Bulletin of Electrical Engineering and Informatics
Vol. 15, No. 3, June 2026, pp. 2734~2746
ISSN: 2302-9285, DOI: 10.11591/eei.v15i3.11030

Two-stage random search–Bayesian optimization for CNN-based short-term load forecasting

Tuan Anh Nguyen, Thanh Ngoc Tran
Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

Article Info

Article history:

Received Jul 15, 2025
Revised Apr 17, 2026
Accepted May 7, 2026

Keywords:

Bayesian optimization
Convolutional neural networks
Hyperparameter optimization
Random search
Short-term load forecasting

ABSTRACT

This study proposes a two-stage hyperparameter optimization pipeline for convolutional neural network (CNN)–based short-term electricity load forecasting. In the first stage, random search is used to broadly explore candidate configurations, including the number of filters in each convolutional layer, batch size, training epochs, and the loss function. In the second stage, Bayesian optimization based on the tree-structured Parzen estimator (TPE), implemented in Optuna, refines promising regions of the hyperparameter space to obtain a better-performing model. The optimized CNN is evaluated using half-hourly (30-minute) electricity demand data from New South Wales (NSW), Victoria (VIC), and Queensland (QLD), and is benchmarked against a baseline CNN, a multilayer perceptron (MLP), an extended short-term memory network, and single-stage optimization variants. Across the three regions, the proposed approach achieves mean absolute percentage error (MAPE) values between 1.05% and 1.14%, representing an improvement of approximately 58% over the baseline CNN. Statistical robustness is examined using paired Wilcoxon signed-rank tests with Holm–Bonferroni correction on per-timestamp errors. Overall, the results indicate that combining random search with Bayesian optimization improves CNN forecasting accuracy across the three studied regions and provides a transparent tuning framework for future replication.

Corresponding Author:

Thanh Ngoc Tran
Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City
Ho Chi Minh City, Vietnam
Email: tranthanhngoc@iuh.edu.vn

1. INTRODUCTION

Short-term load forecasting (STLF) is critical in modern power system operation and planning. Accurate predictions of near-future electricity demand help optimize generation schedules, reduce operational costs, lower reserve capacity requirements, and enhance the overall reliability of the grid. Traditional forecasting methods, such as linear regression, moving averages, support vector regression (SVR) [1], [2], and ARIMA models [3], [4], have been widely used due to their simplicity and ease of implementation. However, these models face limitations in capturing nonlinear patterns and short-term fluctuations commonly observed in half-hourly (30-minute) electricity load data. With the advancement of deep learning, models such as multilayer perceptron (MLP) [5]–[7], long short-term memory (LSTM) [8]–[10], and convolutional neural network (CNN) [11]–[13] have demonstrated outstanding potential in time series forecasting tasks. Among them, CNNs stand out for their ability to effectively extract local features from sequential data. Nonetheless, the performance of deep learning models heavily depends on selecting suitable hyperparameters, such as the number of filters, loss functions, batch size, and the number of training epochs. To improve model accuracy, a common approach is to apply single-stage hyperparameter optimization, in which a search algorithm, such as grid search [14], [15], random search [16], [17], Bayesian optimization [18]–[20] or genetic algorithm (GA) [21]–[25], is directly applied to the whole hyperparameter space to identify the best configuration. While this strategy can be effective, it faces challenges in high-dimensional spaces, including high computational cost, the risk of getting stuck in local optima, or missing optimal regions due to insufficient search flexibility.

Despite these advances, several limitations remain unaddressed in existing research. First, single-stage optimization strategies often suffer from an imbalance between exploration and exploitation, particularly in CNN-based STLF applications. Second, prior studies have rarely conducted comparative evaluations across multiple regional datasets to validate model generalizability. Third, the motivation for selecting CNN over other deep learning models, such as MLP and LSTM, is often underexplored, despite their fundamental differences in architecture and performance behavior. In response to these issues, this study proposes a novel two-stage hyperparameter optimization strategy specifically designed to enhance the performance of CNNs in STLF. The necessity of including CNN, MLP, and LSTM in this study lies in their distinct structures and prevalence in related literature, making the comparison both relevant and valuable.

This study proposes a two-stage hyperparameter optimization strategy for CNN-based STLF. Stage one employs random search for low-cost, broad exploration, while stage two utilizes Bayesian optimization (via Optuna) to fine-tune the model in promising regions [26]–[28]. The optimized CNN model (RANDOM_BO_CNN) is tested on real half-hourly load data from New South Wales (NSW), Victoria (VIC), and Queensland (QLD), and compared to baseline CNN, LSTM, and MLP models using mean absolute error (MAE), mean squared error (MSE), root mean squared error (RMSE), mean absolute percentage error (MAPE), and execution time. Results show improved accuracy and efficiency. Key contributions include the development of a hybrid optimization framework, the application of real-world datasets, and a thorough comparison of models. The main contributions of this study are summarized as follows:

– Proposes a two-stage hyperparameter optimization strategy combining random search and Bayesian optimization (TPE) to improve CNN performance for STLF systematically.
– Provides a comparative evaluation on three regional datasets (NSW, VIC, and QLD) to assess the consistency and transferability of the proposed model across the studied regions.
– Presents a detailed analysis of forecasting accuracy and computational cost trade-offs, offering practical insights for balancing performance and efficiency in deep learning model deployment.
– Supplies a reproducible implementation and parameter search framework, enabling researchers and practitioners to replicate and adapt the approach for other forecasting tasks.

2. RELATED WORK

Deep-learning models such as MLPs, LSTM networks, and especially CNN are widely used in STLF because they capture local temporal patterns in sequential data [29]–[31]. However, the accuracy of CNN is highly sensitive to hyperparameters (number of filters, loss function, batch size, and number of training epochs), which makes systematic tuning a necessity rather than an option [32]–[34]. Based on this need, numerous studies have employed hyperparameter optimization techniques to enhance model performance.

However, most prior hyperparameter-optimization approaches follow a single-stage pattern (grid search, random search, Bayesian optimization, or GA) [35], [36], which reveals limitations when the search space is high-dimensional: computational cost grows rapidly, searches can become trapped in local optima, and it is challenging to balance exploration and exploitation [37], [38]. Consequently, several gaps remain: i) the lack of a deliberate transition from exploration to exploitation when tuning convolutional-neural-network models for STLF; ii) limited cross-regional evaluation, leaving generalizability across different power systems underexplored; and iii) insufficient discussion of why CNN are preferable to MLPs or LSTM networks in specific short-term load-forecasting regimes. Therefore, a more structured hyperparameter-optimization procedure and a broader, region-aware experimental protocol are required.

Building on these limitations, a practical approach is needed. This design helps avoid poor regions early while leveraging the sample efficiency of Bayesian methods in the second stage. The present study employs a RANDOM_BO_CNN for CNN-based STLF. It validates it on three regional datasets: NSW, VIC, and QLD. Beyond aggregate metrics, the evaluation is performed per timestamp and supported by paired Wilcoxon signed-rank tests with the Holm–Bonferroni adjustment. The accuracy–computation trade-off is analyzed to inform deployment. This positions the two-stage approach as a practical hyperparameter-optimization solution that improves accuracy while preserving operational relevance.

3. PROPOSED METHOD

This study proposes a two-stage hyperparameter optimization strategy to enhance the performance of CNNs for STLF, following recent research that combines global exploration and local refinement in hyperparameter optimization [39], [40]. First, random search broadly explores the hyperparameter space with 3-fold cross-validation and negative MAPE as the objective, selecting the best configuration as a starting point. Next, Bayesian optimization (TPE via Optuna) refines the search around promising regions for greater efficiency and reliability. The CNN is tailored for one-dimensional time series, utilizing stacked Conv1D layers, Flatten, dense layers, and dropout to mitigate overfitting, with early stopping to prevent unnecessary epochs. Performance is evaluated using MAPE, MAE, RMSE, and MSE for accuracy and generalization. Three-fold cross-validation and negative MAPE consistently guide optimization toward minimal forecasting error.

3.1. Convolutional neural network model architecture

The proposed one-dimensional CNN processes time-series windows of size (48,1), which is consistent with common CNN-based architectures used in STLF studies [29]–[31]. The backbone architecture comprises four consecutive Conv1D layers with a kernel size of 3, ReLU activation, and "same" padding to preserve temporal length; the number of filters increases with depth (16, 32, 64, 128), expanding representational capacity and enabling progressively richer feature learning. After the convolutional block, the output is flattened and passed to a Dense layer with 64 units and ReLU to learn higher-level feature combinations, followed by a Dropout layer with a rate of 0.2 to reduce overfitting. The output layer uses a single linear unit for point forecasting of a continuous variable. The model is compiled with the Adam optimizer (learning rate=0.001), using MSE as the loss function and MAE as a monitoring metric; unless otherwise specified, training uses a batch size of 32, a 20% validation split, and early stopping (patience=20) to determine the effective number of epochs. This design leverages 1D convolutions with small kernels to emphasize short-range temporal relationships. Increasing filter counts expands representational power without incurring an excessive computational burden. A compact Dense head with dropout provides effective regularization, and Adam, with a conservative learning rate, yields stable convergence. Wall-clock training time is also recorded to support computational-efficiency analysis and experimental comparisons across configurations. To describe the signal processing and feature extraction mechanisms of the CNN model, the key equations are given (1)-(4):

$$S(i,j) = (F * I)(i,j) = \sum_m \sum_n F(m,n) I(i-m, j-n) \tag{1}$$

$$ReLU(x) = max(0,x) \tag{2}$$

$$\mathrm{P(i,j)} = \mathrm{max}_{\mathrm{k,l \in window}}\ \mathrm{I(i+k, j+l} \tag{3}$$

$$\mathrm{y} = \mathrm{W_x} + \mathrm{b} \tag{4}$$

3.2. Two-stage hyperparameter optimization strategy

A two-stage hyperparameter optimization strategy was developed to enhance the performance of the CNN model in STLF. In the first stage, random search was employed to perform a broad, global exploration of the hyperparameter space, quickly identifying regions with promising performance. To ensure statistical reliability, each randomly sampled configuration was evaluated using three-fold cross-validation and the negative MAPE criterion. The best-performing set of hyperparameters obtained from this stage served as the initial seed for the next phase.

In the second stage, bayesian optimization (bayesian optuna framework with the tree-structured Parzen estimator (TPE)) was applied to refine the search locally around the most promising regions discovered earlier. This study employed Optuna's TPE algorithm instead of conventional Gaussian process–based Bayesian optimization because TPE scales more efficiently in high-dimensional and discrete hyperparameter spaces, avoids assumptions about Gaussian distributions, and achieves faster convergence in deep learning applications. This stage used probabilistic modeling to balance exploration and exploitation, thereby converging efficiently toward the global optimum while avoiding redundant evaluations of poor configurations. The process was repeated using three-fold cross-validation and the negative MAPE objective function. After finding the optimal configuration, the final CNN model was trained on the complete training dataset with early stopping to prevent overfitting. Model performance was then evaluated using the MAE, MSE, RMSE, and MAPE metrics on the test dataset.

This two-stage strategy combines global exploration (random search) with local refinement (Bayesian optimization), effectively balancing search breadth and precision. The method avoids local minima by progressively narrowing the search from broad to focused regions. It reduces overall computational search cost while retaining high forecasting accuracy and reproducibility for CNN-based STLF applications.

3.3. Algorithm flowchart

Figure 1 presentation of the proposed two-stage hyperparameter optimization workflow for CNNs in STLF. The process starts with the historical electricity load time series (Y_1, Y_2, …, Y_n), followed by data preprocessing and windowing to clean, normalize, and generate fixed-length input–output sequences. The dataset is then split into training and testing sets (X_train, Y_train, X_test, Y_test) to ensure unbiased evaluation.

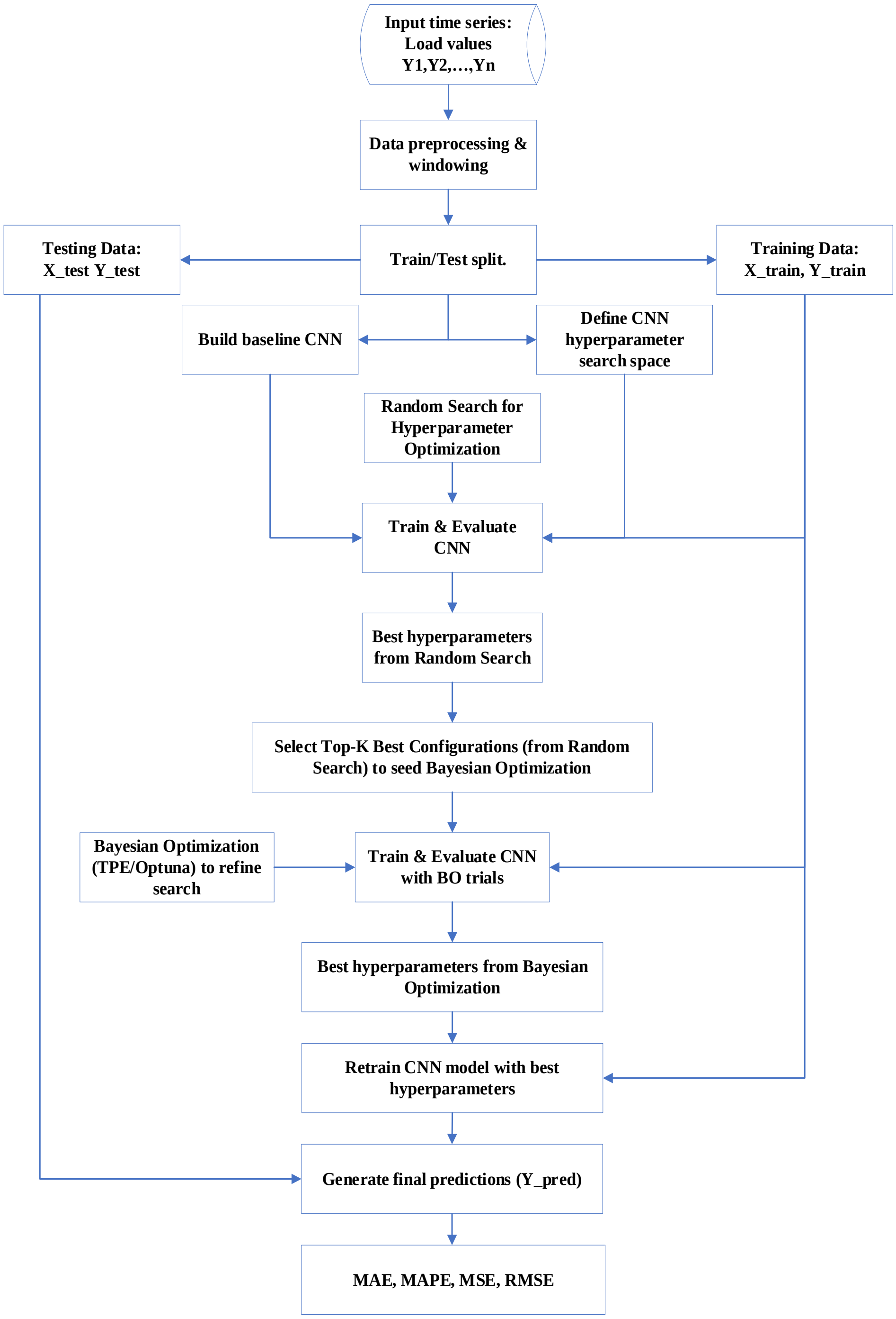


Figure 1. Workflow of the proposed two-stage hyperparameter optimization for CNN-based STLF

In the first stage (random search), a baseline CNN architecture and its hyperparameter search space (filters, batch size, training epochs, loss function, and learning rate) are defined. Random search broadly explores this space by training and evaluating multiple CNN configurations. The Top-K best configurations from random search are then selected and explicitly used to seed the Bayesian optimization stage, making the transition between global exploration and local refinement clear. In the second stage (Bayesian optimization), the TPE via Optuna refines the search around these promising regions. Each Bayesian trial proposes new hyperparameters guided by probabilistic modeling to balance exploration and exploitation. After convergence, the best hyperparameters from Bayesian optimization are used to retrain the CNN model from scratch with the complete training dataset. Finally, the optimized CNN produces predicted loads (Y_pred) for the test set, and performance is assessed using MAE, MAPE, MSE, and RMSE.

3.4. Pseudocode two-stage hyperparameter optimization for CNN in STLF

The following pseudocode summarizes the proposed two-stage hyperparameter optimization procedure for the CNN-based STLF model, including the random search stage and the subsequent Bayesian optimization refinement stage.

```
Input: Preprocessed electricity load dataset
Output: Optimized CNN model
1. Split the dataset into a training set and a testing
set
2. Define CNN hyperparameter search space:
  - Number of filters
  - Batch size
  - Number of epochs
  - Loss function
3. Stage 1: Random Search
  For i = 1 to N_trials (e.g., 100) do:
     - Randomly sample a hyperparameter
       configuration. Hi
     - Perform 3-fold cross-validation on the
       training set
     - Evaluate the model using negative MAPE
  End For
  Select the best configuration, H1, with the lowest
 MAPE
4. Stage 2: Bayesian Optimization (TPE using
Optuna)
  For j = 1 to M_trials (e.g., 100) do:
     - Suggest new configuration Hj near H1 (using
       TPE sampler)
     - Perform 3-fold cross-validation on the
       training set
     - Evaluate the model using negative MAPE
  End For
  Select the best configuration H2 with the lowest
     MAPE
5. Train the final CNN model using H2
  - Apply early stopping (patience = 20 epochs)
  - Monitor validation loss during training
6. Evaluate the final trained model on the test set
  - Calculate performance metrics: MAPE, MAE,
     RMSE, MSE
```

3.5. Experimental setup data

This subsection describes the dataset used to evaluate the proposed RANDOM_BO_CNN model. Real half-hourly electricity load data from NSW, VIC, and QLD were employed for model training and performance comparison. Figure 2 presents the half-hourly (30-minute) electricity load time series for three Australian regions: NSW, VIC, and QLD.

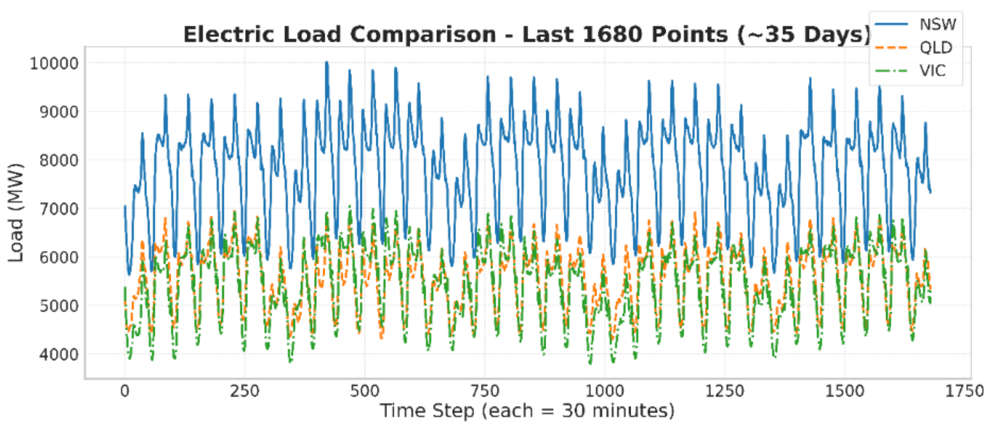


Figure 2. Electricity load profiles in NSW-VIC-QLD

The data were standardized (SETTLEMENTDATE), converted into one-dimensional arrays, and segmented into supervised samples using a 48-step sliding window (one day of input). Each sample consists of 48 input values, followed by the corresponding target output. The most recent 1,680 samples (≈35 days) were retained per region, split chronologically into 80/20 sets, and reshaped to (N, 48, 1) for Conv1D. This process ensures a fair comparison and enables the model to capture daily load patterns for effective short-term forecasting.

3.6. Hyperparameter grid

Table 1 summarizes the hyperparameters used to optimize the CNN model, including the number of filters, batch size, number of epochs, and loss function — key factors that directly affect learning ability and forecasting accuracy.

Table 1. Hyperparameters of the CNN model

Hyperparameter	Search range
Number of filters	{16, 32, 64, 96, 128}
Batch size	{16, 32, 64}
Number of epochs	Range: 50-501
Loss function	{MSE, MAE}
Learning_rate	0.001
Optimization algorithm (stage 1)	Random search
Optimization algorithm (stage 2)	BO-TPE

Filters with sizes {16, 32, 64, 96, 128} were selected to capture both fine local and broader features, based on insights from previous studies. Batch sizes {16, 32, 64} balance fast weight updates and training stability; small batches train quickly but are noisier, while larger ones are more stable but require more memory and time. Epochs: 50-501 controls learning depth: fewer reduce training time and overfitting, while more allow deeper feature learning and higher accuracy. Loss functions {MSE and MAE} were compared to evaluate sensitivity to significant errors MSE versus robustness to outliers MAE. A two-stage optimization strategy was applied: random search broadly explored the hyperparameter space to identify promising ranges, followed by Bayesian optimization (BO-TPE) for fine-tuning. This approach improves search efficiency, avoids local optima, and reduces computational cost. These choices were guided by prior research to balance predictive accuracy and computational efficiency.

3.7. Model evaluation metrics

To evaluate the forecasting accuracy, this study employs four widely used error metrics in STLF: MAE, MAPE, MSE, and RMSE, which are commonly used in previous load forecasting studies [41]–[43]. These metrics quantify absolute, relative, and squared deviations between actual and predicted values, allowing a comprehensive assessment of prediction accuracy and comparison among hyperparameter optimization methods. They are used to analyze the effectiveness of the proposed two-stage optimization strategy (random search→Bayesian optimization) for the CNN model and to benchmark against one-stage approaches, as detailed in the results and discussion section.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{5}$$

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i-\hat{y}_i}{y_i}\right| \tag{6}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{7}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{8}$$

4. RESULTS AND DISCUSSION

Table 2 summarizes the best results (MAE, MSE, RMSE, MAPE, and training time) for NSW, VIC, and QLD. RANDOM_BO_CNN achieves the lowest errors across all regions, with MAPEs of 1.05% (NSW), 1.14% (VIC), and 1.02% (QLD), and substantial reductions in MAE/RMSE compared to the baselines. Compared with plain CNN, MAPE drops by 58.4%/29.4%/27.1%; versus RANDOM_CNN by 52.9%/13.3%/25.1%; and versus BO_CNN by 45.8%/7.8%/12.4%, confirming the random→Bayesian synergy. This accuracy comes at the cost of much longer training times (≈31000 s NSW, 17000 s VIC, 16000 s QLD), while MLP trains the fastest (28–81 s) but is less accurate, and LSTM yields the most significant errors. Despite the heavier training, the two-stage optimization shows robust gains and remains practical with periodic offline retuning and fast inference. Figure 3 compares MAPE values across six forecasting models—MLP, LSTM, CNN, RANDOM_CNN, BO_CNN, and RANDOM_BO_CNN—for NSW, VIC, and QLD.

Table 2. Forecasting performance and the best-performing model on the NSW, VI, QLD dataset

Model	MAE	MSE	RMSE	MAPE%	Time (s)
		NSW			
MLP	245.0682	72888.22	269.9782	3.180223	28
LSTM	773.9056	1498554	1224.154	10.21842	1478
CNN	194.8472	47573.41	218.1133	2.529861	61
RANDOM_CNN	174.0563	42859.82	207.0261	2.232579	6691
BO_CNN	149.9385	30934.19	175.8812	1.940599	12516
RANDOM_BO_CNN	82.56039	12330.04	111.0407	1.052616	31336
		VIC			
MLP	97.7214	15582.2	124.8287	1.8105718	81
LSTM	523.0125	475575.7	689.62	9.685216	1344
CNN	89.6261	12509.21	111.8446	1.6192078	142
RANDOM_CNN	73.16432	8506.277	92.22948	1.3184468	8185
BO_CNN	66.89527	7754.058	88.05713	1.2406533	14740
RANDOM_BO_CNN	62.22491	6766.249	82.25721	1.1434628	17313
		QLD			
MLP	109.7563635	15675.11122	125.2002844	1.9584	65
LSTM	357.2004152	196055.5391	442.781593	6.3259	1600
CNN	80.23974109	9262.897341	96.24394704	1.4051	182
RANDOM_CNN	76.50749791	8136.363382	90.20179257	1.3689	9003
BO_CNN	65.85034819	6191.664483	78.68713035	1.1704	8628
RANDOM_BO_CNN	57.49292847	5378.321506	73.33704048	1.0249	16474

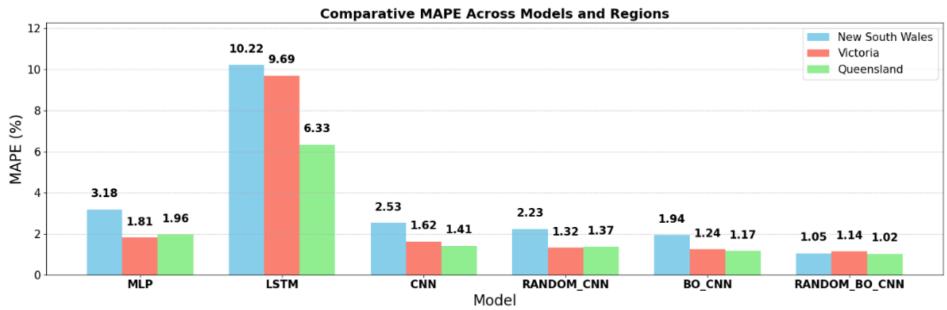


Figure 3. Comparative performance of deep learning models in regional load forecasting

The results show that RANDOM_BO_CNN consistently achieves the lowest MAPE (1.05–1.14%), confirming its superior forecasting accuracy. In contrast, LSTM exhibits the highest error (up to 10.22%), while CNN and its optimized variants demonstrate gradual improvement. The consistent performance ranking across regions indicates that the two-stage Random–Bayesian optimization effectively and robustly enhances the accuracy of the CNN model for STLF. Table 3 summarizes per-timestamp performance (mean±standard deviation) across the NSW, VIC, and QLD datasets.

In all regions, RANDOM_BO_CNN delivers the lowest MAPE_mean—1.05% (NSW), 1.14% (VIC), and 1.02% (QLD)—with N_points=336 for each dataset, and also reduces MAE_mean and RMSE_mean relative to baselines. Using MAPE_mean, the improvements over a plain CNN are approximately −58.5% (NSW), −29.6% (VIC), and −27.7% (QLD); against RANDOM_CNN they are −52.9%, −13.6%, and −25.5%; and versus BO_CNN they are −45.9%, −8.1%, and −12.8%, confirming the synergy of the random → Bayesian sequence. Regarding variability, MAPE_std also decreases versus CNN (NSW: −26.6%, VIC: −15.4%, and QLD: −10.0%), indicating improved robustness; the only nuance is QLD, where RANDOM_BO_CNN shows a slightly higher MAPE_std than BO_CNN (0.81 vs. 0.75), while still achieving the lowest MAPE_mean overall. LSTM yields the most significant errors in all three regions, and MLP trails CNN and the HPO variants. These results support the conclusion that the two-stage RANDOM_BO_CNN improves accuracy and reduces variability across diverse regional datasets.

Table 4 shows that the proposed RANDOM_BO_CNN consistently achieves the lowest MAPE across all three datasets, confirming the effectiveness of the two-stage optimization strategy. Overall performance and ΔMAPE gain of RANDOM_BO_CNN over single-stage optimization.

The relative improvement (ΔMAPE) compared with single-stage RANDOM_CNN and BO_CNN is substantial: NSW shows a reduction of approximately 52.9% versus RANDOM_CNN and 45.8% versus BO_CNN; VIC achieves 13.3% and 7.8% reductions, respectively; and QLD improves by 25.1% and 12.4%. These results demonstrate the synergy of broad exploration via random search followed by guided refinement with Bayesian optimization, leading to more accurate CNN configurations than either stage alone.

Table 3. Overall forecasting performance of models across the NSW-VIC-QLD datasets

Data	Model	MAPE mean	MAPE std	MAE mean	MAE std	RMSE mean	RMSE std	N points
NSW	MLP	3.18	0.53	245.07	46.33	266.38	47.45	336
NSW	LSTM	10.22	2.10	773.91	198.09	1191.06	305.36	336
NSW	CNN	2.53	0.42	194.85	28.09	216.96	24.21	336
NSW	RANDOM_CNN	2.23	0.38	174.06	33.07	204.60	34.10	336
NSW	BO_CNN	1.94	0.12	149.94	9.95	175.73	7.99	336
NSW	RANDOM_BO_CNN	1.05	0.28	82.56	22.97	108.51	25.45	336
QLD	MLP	1.96	0.25	109.76	12.41	124.50	14.33	336
QLD	LSTM	6.33	2.02	357.20	122.20	424.96	134.31	336
QLD	CNN	1.41	0.17	80.24	11.08	95.65	11.55	336
QLD	RANDOM_CNN	1.37	0.15	76.51	9.95	89.72	10.02	336
QLD	BO_CNN	1.17	0.08	65.85	4.81	78.59	4.29	336
QLD	RANDOM_BO_CNN	1.02	0.26	57.49	13.92	71.62	17.06	336
VIC	MLP	1.81	0.18	97.72	7.11	124.59	8.30	336
VIC	LSTM	9.69	2.12	523.01	102.71	673.30	161.07	336
VIC	CNN	1.62	0.17	89.63	10.42	111.54	8.95	336
VIC	RANDOM_CNN	1.32	0.10	73.16	2.79	92.01	6.93	336
VIC	BO_CNN	1.24	0.18	66.90	4.55	87.87	6.17	336
VIC	RANDOM_BO_CNN	1.14	0.20	62.22	5.98	81.94	7.83	336

Table 4. Overall performance and ΔMAPE gain of RANDOM_BO_CNN over single-stage optimization

Dataset	RANDOM_CNN (MAPE %)	BO_CNN (MAPE %)	RANDOM_BO_CNN (MAPE %)	Δ% vs RANDOM_CNN (%)	Δ% vs BO_CNN (%)
NSW	2.23	1.94	1.05	52.9	45.8
VIC	1.32	1.24	1.14	13.3	7.8
QLD	1.37	1.17	1.02	25.1	12.4

Figure 4 presents a MAPE (%) comparison for six models on three regional datasets: NSW (Figure 4(a)), VIC (Figure 4(b)), and QLD (Figure 4(c)). The results are reported as mean±standard deviation at each timestamp (N=336; error bars=±1 standard deviation). Across all three regions, RANDOM_BO_CNN consistently achieves the lowest MAPE—1.05% (NSW), 1.14% (VIC), and 1.02% (QLD)—with short error bars indicating strong robustness. These results demonstrate the effectiveness of the proposed two-stage hyperparameter optimization strategy for improving CNN-based STLF accuracy. The error decreases monotonically along the sequence CNN→RANDOM_CNN→BO_CNN→RANDOM_BO_CNN, confirming the synergy between random search (broad exploration) and Bayesian optimization/TPE (guided refinement). In contrast, MLP is simpler but less accurate than the HPO-enhanced CNN variants, while LSTM is consistently the worst performer, with large means and dispersions (10.22%; 9.69%; 6.33% for NSW, VIC, QLD). These differences are statistically significant, as indicated by paired Wilcoxon tests ($p < 0.05$), suggesting that the two-stage pipeline improves both accuracy and stability, making it suitable for periodic offline retuning followed by rapid inference in deployment.

To confirm that the observed accuracy differences are statistically significant, we applied the Wilcoxon signed-rank test on per-timestamp MAPE, comparing the proposed RANDOM_BO_CNN with RANDOM_CNN and BO_CNN. Given the hypothesis that our model yields lower errors, a one-sided test (alternative="less") was used; the non-parametric Wilcoxon test is suitable for the non-normal error distribution. The Holm–Bonferroni adjustment was applied to control Type I error across the paired comparisons.

Table 5 shows that, for NSW, VIC, and QLD, both comparisons yield adjusted p-values <0.05, confirming the proposed model's improvements are statistically significant. For operating-regime analysis, load was split by quantiles (bottom 10%=trough, middle 80%=middle, top 10%=peak). Forecasting errors increase slightly at peaks: +0.55 percentage points in NSW, +0.14 in VIC, while QLD shows no increase and even −0.27. Despite this, RANDOM_BO_CNN consistently achieves the lowest relative error. These findings, detailed in the Discussion, highlight limitations under extreme load and suggest remedies such as adding exogenous features, extending data for spikes, or exploring hybrid architectures.

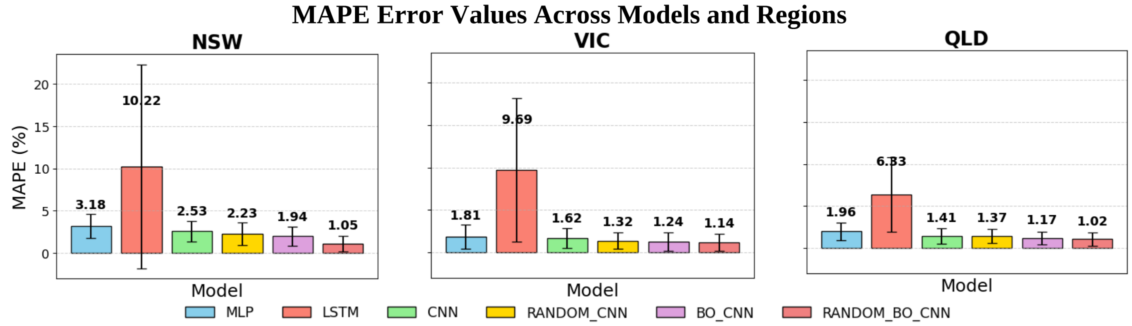


Figure 4. MAPE error values across models and regions: (a) forecasting performance on the NSW dataset, (b) forecasting performance on the VIC dataset, and (c) forecasting performance on the QLD dataset

Table 5. Wilcoxon tests on MAPE with Holm–Bonferroni adjustment

Dataset	(Timestamps)	RANDOM_BO_CNN < RANDOM_CNN (P_Adjusted)	RANDOM_BO_CNN < BO_CNN (P_Adjusted)
NSW	336	<0.05	<0.05
VIC	336	<0.05	<0.05
QLD	336	<0.05	<0.05

Figure 5 illustrates the training times of various forecasting models on datasets from NSW, VIC, and QLD. Specifically, Figure 5(a) presents the results for NSW, Figure 5(b) for VIC, and Figure 5(c) for QLD. Among the models, the MLP model has the shortest training time (ranging from 28 to 81 seconds), followed by the CNN model, which requires less than 200 seconds for training.

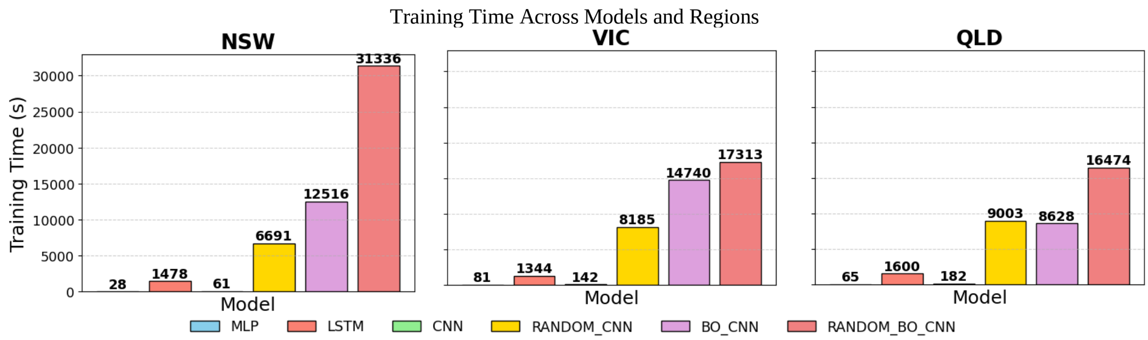
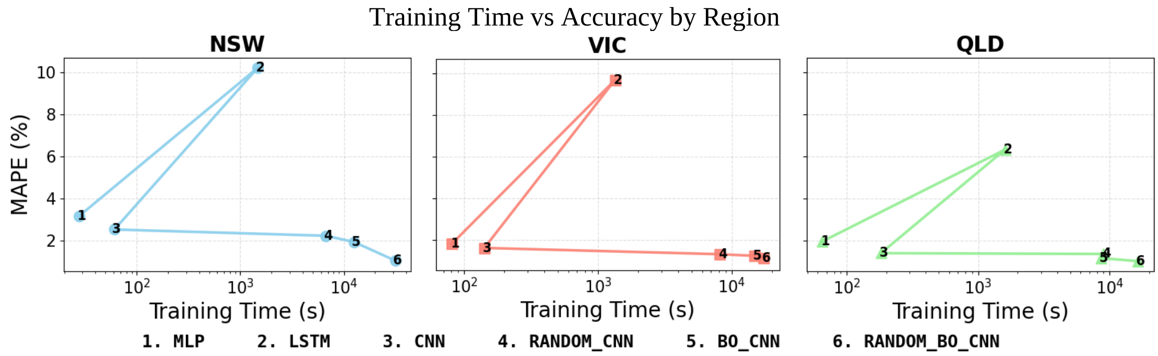


Figure 5. Training time across models and regions: (a) training time on the NSW dataset, (b) training time on the VIC dataset, and (c) training time on the QLD dataset

Due to its sequential architecture, the LSTM model took significantly longer (1,344–1,601 seconds). Models incorporating the two-stage optimization strategy—BO_CNN, RANDOM_CNN, and RANDOM_BO_CNN—required substantially more time, with RANDOM_BO_CNN reaching up to 31,336 seconds in the NSW dataset. These results highlight the trade-off between achieving higher accuracy and increasing training costs. Despite the increased training time, the RANDOM_BO_CNN model consistently outperformed the baseline models in forecasting accuracy across all regional datasets. The experimental findings confirm the effectiveness of the two-stage hyperparameter optimization strategy in enhancing model performance. Although it demands more computational resources, this trade-off is justified in applications where forecasting accuracy is paramount.

Figure 6 illustrates the relationship between training time and forecasting accuracy (MAPE) for six models—MLP, LSTM, CNN, RANDOM_CNN, BO_CNN, and RANDOM_BO_CNN—across three regions. Specifically, Figure 6(a) presents the results for NSW, Figure 6(b) for VIC, and Figure 6(c) for QLD.

The results indicate that optimized models, particularly RANDOM_BO_CNN, achieve significantly higher accuracy despite longer training times. In contrast, LSTM exhibits the longest training duration yet the highest error, suggesting limited suitability for short-term forecasting. This figure illustrates the trade-off between computational cost and forecasting accuracy, showing that RANDOM_BO_CNN offers the best balance, making it the most practical model for real-world load forecasting applications.



(a) (b) (c)

Figure 6. Relationship between computational cost and forecasting accuracy across regions: (a) results on the NSW dataset, (b) results on the VIC dataset, and (c) results on the QLD dataset

Error analysis indicates that the forecasting performance of the proposed model varies across different load regimes. The model exhibits slightly higher deviations during peak load hours, when demand fluctuates abruptly, resulting in temporary increases in MAPE and RMSE values. In contrast, prediction accuracy remains stable during off-peak and moderate-load periods, demonstrating the robustness of the two-stage optimized CNN in capturing dominant daily and seasonal patterns.

Despite the substantial improvement in forecasting accuracy, the model still presents certain limitations. The two-stage optimization process requires a relatively long training time due to the sequential random and Bayesian search phases. However, once trained, the inference process operates rapidly, allowing the model to be effectively applied in real-time operations or periodic retraining scenarios. This trade-off between computational cost and performance accuracy highlights the balance between model complexity and deployment feasibility in practical grid forecasting environments.

5. CONCLUSION

This study presents a two-stage hyperparameter optimization strategy for CNN-based STLF, combining broad exploration through random search with targeted refinement using Bayesian optimization (Optuna). The proposed RANDOM_BO_CNN model improves forecasting accuracy across the three studied regions while preserving a relatively simple CNN architecture. Experiments on half-hourly load data from NSW, VIC, and QLD show that RANDOM_BO_CNN achieves the lowest MAPE values (1.05%, 1.14%, and 1.02%) among the evaluated models, outperforming the conventional CNN, MLP, and LSTM baselines. However, the improved accuracy comes at the cost of longer training time, highlighting the trade-off between predictive performance and computational efficiency.

Future work may investigate multi-objective optimization to better balance accuracy and computational cost, incorporate exogenous variables such as weather and holidays, and explore hybrid architectures such as CNN–LSTM or CNN–Transformer for more complex forecasting tasks. It should also be noted that the present findings are based on three regional datasets within the same electricity-market context; therefore, broader validation under more diverse operating conditions is needed. Owing to its relatively lightweight architecture and fast inference, the proposed framework may have potential for future application in practical load-forecasting workflows, subject to further deployment-oriented validation.

ACKNOWLEDGMENTS

This research received no specific research grant or contract from any funding agency in the public, commercial, or not-for-profit sectors. The authors would like to thank colleagues at the Industrial University of Ho Chi Minh City for their valuable discussions and support.

FUNDING INFORMATION

This research received no specific grant or contract from any funding agency in the public, commercial, or not-for-profit sectors.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Tuan Anh Nguyen	✓	✓	✓	✓	✓	✓	✓		✓	✓		✓	✓	
Thanh Ngoc Tran	✓	✓			✓	✓		✓	✓	✓	✓	✓		✓

C : **C**onceptualization	I : **I**nvestigation	Vi : **Vi**sualization
M : **M**ethodology	R : **R**esources	Su : **Su**pervision
So : **So**ftware	D : **D**ata Curation	P : **P**roject administration
Va : **Va**lidation	O : Writing - **O**riginal Draft	Fu : **Fu**nding acquisition
Fo : **Fo**rmal analysis	E : Writing - Review & **E**diting	

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no conflict of interest.

INFORMED CONSENT

Informed consent was not applicable to this study.

ETHICAL APPROVAL

Ethical approval was not required for this study because it did not involve human participants or animals.

DATA AVAILABILITY

The data supporting the findings of this study are publicly available from the Australian Energy Market Operator (AEMO).

REFERENCES

[1] D. Tao, S. Fang, H. Liu, J. Lu, J. Wang, and Q. Ma, "Support vector regression model for the prediction of buildings' maximum seismic response based on real monitoring data," *Scientific Reports*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-81705-3.

[2] M. Safari, A. H. Rabiee, and J. Joudaki, "Developing a support vector regression (SVR) model for prediction of main and lateral bending angles in laser tube bending process," *Materials*, vol. 16, no. 8, Apr. 2023, doi: 10.3390/ma16083251.

[3] Z. Hamad and I. Abdulrahman, "Deep learning-based load forecasting considering data reshaping using MATLAB\Simulink," *International Journal of Energy and Environmental Engineering*, vol. 13, no. 2, pp. 853–869, Jun. 2022, doi: 10.1007/s40095-022-00480-x.

[4] Y. Huang, Y. Zhao, Z. Wang, X. Liu, and Y. Fu, "Sparse dynamic graph learning for district heat load forecasting," *Applied Energy*, vol. 371, Oct. 2024, doi: 10.1016/j.apenergy.2024.123685.

[5] Z. Tang, T. Ji, J. Kang, Y. Huang, and W. Tang, "Learning global and local features of power load series through transformer and 2D-CNN: An image-based multi-step forecasting approach incorporating phase space reconstruction," *Applied Energy*, vol. 378, Jan. 2025, doi: 10.1016/j.apenergy.2024.124786.

[6] A. Saravanakumar, L. Rajeshkumar, D. Balaji, and M. P. J. Karunan, "Prediction of wear characteristics of AA2219-Gr matrix composites using GRNN and taguchi - based approach," *Arabian Journal for Science and Engineering*, vol. 45, no. 11, pp. 9549–9557, Nov. 2020, doi: 10.1007/s13369-020-04817-8.

[7] W. Lin, D. Wu, and M. Jenkin, "Electric load forecasting for individual households via spatial-temporal knowledge distillation," *IEEE Transactions on Power Systems*, vol. 40, no. 1, pp. 572–584, Jan. 2025, doi: 10.1109/TPWRS.2024.3393926.

[8] H. Kuang, Q. Guo, S. Li, and H. Zhong, "Short-term power load forecasting method in rural areas based on CNN-LSTM," in *2021 IEEE 4th International Electrical and Energy Conference (CIEEC)*, May 2021, pp. 1–5, doi: 10.1109/CIEEC50170.2021.9510777.

[9] J. Wen, Y. Peng, W. Zhang, X. Huang, and Z. Wang, "Short-term power load forecasting based on TCN-LSTM model," in *2024 IEEE 6th International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*, Oct. 2024, pp. 734–738, doi: 10.1109/ICCASIT62299.2024.10827868.

[10] S. Wu *et al.*, "Power load forecasting method based on random matrix theory and CNN-LSTM model," in *2022 IEEE 2nd International Conference on Digital Twins and Parallel Intelligence (DTPI)*, Oct. 2022, pp. 1–6, doi: 10.1109/DTPI55838.2022.9998910.

[11] F. Y. Li and J. F. Xiao, "Multi-channel LSTM-CNN power load multi-step prediction based on feature fusion," in *2022 5th World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM)*, Nov. 2022, pp. 591–594, doi: 10.1109/WCMEIM56910.2022.10021531.

[12] H. Hua, M. Liu, Y. Li, S. Deng, and Q. Wang, "An ensemble framework for short-term load forecasting based on parallel CNN and GRU with improved ResNet," *Electric Power Systems Research*, vol. 216, Mar. 2023, doi: 10.1016/j.epsr.2022.109057.

[13] S. Bian, Z. Wang, W. Song, and X. Zhou, "Feature extraction and classification of time-varying power load characteristics based on PCANet and CNN+Bi-LSTM algorithms," *Electric Power Systems Research*, vol. 217, Apr. 2023, doi: 10.1016/j.epsr.2023.109149.

[14] A. M. N. C. Ribeiro, P. R. X. do Carmo, P. T. Endo, P. Rosati, and T. Lynn, "Short- and very short-term firm-level load forecasting for warehouses: A comparison of machine learning and deep learning models," *Energies*, vol. 15, no. 3, 2022, doi: 10.3390/en15030750.

[15] B. V. S. Vardhan, M. Khedkar, I. Srivastava, P. Thakre, and N. D. Bokde, "A comparative analysis of hyperparameter tuned stochastic short term load forecasting for power system operator," *Energies*, vol. 16, no. 3, 2023, doi: 10.3390/en16031243.

[16] Z. Cahya, D. Cahya, T. Nugroho, A. Zuhri, and W. Agusta, "CNN model with parameter optimisation for fine-grained banana ripening stage classification," in *Proceedings of the 2022 International Conference on Computer, Control, Informatics and Its Applications*, Nov. 2022, pp. 90–94, doi: 10.1145/3575882.3575900.

[17] F. Liu and C. Liang, "Short-term power load forecasting based on AC-BiLSTM model," *Energy Reports*, vol. 11, pp. 1570–1579, Jun. 2024, doi: 10.1016/j.egyr.2024.01.026.

[18] M. Munem, T. M. R. Bashar, M. H. Roni, M. Shahriar, T. B. Shawkat, and H. Rahaman, "Electric power load forecasting based on multivariate LSTM neural network using bayesian optimization," in *2020 IEEE Electric Power and Energy Conference (EPEC)*, Nov. 2020, pp. 1–6, doi: 10.1109/EPEC48502.2020.9320123.

[19] M. Vijay and M. Saravanan, "Solar irradiance forecasting using bayesian optimization based machine learning algorithm to determine the optimal size of a residential PV system," in *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, Apr. 2022, pp. 744–749, doi: 10.1109/ICSCDS53736.2022.9761011.

[20] M. A. Iqbal, J.-M. Gil, and S. K. Kim, "Attention-driven hybrid ensemble approach with bayesian optimization for accurate energy forecasting in Jeju Island's renewable energy system," *IEEE Access*, vol. 13, pp. 7986–8010, 2025, doi: 10.1109/ACCESS.2025.3526943.

[21] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.

[22] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid CNN-LSTM model for short-term individual household load forecasting," *IEEE Access*, vol. 8, pp. 180544–180557, 2020, doi: 10.1109/ACCESS.2020.3028281.

[23] Y. Xie *et al.*, "Short-term load forecasting method based on fuzzy optimization combined model of load feature recognition," *Electrical Engineering*, vol. 107, no. 1, pp. 513–526, Jan. 2025, doi: 10.1007/s00202-024-02539-w.

[24] Z. Liao, H. Pan, X. Fan, Y. Zhang, and L. Kuang, "Multiple wavelet convolutional neural network for short-term load forecasting," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9730–9739, Jun. 2021, doi: 10.1109/JIOT.2020.3026733.

[25] M. Sakib, T. Siddiqui, S. Mustajab, R. M. Alotaibi, N. M. Alshareef, and M. Z. Khan, "An ensemble deep learning framework for energy demand forecasting using genetic algorithm-based feature selection," *PLoS One*, vol. 20, no. 1, pp. 1–28, 2025, doi: 10.1371/journal.pone.0310465.

[26] C. Xu and G. Chen, "Interpretable transformer-based model for probabilistic short-term forecasting of residential net load," *International Journal of Electrical Power & Energy Systems*, vol. 155, Jan. 2024, doi: 10.1016/j.ijepes.2023.109515.

[27] W. Qian, C. Gu, C. Zhu, Z. Jiang, B. Han, and M. Yu, "Short-term load forecasting based on multi-model fusion of CNN-LSTM-LGBM," in *2021 International Conference on Power System Technology (POWERCON)*, Dec. 2021, pp. 934–939, doi: 10.1109/POWERCON53785.2021.9697619.

[28] T. A. Nguyen and T. N. Tran, "Improving short-term electrical load forecasting with dilated convolutional neural networks: A comparative analysis," *Journal of Robotics and Control (JRC)*, vol. 6, no. 2, pp. 560–569, Mar. 2025, doi: 10.18196/jrc.v6i2.24967.

[29] C. Wang, X. Li, Y. Shi, W. Jiang, Q. Song, and X. Li, "Load forecasting method based on CNN and extended LSTM," *Energy Reports*, vol. 12, pp. 2452–2461, Dec. 2024, doi: 10.1016/j.egyr.2024.07.030.

[30] Q. Hua *et al.*, "A short-term power load forecasting method using CNN-GRU with an attention mechanism," *Energies*, vol. 18, no. 1, Dec. 2024, doi: 10.3390/en18010106.

[31] Y. Eren and İ. Küçükdemiral, "A comprehensive review on deep learning approaches for short-term load forecasting," *Renewable and Sustainable Energy Reviews*, vol. 189, Jan. 2024, doi: 10.1016/j.rser.2023.114031.

[32] M. Zulfiqar, K. A. A. Gamage, M. Kamran, and M. B. Rasheed, "Hyperparameter optimization of bayesian neural network using bayesian optimization and intelligent feature engineering for load forecasting," *Sensors*, vol. 22, no. 12, Jun. 2022, doi: 10.3390/s22124446.

[33] P. S. Pravin, J. Z. M. Tan, K. S. Yap, and Z. Wu, "Hyperparameter optimization strategies for machine learning-based stochastic energy efficient scheduling in cyber-physical production systems," *Digital Chemical Engineering*, vol. 4, Sep. 2022, doi: 10.1016/j.dche.2022.100047.

[34] S. Hanifi, A. Cammarono, and H. Zare-Behtash, "Advanced hyperparameter optimization of deep learning models for wind power prediction," *Renewable Energy*, vol. 221, Feb. 2024, doi: 10.1016/j.renene.2023.119700.

[35] C. M. Liyew, E. Di Nardo, S. Ferraris, and R. Meo, "Hyperparameter optimization of machine learning models for predicting actual evapotranspiration," *Machine Learning with Applications*, vol. 20, Jun. 2025, doi: 10.1016/j.mlwa.2025.100661.

[36] B. Bischl *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *WIREs Data Mining and Knowledge Discovery*, vol. 13, no. 2, Mar. 2023, doi: 10.1002/widm.1484.

[37] M. A. K. Raiaan *et al.*, "A systematic review of hyperparameter optimization techniques in convolutional neural networks," *Decision Analytics Journal*, vol. 11, Jun. 2024, doi: 10.1016/j.dajour.2024.100470.

[38] H. Hassani, S. Nikan, and A. Shami, "Improved exploration–exploitation trade-off through adaptive prioritized experience replay," *Neurocomputing*, vol. 614, Jan. 2025, doi: 10.1016/j.neucom.2024.128836.

[39] J. Oh, D. So, J. Jo, N. Kang, E. Hwang, and J. Moon, "Two-stage neural network optimization for robust solar photovoltaic forecasting," *Electronics*, vol. 13, no. 9, Apr. 2024, doi: 10.3390/electronics13091659.

[40] M. Madhiarasan, "Bayesian optimisation algorithm based optimised deep bidirectional long short term memory for global horizontal irradiance prediction in long-term horizon," *Frontiers in Energy Research*, vol. 13, Feb. 2025, doi: 10.3389/fenrg.2025.1499751.

[41] Y. Xu, J. Li, Y. Dong, and P. Du, "Power load combination forecasting system based on longitudinal data selection," *Engineering Applications of Artificial Intelligence*, vol. 130, Apr. 2024, doi: 10.1016/j.engappai.2023.107629.

[42] F. Li, J. Ma, X. Yang, and W. Deng, "A short-term forecasting for multi-factor time series with multiple linear trend fuzzy information granule and cross-association," *Engineering Applications of Artificial Intelligence*, vol. 137, Nov. 2024, doi: 10.1016/j.engappai.2024.109232.

[43] Z. Zhang, Q. Zhang, H. Liang, and B. Gorbani, "Optimizing electric load forecasting with support vector regression/LSTM optimized by flexible Gorilla troops algorithm and neural networks a case study," *Scientific Reports*, vol. 14, no. 1, Sep. 2024, doi: 10.1038/s41598-024-73893-9.

BIOGRAPHIES OF AUTHORS

Tuan Anh Nguyen received a B.Sc. degree from the Industrial University of Ho Chi Minh City, Vietnam 2012, and an M.Sc. degree from the HCMC University of Technology and Education, Vietnam, in 2015. He is a Lecturer with the Faculty of Electrical Engineering Technology at the Industrial University of Ho Chi Minh City, Vietnam. His main research interests include power quality and load forecasting. He can be contacted at email: nguyenanhtuan@iuh.edu.vn.

Thanh Ngoc Tran received the B.Sc. and M.Sc. degrees from the Ho Chi Minh University of Technology, Vietnam, in 2001 and 2006, respectively. He received his Ph.D. in information and electrical engineering degree from Hunan University, Changsha, China, in 2012. He is a Lecturer with the Faculty of Electrical Engineering Technology at the Industrial University of Ho Chi Minh City, Vietnam. His main research interests include HVDC transmission systems, active filters, power quality, and load forecasting. He can be contacted at email: tranthanhngoc@iuh.edu.vn.