$
Gated Recurrent Unit (GRU)	
Reset Gate (r_t)	$\sigma(W_r \cdot [h_{t-1}, x_t] + b_r)$
Update Gate (z_t)	$\sigma(W_z \cdot [h_{t-1}, x_t] + b_z)$
Candidate State ($\tilde{h}_t$)	$\tanh(W_h \cdot [r_t \odot h_{t-1}, x_t] + b_h)$
Hidden State (h_t)	$(1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$

C. Framework: Convolutional-Temporal Hybrid (MATLAB Implementation)

The second framework employs TCN with dilated causal convolutions followed by LSTM processing to capture multi-scale patterns.

1) TCN and Scaling Formulations: Dilated causal convolution is defined as $\text{TCN}_d(t) = \sum_{k=0}^{K-1} f[k] \cdot X[t - d \cdot k]$. Receptive field RF grows as $K \cdot (2^n - 1)$.

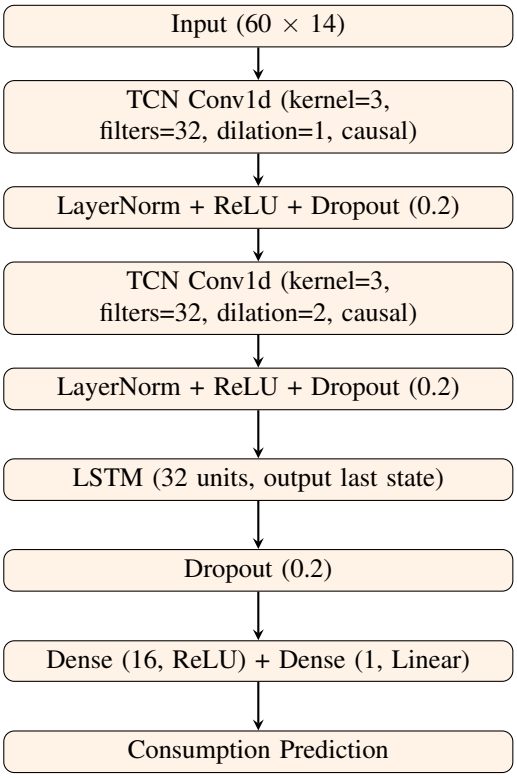

Fig. 2. Architectural flow of the TCN-LSTM model.

2) Encoder-Decoder Integration: TCN acts as an encoder to extract multi-scale features, while the LSTM acts as a decoder to process the sequence and output the final hidden state.

3) Training Configuration: Loss is measured by MSE regression. Adam optimizer with a piecewise learning rate schedule, reducing by 0.5 every 7 epochs, and a gradient threshold of 1.0.

IV. Results and Analysis

Metrics of both (Python) LSTM-GRU and (MATLAB) TCN-LSTM have been obtained through training and testing on existing consumer load profiles. All (Python) LSTM-GRU metrics are reported in the original physical domain (kWh) after applying inverse standardization. On the other hand, (MATLAB) TCN-LSTM metrics are reported in normalized space as per the global scaling strategy used for 250 consumer entities.

Future work will address this by computing entity-level inverse transformations to enable direct kWh-domain comparison.

A. Framework A: Hybrid LSTM-GRU Performance

1) Quantitative Metrics: The model achieves a test RMSE of 0.0076 kWh, MAE of 0.0053 kWh, MAPE of 5.78%, and an R^2 score of 0.9692, indicating that the model explains 96.92% of variance in consumption patterns:

2) Training Convergence and Generalization: Fig. 3 illustrates the training and validation MAE curves over 133 epochs. Both curves converge smoothly, with training MAE stabilising near 0.147 and validation MAE near 0.088 in normalized space. The validation MAE appearing lower than training MAE is an expected artifact of dropout regularization (rates of 0.3 and 0.2) being active during training but disabled during inference, consistent with standard regularized network behaviour [2]. The validation-to-train RMSE gap of 2.61% in normalized space confirms excellent generalization with minimal overfitting, validating the regularization strategy employed.

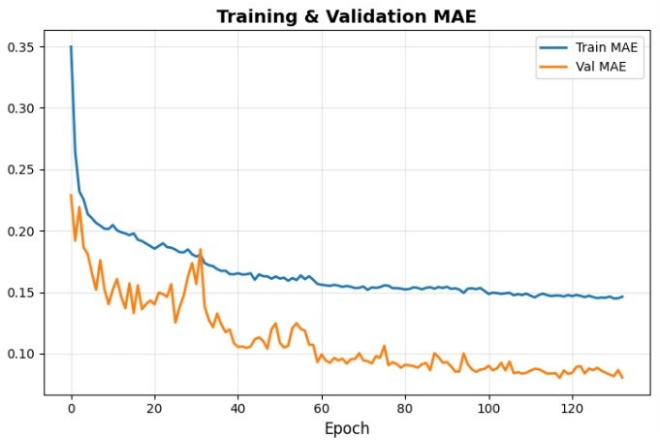

Fig. 3. Training and validation MAE convergence over 133 epochs for the Hybrid LSTM-GRU model. The gap between curves is attributable to dropout deactivation during inference.

3) Prediction Quality Analysis: The test set overlay of actual versus predicted consumption across the first 1000 samples (Fig. 4) demonstrates that the model accurately tracks periodic daily consumption patterns and steady-state baseline behavior. The characteristic recurrent limitation of slight underestimation at sharp consumption spikes is observed, attributable to the smoothing effect of temporal averaging over the 168-timestep lookback window. To mitigate this, weighted loss functions—such as Weighted MSE that assign higher penalties to peak prediction errors relative to steady-state errors—were explored conceptually but not implemented in the current study. Given the importance of peak load prediction for grid stability and blackout prevention, this represents a high-priority direction for future work. This is consistent with findings by Podgorelec et al. [7], who noted that longer lookback horizons improve long-range dependency capture at the cost of peak transient response.

The actual versus predicted scatter plot illustrated in Fig. 5 showed a high linear relationship and strong alignment along the perfect prediction diagonal. This relationship held for the full range of consumption from 0.03 to 0.25 kWh.

4) Error Distribution: The histogram of the prediction error (Fig. 6), which has a zero-mean and an approximately Gaussian distribution, shows that most errors fall within the ± 0.01 kWh band. The negligible mean error confirms the

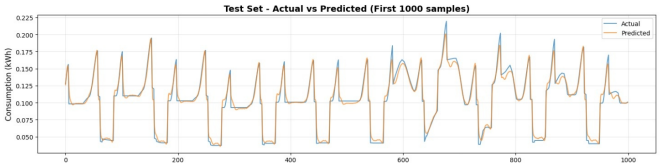

Fig. 4. Test set actual versus predicted consumption (first 1000 samples). The model closely tracks periodic daily consumption patterns, with minor underestimation observed at sharp consumption peaks due to the 168 timestep averaging effect.

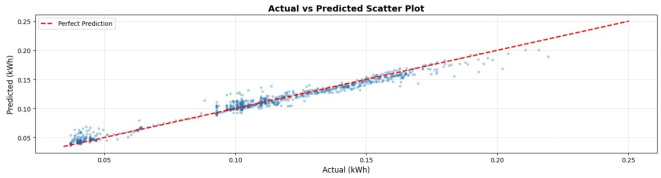

Fig. 5. Actual versus predicted scatter plot for the hybrid LSTM-GRU model. Points align tightly along the perfect prediction diagonal across the full consumption range (0.03–0.25 kWh), confirming low-bias forecasting with minor scatter at low amplitudes.

absence of systematic bias in the predictions. Residual tail errors extending to ± 0.04 kWh correspond to the spike underestimation events identified in Fig. 4 and represent less than 1% of total predictions.

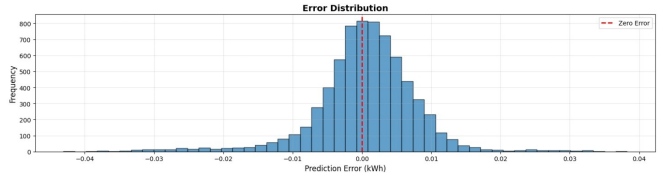

Fig. 6. Prediction error distribution for the hybrid LSTM-GRU model. The near-Gaussian profile centered at zero indicates no systematic bias.

B. Framework B: TCN-LSTM Global Model Performance

1) Aggregate Performance Across Entities: Table II summarizes the global model performance statistics across the full consumer population, with the full training summary output shown in Fig. 7. The model achieves an average RMSE of 0.034768 with a best-entity RMSE of 0.017820 and a worst-entity RMSE of 0.064303. The full model training time of 1253.30 seconds across all 250 entities demonstrates the computational scalability of the global scaling approach.

TABLE II
TCN-LSTM GLOBAL MODEL AGGREGATE PERFORMANCE
(NORMALIZED SPACE, 250 ENTITIES)

Metric	Value
Average RMSE (all entities)	0.034768
Best Entity RMSE	0.017820
Worst Entity RMSE	0.064303
Number of Entities	250
Lookback Window	60 timesteps

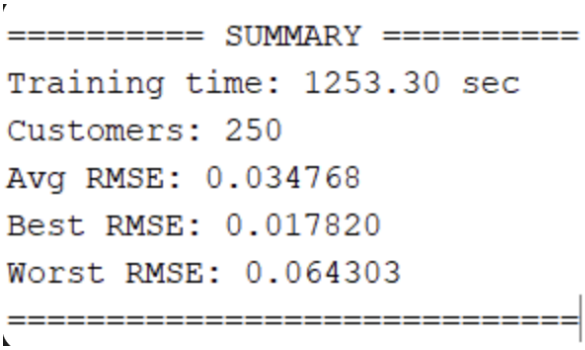

Fig. 7. TCN-LSTM training summary output showing aggregate statistics across 250 consumer entities.

2) Per-Entity Error Distribution: The RMSE values for 250 consumer entities are observed to be right skewed. Further note that almost 190 entity RMSE values are within 0.018-0.030 range indicating quite competitive results under unified model for most consumer entities. The MAE analysis similarly reveals two distinct tiers of entity-level predictability, with a primary cluster at 0.016–0.020 and a secondary cluster at 0.030–0.037, potentially reflecting differing behavioral regularity across consumer categories.

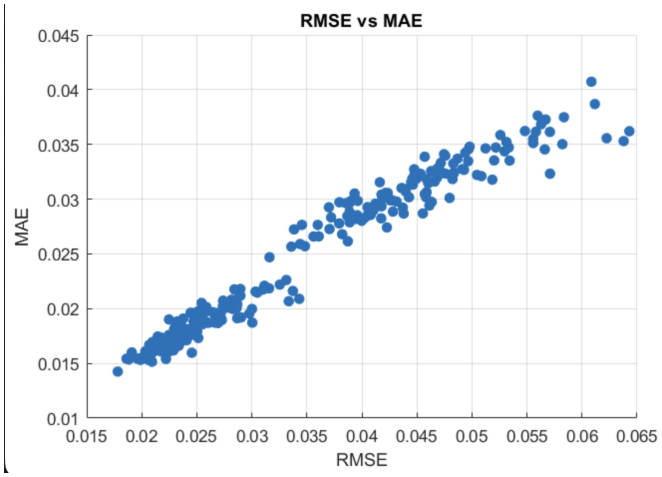

Fig. 8. RMSE versus MAE per-entity scatter plot. The strong linear correlation confirms consistent error scaling across all 250 consumer entities.

3) RMSE–MAE Correlation: The scatter plot RMSE versus MAE (Fig. 8) shows a good linear correlation for all 250 entities. The best entity has an RMSE of 0.017820, while the worst has an RMSE of 0.064303. The average across entities is 0.034768, hence there is variability in consumption regularity amongst the consumers.

C. Comparative Framework Analysis

Table III provides a side-by-side comparison of the two frameworks across key architectural and performance dimensions.

The two frameworks represent fundamentally complementary design philosophies. The LSTM-GRU model uses a 168-timestep lookback to capture full weekly periodicity, making

TABLE III
DUAL FRAMEWORK ARCHITECTURAL AND PERFORMANCE COMPARISON

Aspect	**LSTM-GRU (Python)**	**TCN-LSTM (MATLAB)**
Mechanism	Sequential Recurrent	Parallel Conv. + Recurrent
Lookback	168 steps (≈1 week)	60 steps (≈15 hrs)
Strength	Long-Range Dependencies	High-Freq. Pattern Extract.
Scaling	Entity-Specific Std.	Global Unified Std.
Entities	Single	250 (simultaneous)
Training	133 epochs	20 epochs (1253 s)
Test RMSE	0.0076 kWh (physical)	0.0348 avg (normalized)
R^2	0.9692	N/A (global aggregate)

it particularly effective for entities with strong day-of-week behavioral cycles. The TCN-LSTM model, operating with a 60-timestep window, leverages dilated causal convolutions to extract intra-day high-frequency patterns across all entities simultaneously, achieving competitive per-entity accuracy with a single unified model.

The bimodal MAE distribution in the TCN-LSTM results suggests that entities with irregular or aperiodic consumption profiles are less well-served by the global model's shared representation. The strong RMSE–MAE linearity across all 250 entities further confirms that neither framework exhibits pathological failure modes for specific consumer types.

To contextualize performance against the broader literature, the LSTM–GRU model's test RMSE of 0.0076 kWh and R^2 of 0.9692 compares favorably with recent hybrid architectures. Bao et al. [1] reported competitive short-term forecasting accuracy using a dual-attention BiLSTM, while Sajjad et al. [**?**] achieved strong results with a CNN–GRU hybrid. The MAPE of 5.78% achieved here is consistent with top-performing models in the literature [**?**], [2], indicating that hybrid architectures combining complementary processing mechanisms yield superior generalization compared to individual component models, whether the combination is recurrent–recurrent (LSTM–GRU) or convolutional–recurrent (TCN–LSTM).

The 1253-second training time for 250 entities with the global TCN–LSTM model demonstrates practical scalability, though real-time inference latency benchmarking remains an avenue for future study.

A limitation of the current study is that the LSTM–GRU framework was evaluated only under entity-specific standardisation, while the TCN–LSTM was evaluated only under the global scaling strategy. Cross-evaluating both scaling strategies on both architectures is reserved for future work, as it would require substantial retraining effort and is beyond the scope of this initial dual-framework investigation.

V. CONCLUSION AND FUTURE WORK

This paper presents dual hybrid frameworks validating multiple architectural approaches to electricity forecasting. LSTM-GRU achieves exceptional performance (96.92% explained variance, 5.78% MAPE) with excellent generalization (2.61% val-train gap). TCN-LSTM provides an alternative approach via a convolutional-temporal hybrid and global scaling.

Key findings validate the broader literature: hybrid superiority [**?**], [2], TCN efficacy [9], TCN-LSTM hybridization [10], global scaling [11], encoder-decoder relationships [12] are all confirmed by the experimental results.

Future work will address multi-step ahead forecasting horizons, real-time deployment with latency profiling, adversarial robustness testing, interpretability enhancement via attention visualisation, continual learning under distribution shift, ensemble combination of both frameworks, and transfer learning across geographic regions.

Both frameworks provide reliable forecasts enabling utilities to optimize operations, manage demand, allocate resources, and maintain grid stability.

REFERENCES

[1] G. Bao et al., "Short-term Electricity Load Forecasting Based on Dual Attention Mechanism and BiLSTM," 2023 8th International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 2023, pp. 229-233.

[2] A. Unlu, M. Peña and Z. Wang, "Comparison of the Combined Deep Learning Methods for Load Forecasting," 2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 2023, pp. 1-5.

[3] V. R and N. M. Jothi Swaroopan, "Contrast of Stacked GRU-RNN and Bidirectional LSTM network Based Process for Forecasting Renewable Energy for Smart Grid," 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE), Chennai, India, 2023, pp. 1-6.

[4] C. Luo et al., "Medium- and Long- Term Electricity Load Forecasting Based on ARIMA-GM-LSTM," 2023 IEEE 7th Conference on Energy Internet and Energy System Integration (EI2), Hangzhou, China, 2023, pp. 4318-4324.

[5] Y. Wang et al., "Research on Short-Term Electricity Load Forecasting Model Based on SWT-ResNet-LSTM," 2023 3rd International Conference on Energy, Power and Electrical Engineering (EPEE), Wuhan, China, 2023, pp. 1268-1272.

[6] S. Chen, R. Lin and W. Zeng, "Short-Term Load Forecasting Method Based on ARIMA and LSTM," 2022 IEEE 22nd International Conference on Communication Technology (ICCT), Nanjing, China, 2022, pp. 1913-1917.

[7] V. Podgorelec et al., "Long-term Electricity Consumption Forecasting: How Different Models Forecast into the Future," 2024 International Conference on Smart Systems and Technologies (SST), Osijek, Croatia, 2024, pp. 139-144.

[8] L. Wensheng et al., "A Region-Level Integrated Energy Load Forecasting Method Based on CNN-LSTM Model with User Energy Label Differentiation," 2020 5th International Conference on Power and Renewable Energy (ICPRE), Shanghai, China, 2020, pp. 154-159.

[9] P. Lara-Benítez, M. Carranza-García, J. M. Luna-Romera, and J. C. Riquelme, "Temporal Convolutional Networks Applied to Energy-Related Time Series Forecasting," *Applied Sciences*, vol. 10, no. 7, p. 2322, 2020, doi: 10.3390/app10072322.

[10] L. Heng, C. Hao, and L. C. Nan, "Load forecasting method based on CEEMDAN and TCN–LSTM," *PLOS ONE*, vol. 19, no. 7, p. e0300496, 2024, doi: 10.1371/journal.pone.0300496.

[11] M. Grabner, Y. Wang, Q. Wen, B. Blažič, and V. Štruc, "A Global Modeling Framework for Load Forecasting in Distribution Networks," *IEEE Transactions on Smart Grid*, vol. 14, no. 6, pp. 4927–4941, Nov. 2023, doi: 10.1109/TSG.2023.3264525.

[12] S. Du, T. Li, and S.-J. Horng, "Time Series Forecasting Using Sequence-to-Sequence Deep Learning Framework," in *Proc. 9th Int. Symp. Parallel Architectures, Algorithms and Programming (PAAP)*, Taipei, Taiwan, 2018, pp. 171–176, doi: 10.1109/PAAP.2018.00037.