

Received 14 August 2025, accepted 4 September 2025,
date of publication 8 September 2025, date of current version 15 September 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3607330

RESEARCH ARTICLE

Optimizing Deep Residual Networks for Short-Term Load Forecasting With Multidimensional Weather Data and Principal Component Analysis

JUNCHEN LIU¹, FAISUL ARIF AHMAD¹, (Member, IEEE), KHAIRULMIZAM SAMSUDIN¹,
FAZIRULHISYAM HASHIM¹, (Member, IEEE),
AND MOHD ZAINAL ABIDIN AB KADIR², (Senior Member, IEEE)

¹Department of Computer and Communication Systems Engineering, Faculty of Engineering, Universiti Putra Malaysia (UPM), Serdang, Selangor 43400, Malaysia

²Advanced Lightning, Power and Energy Research Centre (ALPER), Faculty of Engineering, Universiti Putra Malaysia (UPM), Serdang, Selangor 43400, Malaysia

Corresponding author: Faisul Arif Ahmad (faisul@upm.edu.my)

ABSTRACT Short-term load forecasting (STLF) is essential for power system operations, supporting efficient grid management and resource planning. Deep Residual Networks (DRNs) have emerged as a promising architecture for STLF, offering a balanced solution by combining training stability, deep feature extraction, and reduced gradient degradation compared to Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformers. This study focuses on optimizing DRN for STLF by evaluating different combinations of multidimensional weather variables and applying Principal Component Analysis (PCA) to address feature complexity. Using the Malaysia dataset, which includes historical load, time, and weather variables such as temperature, rainfall, and wind speed, the impact of different variable combinations on forecasting precision is evaluated. Experimental results show that the DRN model outperforms baseline models including CNNs, RNN-based models, and Transformers, achieving a Mean Absolute Percentage Error (MAPE) of 0.052514 and a coefficient of determination (R^2) of 0.927993. Building upon this, the proposed PCA-DRN further improves forecasting performance, achieving a MAPE of 0.049994 and an R^2 of 0.934473, representing a 4.80% reduction in MAPE and a 0.65% increase in R^2 compared to the original DRN. These findings emphasize the importance of feature selection and dimensionality reduction in optimizing STLF models, particularly for tropical regions with relatively stable weather patterns.

INDEX TERMS DNN, DRN, PCA, STLF.

I. INTRODUCTION

In modern power systems, load forecasting (LF) is a critical tool for optimizing grid operations and ensuring reliability. It involves predicting future electricity demand, supporting power companies in planning, operating, and managing the grid more effectively. Accurate LF enhances energy efficiency, lowers operational expenses, and ensures

supply stability. As electricity demand rises and consumption patterns diversify, LF becomes increasingly essential and complex [1].

LF is typically categorized into four types: Very Short-Term Load Forecasting (VSTLF), Short-Term Load Forecasting (STLF), Medium-Term Load Forecasting (MTLF), and Long-Term Load Forecasting (LTLF), as illustrated in Figure 1 and distinguished by the forecasting horizon [2]. Among them, STLF generally refers to predicting electricity demand from one hour up to one week

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang¹.

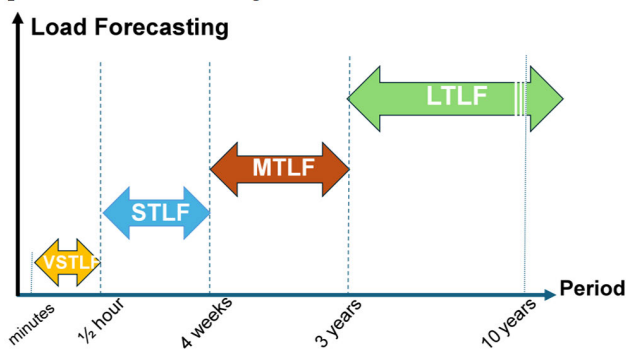


FIGURE 1. Classification of load forecasting.

ahead, and plays a crucial role in day-to-day power system operations and scheduling.

The evolving demands of power system operations require faster decision-making and better handling of uncertainties. LF serves various applications, including energy trading, system security analysis, unit commitment, economic power generation, and performance monitoring. As forecasting precision directly impacts grid operations, improving the precision of predictions has become increasingly critical. Inaccurate forecasts can lead to unexpected costs, underscoring the importance of reliable LF for daily operations and load flow analysis [3], [4].

Numerous methods have been proposed to address STLF challenges, broadly divided into traditional and modern approaches. Traditional methods, such as linear [5] or non-parametric methods [6], [7], [8], support vector regression (SVR) [9], [10], [11], autoregressive models [12], and fuzzy logic [13], [14], often struggle with limitations like oversimplification, difficulty in modeling complex load patterns, and susceptibility to overfitting as input variables increase [10], [15].

To overcome these shortcomings, modern techniques, particularly artificial neural networks (ANNs), have emerged as a popular choice for STLF systems. By leveraging deep learning, ANN-based methods capture intricate load patterns, enhance prediction precision, and mitigate overfitting risks [16], [17]. However, increasing the number of inputs, hidden nodes, or layers can still make these networks prone to overfitting [20]. To address this, advanced ANN variants such as radial basis function (RBF) networks [18], wavelet-based networks [19], and extreme learning machines (ELM) [20] have been developed for STLF applications.

In recent years, deep neural networks (DNNs), characterized by their multiple hidden layers, have gained prominence in LF. By employing hierarchical feature representation, DNNs effectively model complex load dynamics. Research trends in LF have shifted from traditional shallow networks to specialized deep learning architectures that integrate diverse data sources to capture intricate temporal and spatial dependencies. These advancements, driven by the proven efficacy of deep learning, highlight the field's transition toward more sophisticated approaches [21], [22], [23]. Figure 2 presents a

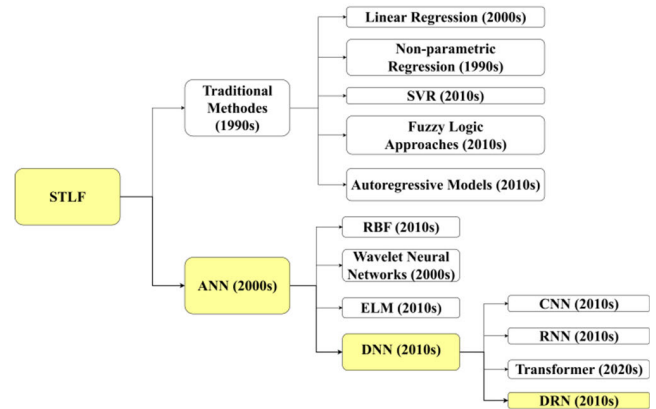


FIGURE 2. Technical Roadmap of STLF [2].

technical roadmap of STLF methodologies, tracing the progression from traditional statistical approaches to advanced deep neural networks (DNNs). According to a review of articles indexed in the Web of Science, traditional statistical methods in STLF began to emerge in the 1990s, while the application of artificial intelligence techniques started to gain prominence in the 2000s. The roadmap highlights key developments and transitions in load forecasting research over time [2].

Recent studies have moved beyond shallow networks, incorporating diverse data into network designs to enhance forecasting precision. Convolutional Neural Networks (CNNs) [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], well-suited for extracting local features, have demonstrated strong performance in processing temporal patterns in load data. However, their inability to capture long-term dependencies and challenges with vanishing gradients in deeper architectures limit their utility for complex LF tasks.

Recurrent Neural Networks (RNNs) [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51], including Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), effectively model sequential dependencies by introducing memory mechanisms that address vanishing gradient issues. These architectures capture both short-term and long-term dependencies, but their step-by-step processing increases computational demands, making them less efficient for very long sequences. Additionally, deeper RNNs, including advanced variants like Bidirectional LSTM (BiLSTM) and Bidirectional GRU (BiGRU), face scalability challenges due to gradient issues and heightened computational complexity.

Transformers [52], [53], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67], [68], [69], leveraging self-attention mechanisms, have emerged as a promising tool for time series forecasting. They excel in modeling long-range dependencies and efficiently handle sequences of varying lengths. However, their computational cost grows quadratically with sequence length, posing challenges for ultra-long sequences. Similar to CNNs and RNNs, training deep Transformer architectures can be unstable, necessitating architectural refinements to address these issues.

As models grow deeper, training challenges in CNNs, RNNs, and Transformers often limit their scalability, hindering their ability to learn complex patterns. To address these issues, Chen et al. [70] proposed a Deep Residual Network (DRN) for STLF and introduced modifications to enhance its performance. Unlike conventional structures, DRNs incorporate residual connections to alleviate gradient vanishing problems, enabling stable training of deep architectures. These networks effectively leverage historical load, temperature, and time data as input, reducing the need for extensive feature engineering while automatically extracting complex features. CNNs, although effective at capturing local patterns, struggle with modeling long-term dependencies. RNN-based models are better at handling temporal sequences but are prone to gradient degradation and high computational costs. While Transformers can efficiently model long-range dependencies, their high resource demands and instability in deeper configurations pose challenges for real-time forecasting. By addressing these limitations, DRNs provide a balanced solution—combining depth, training stability, and expressive power—making them particularly well-suited for STLF tasks that demand both high accuracy and operational efficiency.

Currently, research on DRN in STLF mainly focuses on historical load data, time variables, and temperature variables [70], [71], [72], [73], [74], [75], [76], [77], [78], [79]. However, weather factors are often more diverse, including not only temperature but also rainfall, wind speed, and other elements. Therefore, optimizing DRN models to incorporate more comprehensive weather data remains a promising area of research. Thus, optimizing the DRN model to integrate more comprehensive meteorological data remains an area worth investigating. Principal Component Analysis (PCA) has been utilized in STLF as an effective dimensionality reduction method, helping to identify key features from complex meteorological datasets [80], [81]. Incorporating multidimensional weather variables into DRN models has the potential to enhance forecasting performance by capturing additional meteorological influences. However, as the number of input variables increases, model complexity also rises, potentially leading to overfitting and reduced computational efficiency. By using PCA to extract key features, the complexity introduced by high-dimensional data can be alleviated, which in turn may improve the model's generalization ability.

This study plans to use the Malaysia dataset with multiple weather variables for training, which includes not only historical load variables, time variables, and temperature variables but also additional meteorological variables such as rainfall and wind speed. Through experiments, the study aims to evaluate the impact of historical load variables, time variables, and different combinations of meteorological variables on the performance of DRN models in STLF. At the same time, PCA is utilized to extract key features and simplify the input, mitigating potential risks of overfitting introduced by high-dimensional meteorological variables. This approach

provides a systematic framework to analyze the role of meteorological variables in load forecasting and supports the optimization of DRN models for STLF.

The main contributions of this study are threefold. First, a PCA-DRN framework is developed to address the challenges of high-dimensional and correlated meteorological inputs, effectively reducing feature redundancy and improving forecast accuracy for STLF. Second, a comprehensive comparative analysis of different weather feature combinations is conducted using the Malaysia dataset, revealing that temperature is the dominant factor influencing electricity demand in Malaysia's tropical climate. Third, the proposed approach is rigorously validated through ablation experiments, statistical significance testing, and robustness evaluation under missing-data scenarios, demonstrating its superior predictive accuracy.

The remainder of this paper is organized as follows: Section II reviews the application of DRN for STLF; Section III provides an overview of the methods in this study, including data preprocessing, variable selection, and experimental setup; Section IV presents the experimental results and discusses in detail the impact of adopting different variable combinations on the model's performance using the Malaysia dataset; finally, Section V summarizes the main findings of this study and proposes directions and recommendations for future research.

II. DRN FOR STLF

A. DRN STRUCTURE IN BASIC

The DRN is employed to unravel the intricate nonlinear interplay between input data and the resulting output [82]. Generally, a neural network's learning potential escalates with increased model depth. Yet, paradoxically, this depth might, in reality, impede the deep learning model's efficacy. This decline in performance could stem from either the intrinsic complexities of the data or the sophisticated nature of the model itself. To address this challenge, residual blocks are incorporated into the architecture. In these blocks, the learning process isn't about mapping directly from input to output but rather about mapping from input to a residual function. This approach facilitates the effective training of deeper networks by optimizing the learning process through residual connections, ensuring better gradient flow and reducing the risk of vanishing gradients. As depicted in Figure 3, a residual network (ResNet) features two sequential levels bridged by a skip connection.

A skip connection typically operates as an identical mapping when the dimensions of its input and output align. Under these conditions, the corresponding ResNet's output is as follows Equation (1):

$$y_{\text{output}} = x_{\text{input}} + F(x_{\text{input}}, \Theta) \quad (1)$$

where x_{input} represents the input of the ResNet, y_{output} denotes the block output, F is the residual mapping function, and Θ indicates the learnable parameters within this function.

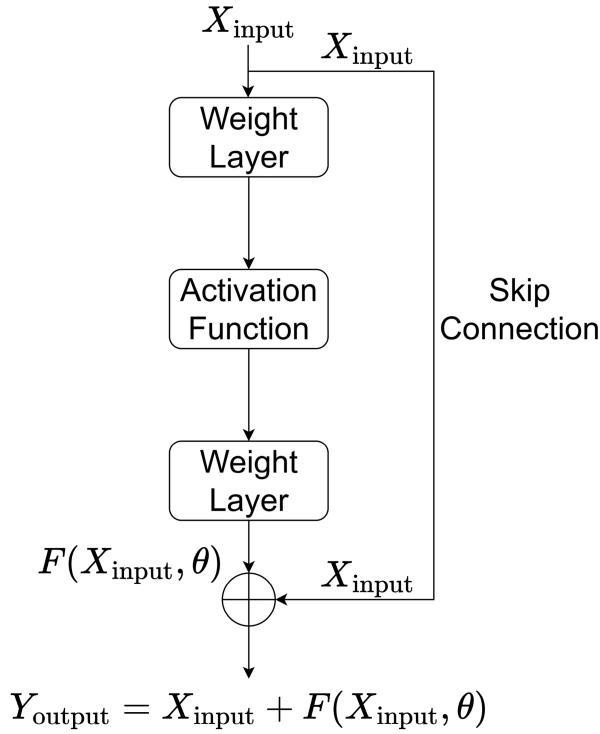


FIGURE 3. The structure of ResNet [70].

However, when input and output dimensions differ, the skip connection assumes the role of a linear projection. In such instances, the associated ResNet yields an output that integrates this linear projection (L_p), as outlined in Equation (2):

$$y_{output} = L_p * x_{Input} + F(x_{Input}, \Theta) \quad (2)$$

Stacking multiple residual blocks allows for the straightforward construction of a Multi-layer ResNet. Figure 4 depicts the structure of a Multi-layer ResNet.

This skip connection indicates that the learning capacity of a residual block (ResBlock) or ResNet is, at minimum, on par with that of an equivalently layered stack. When n residual blocks are sequentially arranged, the Equation (3) for forward-propagation is as follows:

$$y(x) = x_0 + \sum_{j=1}^n F(x_{j-1}, \Theta_j - 1) \quad (3)$$

where x_0 is the input of the residual network, x_n the output of the residual network, and $\Theta_j = \{\theta_{j,z} | 1 \leq z \leq Z\}$ the set of weights associated with the j th residual block, Z being the number of layers within the block.

The back propagation of the overall loss of the neural network to x_0 can then be calculated as Equation (4):

$$\frac{\partial \text{loss}}{\partial x_0} = \frac{\partial \text{loss}}{\partial x_n} \left(1 + \frac{\partial}{\partial x_0} \sum_{j=1}^n F(x_{j-1}, \Theta_j - 1) \right) \quad (4)$$

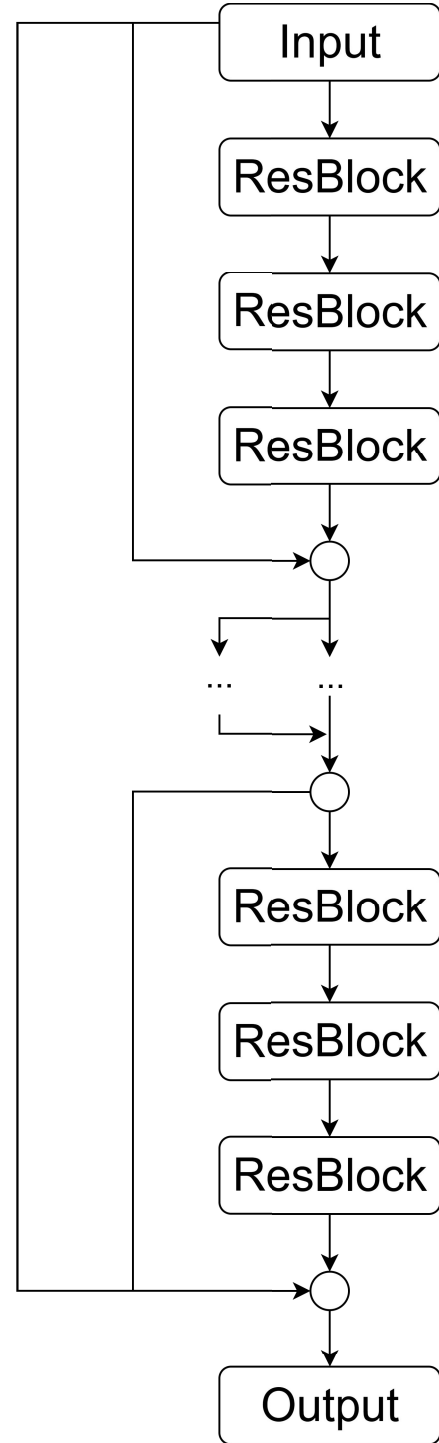


FIGURE 4. The structure of a Multi-layer ResNet [70].

In the given Equation, represents the total loss of the neural network. The presence of “1” signifies that gradients from the network’s output can be directly propagated backward to its input. This direct back-propagation reduces the likelihood of gradient vanishing, a common issue when gradients must traverse multiple layers before reaching the input, thus enhancing the network’s learning efficiency.

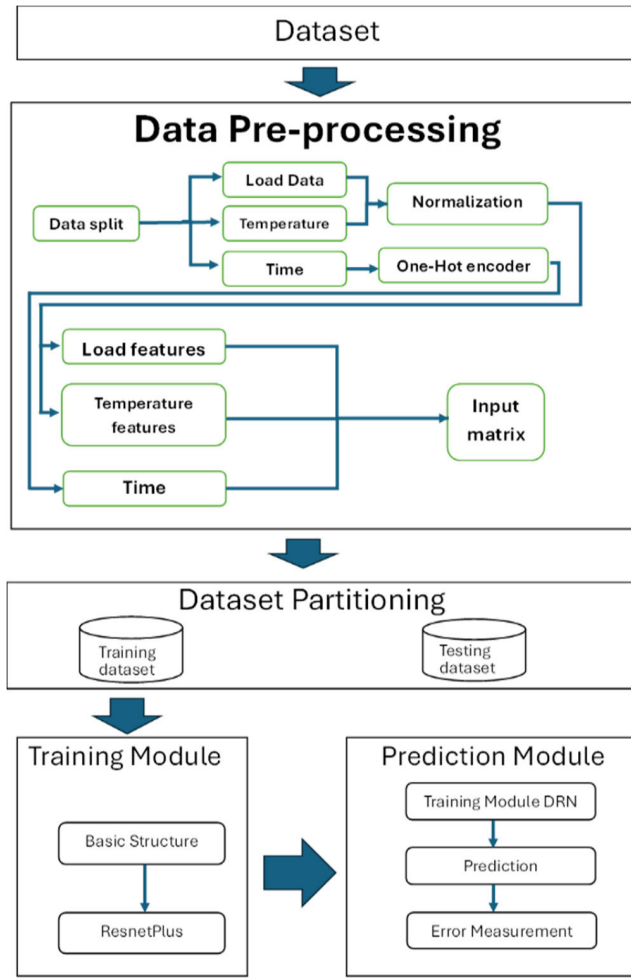


FIGURE 5. Workflow of the DRN for STLF.

B. STRUCTURE OF DRN FOR STLF

Figure 5 depicts the workflow of the DRN model, highlighting the preprocessing steps applied to time variables, load data, and temperature variables. The preprocessing involves normalizing numerical data and using one-hot encoding for categorical variables to prepare the input matrix. This matrix then serves as the input for the DRN architecture, which includes both training and prediction stages. The figure outlines the entire process, from constructing the input matrix to making predictions and evaluating errors. The DRN for STLF, based on the structure detailed in [70], primarily comprises a basic structure and the modified ResNet (ResNetPlus). ResNetPlus, an enhanced version of ResNet designed to improve 24-hour load forecasting performance, retains the block structure of ResNet while incorporating refinements for better prediction precision.

To begin with, a neural network featuring densely connected layers, commonly referred to as the ‘basic structure’ is utilized. This foundational architecture is responsible for generating an initial load forecast for the upcoming 24 hours. The visual depiction of the model employing the basic structure is presented in Figure 6. Within this architecture, every

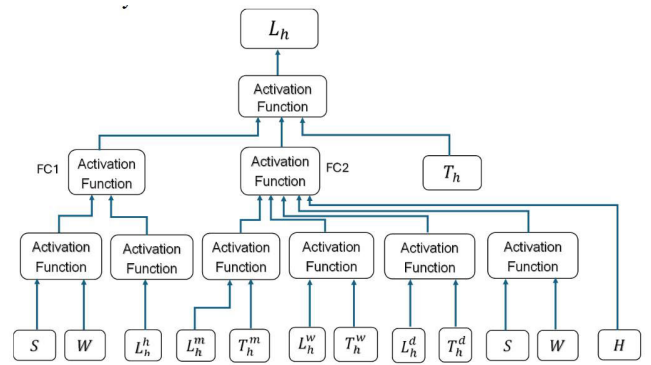


FIGURE 6. The basic structure for one-hour load forecasting [70].

fully connected (FC) layer corresponding to $[L_h^{\text{day}}, T_h^{\text{day}}]$, $[L_h^{\text{week}}, T_h^{\text{week}}]$, $[L_h^{\text{month}}, T_h^{\text{month}}]$ and L_h^{hour} comprises 10 hidden nodes. On the other hand, the fully-connected layers associated with $[S, W]$ are equipped with 5 hidden nodes. Additionally, both fully connected layer FC1, FC2, and the fully-connected layer preceding L_h contain 10 hidden nodes. It is important to note that all layers, except for the output layer, utilize the Scaled Exponential Linear Unit (SELU) activation function to enhance model training stability and performance.

The SELU activation function is defined as follows in Equation (5):

$$f(x) = \begin{cases} \lambda x & \text{if } x > 0 \\ \lambda \alpha (e^x - 1) & \text{if } x \leq 0 \end{cases} \quad (5)$$

In this expression, x represents the input value. The parameters $\lambda \approx 1.05$ and $\alpha \approx 1.67$ serve as scaling factors, where λ ensures normalization and α adjusts the response for negative inputs. SELU promotes self-normalization by maintaining consistent mean and variance throughout the network layers.

In this framework, L_h^{month} denotes the load values for the corresponding hour from the days 1, 2, and 3 months before the predicted day. L_h^{week} signifies the load values for the same hour from 1 to 8 weeks prior, and L_h^{day} corresponds to the loads of the same hour for each day of the preceding week. L_h^{hour} represents the load values for the same hour from the previous 24 hours.

Additionally, T_h^{month} , T_h^{week} and T_h^{day} are the temperature readings concurrent with L_h^{month} , L_h^{week} and L_h^{day} , respectively. T_h is the actual temperature forecasted for the next day. S , W , and H are one-hot encoded variables representing the season, weekday, and holiday status, respectively. The output from this basic structure, denoted as L_h , serves as the input for the second segment of the model, enhancing the forecasting precision.

The ResNetPlus model builds upon the foundational principles of the original ResNet architecture, introducing key improvements to enhance performance and structural depth. It features a series of residual blocks, with each block composed of two hidden layers, each containing 20 neurons and

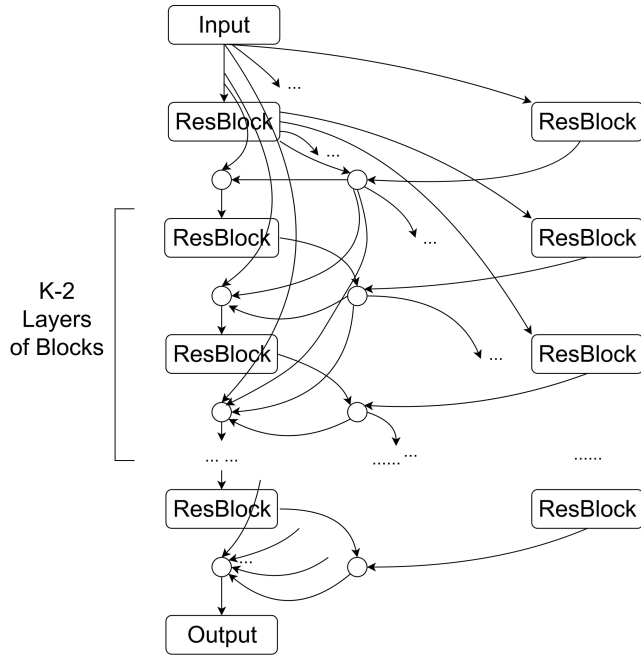


FIGURE 7. A depiction of the modified structure of the ResNetPlus [70].

employing the same activation function. Four such blocks are sequentially connected to form a unit, and this unit is repeated across ten layers, allowing the model to capture deeper and more complex patterns. A distinctive feature of ResNetPlus is its long-range skip connection, which links the output of the final block directly back to the input of the network, thereby influencing the final prediction. This architectural enhancement improves the network's representational capacity while maintaining the residual nature of the original design. As illustrated in Figure 7, ResNetPlus retains the core hyperparameters used in standard ResNet blocks but extends the architecture to fully leverage the advantages of deep residual learning.

The model's loss, represented as Loss, is the sum of two separate components in order to efficiently train the models. Equation (6):

$$\text{Loss} = \text{Loss}_E + \text{Loss}_R \quad (6)$$

where Loss_E quantifies the discrepancy in predictions, and Loss_R serves as a penalty term for out-of-range values, designed to expedite the training phase. Particularly, Loss_E is articulated as Equation (7):

$$\text{Loss}_E = \frac{1}{\text{Num}H} \sum_{j=1}^N \sum_{h=1}^H \frac{|\hat{y}_{(j,h)} - y_{(j,h)}|}{y_{(j,h)}} \quad (7)$$

where $\hat{y}_{(j,h)}$ represents the model's output and $y_{(j,h)}$ denotes the actual normalized load for the h th hour of the j th day. Here, Num symbolizes the number of data samples, while H indicates the number of hourly loads within a day (notably, $H = 24$ in this scenario). This metric, commonly recognized as the MAPE, is employed both as a measure of error and as a

criterion for assessing the forecast results of the models. The second term, Loss_R , is computed as Equation (8):

$$\text{Loss}_R = \frac{1}{2\text{Num}} \sum_{j=1}^{\text{Num}} \max \left(0, \max_h \hat{y}_{(j,h)} - \max_h y_{(j,h)} \right) + \max \left(0, \min_h y_{(j,h)} - \min_h \hat{y}_{(j,h)} \right) \quad (8)$$

This term speeds up the first round of training by penalizing the model if the expected daily load curves diverge from the actual load ranges. This phrase emphasizes the cost of overestimating the load curves' peaks and underestimating their troughs when the model starts to produce forecasts with higher precision.

C. LIMITATIONS IN UTILIZING WEATHER VARIABLES IN DRN FOR STLF

Even with DRN's improvements for STLF, there are still significant gaps in the current body of research. Most DRN-based STLF studies [70], [71], [72], [73], [74], [75], [77], [78], [79] primarily focus on a narrow set of input variables, particularly historical load, time features, and temperature. Although a few studies, such as [76], incorporate additional weather variables like humidity, the majority still rely heavily on temperature as the sole meteorological input. While temperature clearly has a significant impact on predicting energy demand, limiting the model to this single factor may overlook the broader and more complex influence of weather on load patterns.

Furthermore, many of these studies do not fully examine important meteorological factors such as wind speed and rainfall, which can significantly influence energy consumption patterns. For example, wind speed affects both natural ventilation and electricity generation in renewable systems, while rainfall impacts outdoor activity levels and cooling demands. The exclusion of these variables may lead to oversimplified models, especially in regions with diverse climatic conditions like Malaysia, where seasonal and weather-related fluctuations are pronounced.

Adding more input variables to the model increases its complexity, which is another drawback. Although expanding the variety of meteorological data may improve model precision, it also raises the risk of overfitting—particularly in deep architectures like DRNs. High-dimensional input data can reduce generalizability and increase computational demands. However, systematic strategies to mitigate these risks—such as feature selection or dimensionality reduction techniques—are often absent in current research.

In conclusion, the challenges of handling high-dimensional weather data and the underutilization of diverse meteorological features remain key limitations in current DRN-based STLF research. These issues highlight the need for further exploration of feature engineering and model optimization techniques to strike a balance between predictive performance and architectural complexity.

III. PRINCIPAL COMPONENT ANALYSIS (PCA) IN STLF

A popular method for processing and analyzing high-dimensional data, particularly in STLF, is PCA, a traditional dimensionality reduction approach [80], [81]. As much of the original data as feasible is preserved in the lower-dimensional space by projecting the original data onto a new orthogonal coordinate system using linear transformations [83]. Finding the principle components, or the main directions of the data distribution, and maximizing the variance of the data are the objectives of PCA. The precision and computing efficiency of subsequent analyses are frequently weakened in high-dimensional datasets due to feature redundancy and multicollinearity. By eliminating correlations and lowering feature dimensionality, PCA efficiently streamlines the data structure and produces cleaner, more effective inputs for machine learning models.

In terms of mathematics, PCA starts by creating the covariance matrix C , which quantifies the linear correlations between features as indicated by Equation (9):

$$C = \frac{1}{n-1} X^T X \quad (9)$$

where n is the number of samples, and X is the standardized data matrix.

Next, using eigen decomposition, the covariance matrix's eigenvalues λ_i and eigenvectors v_i are determined, as indicated by Equation (10):

$$Cv_i = \lambda_i v_i \quad (10)$$

The variance explained by each major component is represented by the eigenvalues λ_i , and the directions of these components are defined by the eigenvectors v_i . The top eigenvectors that correspond to the biggest eigenvalues are chosen as the principal components after the eigenvalues are ranked in descending order. To guarantee that the majority of the original information is retained in the reduced data, the top k eigenvectors are usually chosen so that the cumulative explained variance achieves a threshold, which is frequently set at 90% [81].

Equation (11) illustrates how the reduced data is finally obtained by projecting the original data onto the chosen primary components:

$$X_{\text{reduced}} = XW \quad (11)$$

where the top k eigenvectors form the matrix W . This procedure makes it possible to reduce the dimensionality of the data while keeping the most important information, which makes subsequent analysis more effective.

Building on its theoretical strengths, PCA has been effectively applied in STLF to reduce feature dimensionality, eliminate redundancy, and enhance model performance. In Bianchi et al. [80], PCA was applied to compress high-dimensional daily load profiles in the Rome power system. By simplifying the input structure and allowing separate forecasting of principal components using Echo State Networks, the model achieved a 16.5% reduction in normalized

root mean square error (NRMSE) compared to a baseline Autoregressive Integrated Moving Average Model (ARIMA) model.

However, the study also indicated that the orthogonality among principal components degrades over time, leading to cumulative approximation errors during long-term forecasting. In Bian et al. [81], PCA was combined with a grey model to extract core temperature-related features, reducing computational complexity and achieving stable prediction performance with errors maintained within a $\pm 0.04\%$ range. Nevertheless, the linear nature of PCA limits its ability to represent nonlinear dependencies between weather conditions and load behavior, which may affect its generalizability in more complex forecasting scenarios.

While PCA has demonstrated clear benefits in simplifying high-dimensional inputs and improving forecasting precision in STLF applications, several general limitations remain that warrant attention. As a linear transformation technique, PCA is inherently constrained by its assumption of linear relationships among variables, which may not adequately capture the nonlinear dynamics present in real-world load and weather data. Additionally, PCA is sensitive to noise, as irrelevant or unstable features can distort the principal components and compromise model robustness. Interpretability also poses a challenge, since principal components are abstract combinations of original variables, making it difficult to trace specific feature contributions—an important consideration for transparent and explainable forecasting in practical settings.

Cleaning, transformation, reduction, and discretization are crucial phases in the pre-processing stage. By methodically evaluating and refining data, these procedures enable more precise and effective forecasts. To further optimize data for analysis, other sub-phases could be added, depending on the dataset's properties, technique, and input requirements.

IV. RESEARCH METHODOLOGY

A. RESEARCH DATA

1) PRE-PROCESSING OF DATA

Irregularities including noise, partial records, missing values, and unprocessed formats are frequently seen in datasets [84]. Analytical precision may be compromised by these problems with raw data, which may lead to mistakes or misunderstandings. In order to improve system efficiency, increase dependability, and allow valuable insights from real-world information, pre-processing is therefore essential.

Cleaning, transformation, reduction, and discretization are crucial phases in the pre-processing stage. By methodically evaluating and refining data, these procedures enable more precise and effective forecasts. To further optimize data for analysis, other sub-phases could be added, depending on the dataset's properties, technique, and input requirements.

2) DATA DESCRIPTION

This study utilizes the Malaysia dataset (data sources are supplied in Appendix), which provides a unique perspective

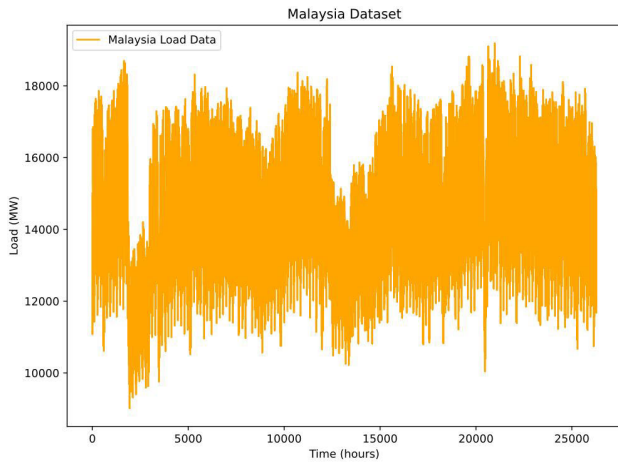


FIGURE 8. Load data in Malaysia dataset.

on load forecasting in a tropical climate. The dataset, obtained from the Malaysia Grid System Operator for load data and the Malaysian Meteorological Department for weather data, spans January 2020 to December 2022 and includes hourly load data along with daily weather variables such as rain-fall, mean temperature, minimum temperature, maximum temperature, mean wind speed, maximum wind speed, and maximum wind direction.

The Malaysia dataset reflects relatively stable demand patterns with moderate variations, making it suitable for exploring load forecasting in environments with limited seasonal influence. The load in this dataset generally ranges from 10,000 megawatts (MW) to 18,000 MW. Figure 8 illustrates the load data, highlighting its characteristics and relevance for evaluating forecasting algorithms in tropical climates.

3) INPUT FEATURES FOR DRN ON THE MALAYSIA DATASET

Because of its daily temporal granularity, the Malaysia dataset requires particular feature processing in the DRN model. Unlike the finer-grained hourly data used in other contexts, this dataset contains daily temperature variables like T_{mean} , T_{max} , T_{min} . To accommodate this, the basic structure was modified to directly integrate daily temperature data as input, as depicted in Figure 9. In this revised setup, daily temperature features (T_{mean} , T_{max} , T_{min}) are combined into a single input without further temporal breakdown. The technique for analyzing load characteristics (L_h^{month} , L_h^{week} , L_h^{day}) stays unaltered, continuing to extract insights from the preceding 24 hours, 8 weeks, and 3 months. Additional date-related inputs, including S , W , and H , are merged with load and temperature characteristics to produce the final model input. The two main seasons of the year are rainy and dry, according to the Malaysia dataset, and important holidays include Eid al-Fitr and Malaysia Independence Day.

This change simplifies preprocessing by enabling the direct use of daily temperature characteristics without the need to artificially extend them into hourly data. The updated structure guarantees that the model adjusts to Malaysian

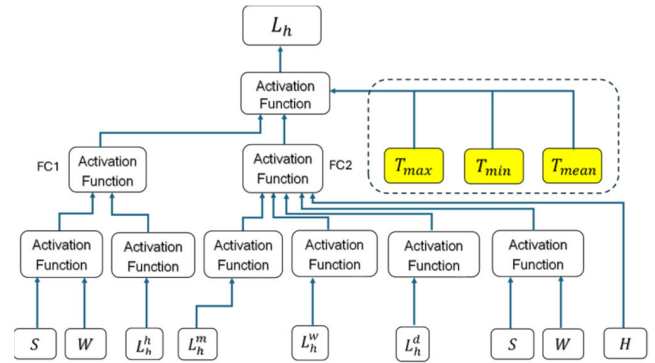


FIGURE 9. Adjusted basic structure for one-hour load forecasting in Malaysia dataset [2].

TABLE 1. Input feature combinations for DRN model in STLF.

Model	Scenario	Time	Load	Tempera ture	Wind	Rainf all
DRN	1	✓	✓	✓		
	2	✓	✓		✓	
	3	✓	✓			✓
	4	✓	✓	✓	✓	
	5	✓	✓	✓		✓
	6	✓	✓		✓	✓
	7	✓	✓	✓	✓	✓

datasets with different temporal resolutions by processing load and date-related characteristics consistently.

B. PCA-DRN STRUCTURE FOR STLF

To evaluate the performance of the DRN model under various weather-related input conditions, this study constructs a series of input feature combinations based on the Malaysia dataset. The dataset offers rich meteorological information, including temperature, wind speed, and rainfall, which are known to influence electricity demand patterns. Given the diverse nature of these weather variables and their potential impact on load forecasting, it is essential to assess how different combinations affect the predictive capability of the DRN.

Therefore, multiple models are configured by systematically varying the inclusion of temperature, wind, and rainfall variables, while consistently retaining historical load and time features, which serve as the core predictors. These combinations allow for a comparative analysis of the model's sensitivity to different types of weather inputs and the effectiveness of each configuration in improving forecasting accuracy. The details of these input feature combinations are presented in Table 1.

To better address the complexity and high dimensionality of weather-related inputs, the proposed PCA-DRN model for STLF integrates PCA with DRNs. This hybrid structure leverages the strengths of both components: PCA acts as a dimensionality reduction technique that removes redundancy

and noise from meteorological variables, while the DRN component is designed to capture deep temporal dependencies within time-series data. By incorporating PCA during the data preparation phase, the model reduces input complexity, improves computational efficiency, and preserves the most informative features. These principal components, along with historical load and time variables, constitute the final input matrix fed into the DRN for forecasting. As illustrated in Figure 10, the whole workflow of the PCA-DRN model comprises data preparation, model training and prediction, and performance evaluation. Weather, historical load, and time variables are segmented, normalized, and one-hot encoded during the data preparation step. By eliminating redundant variables and preserving important information, principal components are extracted from meteorological variables using PCA, producing a low-dimensional principal component matrix. The final input matrix is made up of historical load features, time features, and the chosen principal component features (accumulating until they surpass 90% of the variance). After feeding the input matrix into the DRN, the load forecasting is finished when the residual connection structure captures deep temporal relationships. The error measurement module assesses the model's performance, while the prediction module generates daily load projections.

C. DESIGN OF EXPERIMENTS

This study employs DRN as the primary training framework to investigate the impact of multidimensional weather factors on STLF performance. Multiple combinations of meteorological variables, including temperature, wind speed, and rainfall, were systematically tested to evaluate their individual and joint contributions. To enhance generalization and reduce input dimensionality, PCA was applied to extract key features, resulting in the development of a PCA-enhanced DRN model. In addition to DRN and PCA-DRN, several baseline deep learning architectures—CNN, RNN-based models (LSTM, GRU, BiLSTM, BiGRU), Transformer, Multi-layer ResNet, and PCA-Multi-layer ResNet (included as part of the ablation study to isolate and assess the contribution of PCA)—were implemented for comparison. The CNN was designed as a one-dimensional Conv1D network with 64 filters, a kernel size of 3, ReLU activation, and 'same' padding, while the RNN-based models used 64 recurrent units with default activations. The Transformer adopted a classical configuration with one encoder layer, eight 64-dimensional attention heads, a 64-dimensional embedding, a 2048-dimensional feed-forward network, positional encoding, and a 0.1 dropout rate. The Multi-layer ResNet consisted of 10 stacked ResNet, matching the depth of the DRN architecture, and its PCA-enhanced variant was also included for comparison. All models were trained and tested under identical experimental conditions, with the same input features and preprocessing procedures, enabling a fair and objective performance comparison.

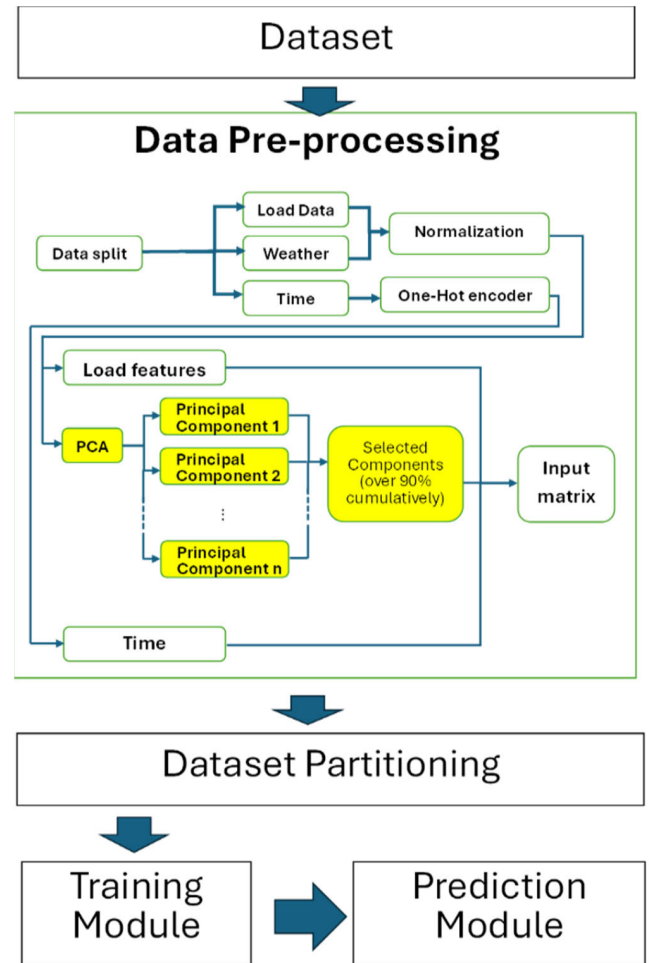


FIGURE 10. Workflow of the PCA-DRN Model for STLF.

A snapshot ensemble strategy was adopted, saving model weights (snapshots) at regular intervals during training [70], [85]. This approach reduces the risk of overfitting a single model while improving prediction stability and generalization by averaging outputs from multiple snapshots. The training process comprised 600 initial epochs followed by two rounds of 50 short-term epochs, yielding three snapshots saved at the end of each short-term phase. Each individual model was trained within eight hours, although extending beyond this threshold did not adversely affect predictive accuracy; this benchmark reflected practical computational efficiency rather than a limiting factor.

Default parameters were selected based on prior research. The adaptive moment estimation (Adam) optimizer with adaptive learning rate adjustment was used, initialized at 0.001 [86]. The Malaysia dataset, spanning 2020–2022, includes load data and multidimensional meteorological factors, making it suitable for investigating STLF in tropical climates. All experiments were conducted using Keras 2.10.0 and TensorFlow 2.10.0 in a Python 3.8 environment, running on a Lenovo laptop equipped with an AMD Ryzen 7 6800H CPU and Radeon Graphics. This setup provided a stable training environment and a solid foundation for future research and model optimization.

To rigorously assess whether the performance improvement of PCA-DRN over DRN is meaningful, a non-parametric Bootstrap resampling procedure with 10,000 iterations was incorporated into the experimental design. Unlike the paired Student's t-test, which assumes normality of the paired differences, the Bootstrap method makes no distributional assumptions, providing a robust framework for comparing model performance [87]. Statistical significance was determined by examining whether the 95% confidence interval of the mean difference between the two models' performance values excluded zero. A 95% confidence interval entirely greater than zero indicates a statistically significant improvement at the 95% confidence level, whereas a 95% confidence interval including zero suggests that the difference is not significant. MAPE was selected as the evaluation metric because it offers an interpretable, scale-independent measure of relative prediction error that is widely used in STLF.

To further evaluate the stability of the PCA-DRN model under imperfect data conditions, a robustness test based on the method proposed by [88] was conducted by introducing artificial missingness into the weather features of the test set.

Specifically, 5%, 10%, and 20% of the meteorological data points were randomly removed and imputed using the mean values derived from the training set. This imputation was performed in the standardized feature space to ensure consistency with the PCA preprocessing. The retrained PCA-DRN models were then evaluated on the modified test datasets, and their forecasting accuracy was compared against the PCA-DRN model trained on the complete dataset, thereby assessing their resilience to sensor failures or delayed weather information.

Overall, the paper systematically examines how multidimensional weather factors affect load forecasting performance and how well PCA simplifies input characteristics using this experimental approach. These results offer a useful foundation for optimizing the model.

D. EVALUATION METRICS

Researchers use a range of criteria to evaluate prediction precision in order to evaluate the performance of different DRN models in STLF [70], [71], [72], [73], [74], [75], [76], [77], [78], [79]. Among these, Mean Absolute Percentage Error (MAPE) is the most commonly used and representative metric, due to its intuitive interpretation and widespread adoption in load forecasting studies. Other frequently used metrics include Mean Absolute Error (MAE), MSE, Normalized Mean Square Error (NMSE), Correlation Coefficient (R), and Coefficient of Determination (R^2). Different studies may adopt different evaluation metrics based on specific objectives and dataset characteristics. Equations (12) through (18) show the corresponding formulas.

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \quad (12)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (13)$$

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (14)$$

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (15)$$

$$\text{NMSE} = \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{N \cdot \sigma_y^2} \quad (16)$$

$$R = \frac{\sum_{i=1}^N (y_i - \hat{y}) (\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^N (y_i - \hat{y})^2 \sum_{i=1}^N (\hat{y}_i - \bar{\hat{y}})^2}} \quad (17)$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (18)$$

These metrics' parameters are as follows: N is the total number of samples; y_i is the actual value for the i -th sample; $\hat{y}_i$ is the predicted value for the same sample; $\hat{y}$ and $\bar{\hat{y}}$ are the mean values of the actual and predicted data, respectively; and σ_y^2 is the variance of the actual values, which is used in NMSE calculations. These parameters enable the metrics to account for error magnitude, prediction precision, and the relationship between actual and predicted values, allowing them to assess performance in a comprehensive manner. Higher R and R^2 values suggest improved precision and better model fitting, whereas lower values for MAPE, RMSE, MAE, MSE, and NMSE often indicate fewer prediction errors and greater generalization.

V. RESULTS AND DISCUSSION

The experiment was used to forecast the entire year of 2022 after being trained on data from Malaysia from 2020 to 2021.

A. IMPACT OF DIFFERENT WEATHER VARIABLE INPUTS

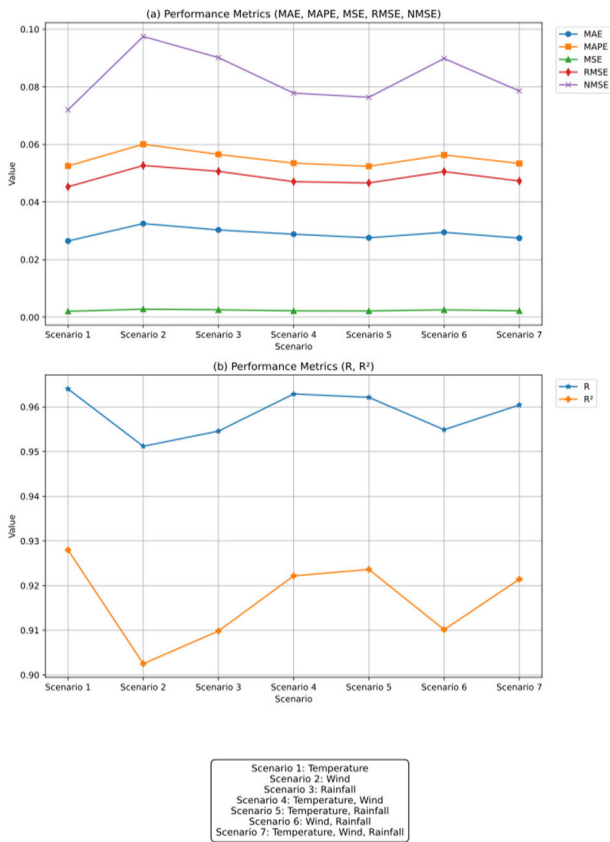
To examine the influence of meteorological factors on STLF, several input combinations of weather variables were tested, as shown in Table 2. Specifically, temperature, wind, and rainfall were each evaluated both individually and in combination, aiming to assess their respective and joint contributions to the DRN model's predictive performance. These experiments were conducted using the Malaysia dataset, and the results are also summarized in Table 2 and illustrated in Figure 11.

With the lowest MAPE (0.052514) and the highest R^2 (0.927993), the model that utilizes only temperature data demonstrates superior predictive accuracy. In contrast, introducing rainfall or wind individually as inputs decreases the model's precision, as reflected in higher MAPE values—0.056534 for rainfall and 0.060087 for wind—accompanied by lower R^2 values (0.909838 and 0.902506, respectively).

When temperature is combined with rainfall, the model achieves a slightly better performance (MAPE: 0.052405, R^2 : 0.923621) than the combinations involving wind (MAPE: 0.053510 for temperature and wind, R^2 : 0.922178) or all three

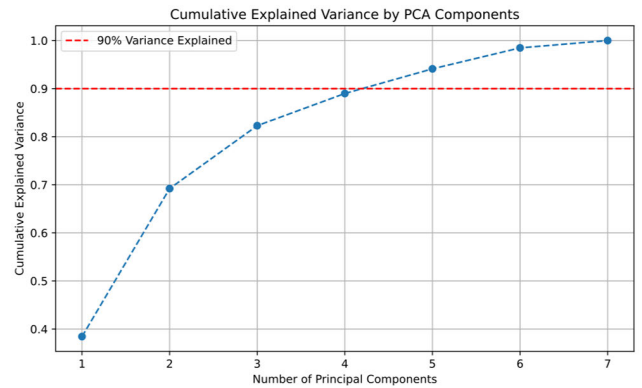
TABLE 2. Performance metrics for DRN model in STLF with different weather variables.

Scenario	MAPE	RMSE	MAE	MSE	NMSE	R	R ²
1	0.0525	0.0452	0.026	0.002	0.0720	0.964	0.927
	14	78	467	05	07	032	993
2	0.0600	0.0526	0.032	0.002	0.0974	0.951	0.902
	87	85	506	776	94	189	506
3	0.0565	0.0506	0.030	0.002	0.0901	0.954	0.909
	34	65	301	567	62	590	838
4	0.0535	0.0470	0.028	0.002	0.0778	0.962	0.922
	10	71	826	216	22	905	178
5	0.0524	0.0466	0.027	0.002	0.0763	0.962	0.923
	05	32	582	175	79	146	621
6	0.0563	0.0505	0.029	0.002	0.0898	0.954	0.910
	54	81	489	558	61	907	139
7	0.0534	0.0472	0.027	0.002	0.0785	0.960	0.921
	14	98	454	237	75	438	425

**FIGURE 11.** Performance Metrics for DRN for STLF with Different Weather Variables: (a) MAPE, RMSE, MAE, MSE, NMSE; (b) R and R².

weather variables (MAPE: 0.053414, R²: 0.921425). Nevertheless, none of these combinations surpass the predictive performance of using temperature alone.

These findings underscore the dominant role of temperature in STLF under tropical climate conditions represented by the Malaysia dataset. Adding wind or rainfall tends to introduce more complexity without substantial gains in accuracy and, in some cases, may even degrade performance—potentially due to increased noise or overfitting risks.

**FIGURE 12.** Cumulative explained variance by PCA components.

B. PCA IN REDUCING DIMENSIONALITY

To reduce the dimensionality of the weather feature set in the Malaysia dataset and mitigate multicollinearity, PCA was applied. The number of retained components was determined using a cumulative explained variance criterion of over 90%, resulting in the selection of five principal components. These five components collectively capture 94.12% of the total variance in the original weather variables, with individual contributions of 37.88% (PCA1), 19.62% (PCA2), 14.23% (PCA3), 12.08% (PCA4), and 8.54% (PCA5), respectively. Figure 12 illustrates the cumulative explained variance of the principal components.

The loadings of the retained components are shown in Table 3. The loadings indicate the contribution of each original weather variable to a principal component, enabling the interpretation of the underlying features represented by each component. PCA1 is dominated by negative contributions from mean, minimum, and maximum temperatures, indicating that it primarily reflects overall temperature variations. PCA2 shows strong positive loadings for maximum wind speed (0.7018) and rainfall (0.4422), highlighting its representation of extreme wind and precipitation patterns. PCA3 is characterized by a dominant negative loading on maximum wind direction (−0.9463), making it a wind direction-focused component. PCA4 shows a strong negative contribution from mean wind speed (−0.7924) and a moderate positive contribution from maximum temperature (0.4138), reflecting wind–temperature interaction effects. PCA5 is primarily influenced by rainfall (0.7097) and mean wind speed (0.3875), representing rainfall–wind coupling.

By transforming the correlated weather variables into these five uncorrelated principal components, PCA not only reduces redundancy and multicollinearity but also provides a more compact and insightful feature representation for subsequent modeling.

C. EXPERIMENT RESULTS

The experiment evaluated and compared the performance of various deep learning models for STLF, as presented in Table 4 and illustrated in Figure 13. The models include

TABLE 3. PCA component loadings.

Weather Variable	PCA1	PCA2	PCA3	PCA4	PCA5
Rainfall	0.3261 46	0.4421 54	0.099 971	0.3817 35	0.709 699
Mean Temperature	- 0.5798 88	- 0.0823 23	- 0.091 294	0.1419 74	0.086 094
Minimum Temperature	- 0.5036 39	- 0.2037 54	- 0.011 57	0.1472 26	0.123 478
Maximum Temperature	- 0.4424 09	- 0.3655 51	- 0.130 403	0.4137 82	- 0.051 551
Mean Wind Speed	- 0.2930 4	- 0.3188 46	- 0.079 191	- 0.7923 6	0.387 48
Maximum Wind Speed	- 0.1051 15	- 0.7018 37	- 0.250 668	- 0.0879 86	- 0.512 842
Maximum Wind Direction	- 0.1051 59	- 0.1683 48	- 0.946 27	- 0.0751 94	- 0.240 546

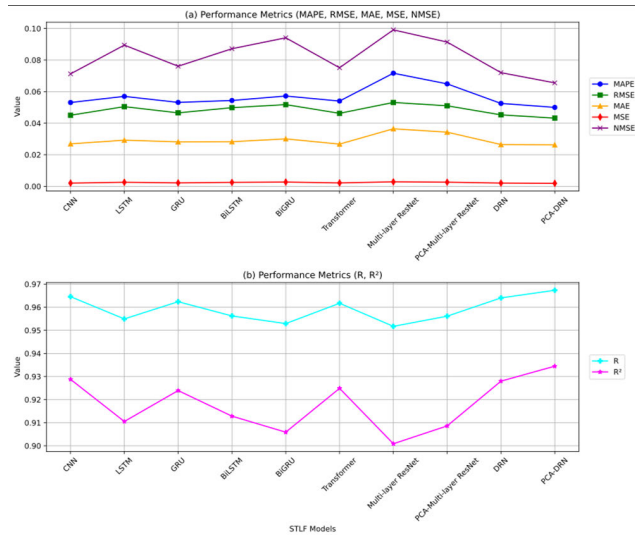


FIGURE 13. Comparison of Performance Metrics Across Different Models: (a) MAPE, RMSE, MAE, MSE, NMSE; (b) R and R².

CNN, LSTM, GRU, BiLSTM, BiGRU, Transformer, Multi-layer ResNet, PCA-Multi-layer ResNet and DRN, along with a PCA-DRN. The results show variations in predictive precision across the different architectures. Notably, the DRN outperforms most baseline models, while incorporating PCA into the DRN further improves performance. To provide additional insights into the training behavior, the loss curves of DRN and PCA-DRN were also plotted, as shown in Figure 14.

Firstly, the DRN model demonstrates superior predictive performance compared to other baseline models. Specifically, DRN, which adopts a 10-layer ResNetPlus structure, achieves a lower MAPE (0.052514) and higher R² (0.927993) than CNN (MAPE: 0.053096, R²: 0.928768), GRU (MAPE:

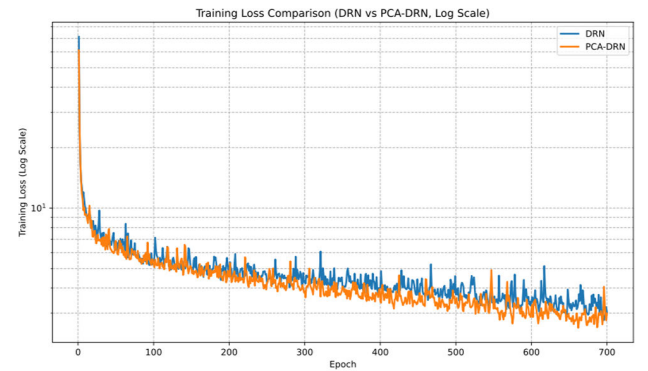


FIGURE 14. Training Loss Comparison (DRN vs PCA-DRN).

0.053170, R²: 0.923920), BiLSTM (MAPE: 0.054373, R²: 0.912835), and BiGRU (MAPE: 0.057201, R²: 0.905904). This indicates that DRN is more effective at capturing complex temporal patterns in load forecasting. Although the Transformer model performs closely (MAPE: 0.054016, R²: 0.924876), it still underperforms relative to DRN. To further analyze the effect of PCA and residual network architecture, a Multi-layer ResNet model with the same depth (10 layers) but a conventional ResNet structure was introduced as an ablation model. Its performance (MAPE: 0.071664, R²: 0.900881) is notably worse than DRN, suggesting that the ResNetPlus design offers a meaningful advantage over a conventional ResNet with the same depth.

Secondly, incorporating PCA further improves the performance of residual networks. The PCA-DRN achieves the lowest MAPE (0.049994) and the highest R² (0.934473) across all evaluated models, outperforming the original DRN by reducing MAPE by approximately 4.79% and increasing R² by 0.65 percentage points. Similarly, applying PCA to the Multi-layer ResNet structure (PCA-Multi-layer ResNet) also yields substantial improvements over its non-PCA counterpart (MAPE reduced from 0.071664 to 0.064901, R² increased from 0.900881 to 0.908612). These ablation experiments confirm that PCA effectively reduces multicollinearity and eliminates redundant features, thereby enhancing the model's generalization ability and predictive precision.

The training loss curves in Figure 14, plotted on a logarithmic scale, indicate that both models experience a rapid decline in loss during the initial epochs, followed by a gradual stabilization. Notably, PCA-DRN achieves a lower overall loss and a smoother convergence trajectory compared to DRN, reflecting improved training stability. These observations suggest that incorporating PCA not only accelerates convergence but also reduces fluctuations during optimization, implying a potential regularizing effect that enhances the robustness of the model training process.

The improvement of PCA-DRN can also be visually observed in Figure 15, which compares the predicted load generated by PCA-DRN with the actual load, showing a close alignment between predicted and actual values.

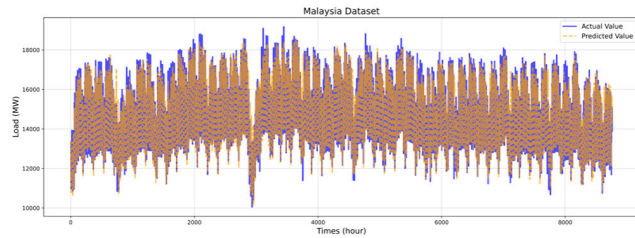


FIGURE 15. The comparison of PCA-DRN prediction with the actual load.

Beyond predictive performance, the computational overhead of the two DRN variants was also analyzed to assess their practicality for real-time deployment. Table 5 presents the comparison results. The baseline DRN model utilized only temperature features, while the PCA-DRN incorporated multiple weather variables (temperature, wind speed, and rainfall) and applied PCA to reduce dimensionality. Despite the increase in input diversity, the introduction of PCA only slightly increased the model parameters by 0.4% (from 111,544 to 112,024), indicating negligible growth in model size. The total training time also showed minimal change, with a difference of less than 1%. Notably, PCA significantly improved inference efficiency, reducing the single-sample prediction time from 5.95 ms to 5.40 ms (a reduction of 9.2%). This improvement suggests that PCA effectively mitigated the computational cost introduced by additional features, enabling faster forward passes while maintaining the model's representational capability. Considering the hourly forecasting horizon, these sub-6 ms inference times confirm that both models are well-suited for real-time deployment in practical STLF applications.

In summary, DRN outperforms conventional CNN, RNN-based, and Transformer models even without PCA, and the inclusion of PCA leads to additional improvements. These findings demonstrate that integrating PCA into the data preparation stage provides a practical and effective approach to enhance both predictive accuracy and computational efficiency in DRN-based STLF models.

D. STATISTICAL SIGNIFICANCE TESTING

To determine whether the performance improvement of PCA-DRN over DRN is statistically significant, a non-parametric Bootstrap resampling procedure with 10,000 iterations was applied to the sample-wise MAPE differences between the two models' predictions. The PCA-DRN model achieved a lower mean MAPE (0.049994 ± 0.088989) compared to the DRN model (0.052514 ± 0.106031). The Bootstrap analysis yielded a mean difference of 0.002520, with a 95% confidence interval of [0.001409, 0.003658], which is entirely greater than 0. These results confirm that the reduction in prediction error achieved by PCA-DRN is statistically significant at the 95% confidence level, indicating that the improvement is unlikely to be due to random variation.

TABLE 4. Comparison of performance metrics across different models.

Model	MAPE	RMSE	MAE	MSE	NMSE	R	R ²
CNN	0.0530 96	0.0450 34	0.026 905	0.002 028	0.0712 32	0.96 4570	0.92 876 8
LSTM	0.0569 78	0.0504 70	0.029 211	0.002 547	0.0894 69	0.95 4934	0.91 053 1
GRU	0.0531 70	0.0465 41	0.028 172	0.002 166	0.0760 80	0.96 2390	0.92 392 0
BiLSTM	0.0543 73	0.0498 16	0.028 251	0.002 482	0.0871 65	0.95 6222	0.91 283 5
BiGRU	0.0572 01	0.0517 59	0.030 055	0.002 679	0.0940 96	0.95 2908	0.90 590 4
Transformer	0.0540 16	0.0462 47	0.026 741	0.002 139	0.0751 24	0.96 1710	0.92 487
Multi-layer ResNet	0.0716 64	0.0531 22	0.036 444	0.002 822	0.0991 19	0.95 1720	0.90 088 1
PCA-Multi-layer ResNet	0.0649 01	0.0510 08	0.034 299	0.002 602	0.0913 88	0.95 6120	0.90 861 2
DRN	0.0525 14	0.0452 78	0.026 467	0.002 05	0.0720 07	0.96 4032	0.92 799 3
PCA-DRN	0.0499 94	0.0431 93	0.026 254	0.001 866	0.0655 27	0.96 7339	0.93 447 3

TABLE 5. Comparison of performance metrics across different models.

Model	Parameters	Training Time (s)	Inference Time (s)	Inference Time / Sample (ms)
DRN	111544	1425.77 3076	2.172467	5.951963
PCA-DRN	112024	1436.78 8505	1.970398	5.398351

TABLE 6. R² of PCA-DRN under different missing rates.

Missing Rate (%)	R ²
0	0.934473
5	0.873494
10	0.879152
20	0.891214

E. ROBUSTNESS TESTING

To evaluate the robustness of the proposed PCA-DRN model under incomplete data scenarios, different levels of missingness were introduced to the input features (5%, 10%, and 20%), and the model performance was assessed in terms of R². As shown in Table 6, the R² score decreases substantially from 0.934473 to 0.873494 when 5% of the data are missing, indicating that even a small amount of missing information noticeably affects model accuracy. Interestingly, when the missing rate increases to 10% and 20%, the R²

values slightly recover to 0.879152 and 0.891214, respectively. Although performance decreases compared to the complete dataset, the model consistently maintains R^2 values above 0.87 even under 20% missingness, demonstrating that PCA-DRN retains acceptable forecasting accuracy and exhibits robustness against data incompleteness. This counterintuitive trend may result from the regularizing effect of data imputation, which can help mitigate overfitting under moderate missingness. Nevertheless, all missing-data scenarios yield lower R^2 scores compared to the complete dataset, underscoring the importance of high-quality and complete input data for achieving optimal forecasting performance.

F. SUMMARY

The experimental results provided several important insights. First, temperature emerged as the most dominant weather variable, with the DRN model using only temperature achieving the lowest MAPE (0.052514) and the highest R^2 (0.927993). In contrast, including wind speed or rainfall individually reduced predictive accuracy, while combining these variables with temperature did not yield further improvements. This finding underscores that in Malaysia's tropical climate, temperature largely drives electricity demand, whereas additional weather variables may introduce noise or overfitting risks without offering substantial predictive benefits.

Second, the DRN outperformed baseline deep learning models, including CNN, LSTM, GRU, BiLSTM, BiGRU, Transformer, and a conventional multi-layer ResNet, demonstrating its ability to capture complex temporal dependencies in load data. The 10-layer ResNetPlus structure used in the DRN was particularly effective in improving forecasting precision compared to a traditional ResNet of equal depth.

Third, PCA proved to be an effective dimensionality reduction technique for handling high-dimensional meteorological data. The proposed PCA-DRN achieved the best overall performance (MAPE: 0.049994, R^2 : 0.934473), improving MAPE by 4.79% and R^2 by 0.65 percentage points compared to the baseline DRN. PCA effectively reduced feature redundancy and multicollinearity, leading to improved generalization, faster convergence, and a smoother training process.

In addition, a non-parametric Bootstrap analysis with 10,000 resamples verified that the performance improvement of PCA-DRN over DRN was statistically significant, as the 95% confidence interval of the mean difference did not include zero, underscoring the reliability of the results. Robustness testing further indicated that the PCA-DRN maintained acceptable performance even under scenarios with up to 20% artificially introduced missing weather data, confirming its resilience to incomplete inputs. Computational analysis also showed that PCA slightly reduced inference time despite the inclusion of additional weather variables, supporting its practical deployment for real-time forecasting applications.

While these findings provide valuable insights into load forecasting in a tropical climate, their generalizability to other climatic conditions warrants careful consideration. In temperate regions, seasonal variations in temperature, heating demand, and renewable energy integration may significantly alter the relative importance of weather variables. In arid regions, factors such as extreme temperatures and low humidity could lead to different consumption behaviors. Therefore, the observed dominance of temperature in Malaysia's tropical context may not directly translate to these regions, highlighting the need for further research to validate and adapt the proposed framework under diverse climatic conditions.

Collectively, these findings emphasize the dominant role of temperature in tropical STLF, the effectiveness of residual networks in improving predictive performance, and the value of PCA in enhancing both accuracy and computational efficiency. At the same time, they highlight the limited contribution of wind and rainfall in this context, suggesting that careful feature selection and dimensionality reduction are crucial for balancing model complexity and forecasting precision.

VI. CONCLUSION

This study proposed a PCA-DRN for STLF using the Malaysia dataset, integrating multidimensional weather features with residual deep learning and dimensionality reduction techniques. The experimental results demonstrated that temperature is the most critical meteorological factor influencing load demand in Malaysia's tropical climate, while incorporating wind speed and rainfall, either individually or in combination with temperature, did not yield notable improvements and, in some cases, slightly degraded forecasting performance. These findings underscore the dominant role of temperature in relatively stable tropical environments and highlight the importance of careful feature selection when introducing additional meteorological variables. The integration of Principal Component Analysis into the DRN framework effectively reduced feature redundancy and multicollinearity, leading to enhanced generalization and computational efficiency. The proposed PCA-DRN achieved the best overall performance, with a 4.79% reduction in MAPE and a 0.65-point increase in R^2 compared to the baseline DRN, while also exhibiting smoother convergence and more stable training behavior. Furthermore, robustness testing confirmed that the PCA-DRN maintained acceptable forecasting accuracy under incomplete data conditions, and computational analysis indicated efficient inference speeds suitable for real-time operational deployment.

Looking ahead, several promising research directions emerge from this work. Future studies could investigate more advanced gradient optimization strategies, such as Nesterov accelerated gradient descent [89], to further improve convergence speed, enhance training stability, and reduce sensitivity to initialization, which is particularly important for deep residual networks with complex architectures. In parallel, developing innovative architectural designs beyond

the current residual framework—such as integrating hybrid modules that combine convolutional, recurrent, and attention-based components—could enable the model to capture both local and long-term dependencies in load–weather interactions more effectively. Furthermore, extending the framework to temperate and arid regions will be essential for evaluating its generalizability, as the relative importance of meteorological variables may shift significantly in climates with pronounced seasonal changes or extreme environmental conditions. Such cross-climate validation would provide deeper insights into how weather factors influence electricity consumption across diverse contexts. By combining residual deep learning with dimensionality reduction, the proposed PCA-DRN not only improves forecasting accuracy, efficiency, and robustness in tropical power systems but also establishes a versatile foundation for developing next-generation forecasting models that can adapt to varying climatic and operational challenges.

ABBREVIATIONS

The following abbreviations are used in this manuscript:

Abbreviation	Full name.
Adam	Adaptive Moment Estimation.
ARIMA	Autoregressive Integrated Moving Average Model.
ANN	Artificial Neural Network.
BiGRU	Bidirectional Gated Recurrent Unit.
BiLSTM	Bidirectional Long Short-Term Memory.
CNN	Convolutional Neural Network.
DNN	Deep Neural Network.
DRN	Deep Residual Network.
ELM	Extreme Learning Machine.
ESN	Echo State Network.
GRU	Gated Recurrent Unit.
LSTM	Long Short-Term Memory
LTLF	Long-Term Load Forecasting.
MAE	Mean Absolute Error.
MAPE	Mean Absolute Percentage Error.
MSE	Mean Square Error.
MTLF	Medium-Term Load Forecasting.
MW	Megawatt.
NMSE	Normalized Mean Square Error.
PCA	Principal Component Analysis.
R	Correlation Coefficient.
R ²	Coefficient of Determination.
RBF	Radial Basis Function.
RNN	Recurrent Neural Network.
SELU	Scaled Exponential Linear Unit.
STLF	Short-Term Load Forecasting.
SVR	Support Vector Regression.
VSTLF	Very Short-Term Load Forecasting.

APPENDIX

Malaysia dataset:

1. Load dataset <https://www.gso.org.my/SystemData/SystemDemand.aspx>
2. Weather dataset <https://www.met.gov.my/>

ACKNOWLEDGMENT

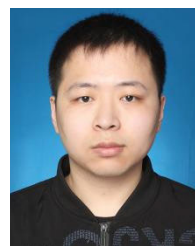
Conceptualization: Junchen Liu and Faisal Arif Ahmad; Methodology: Junchen Liu; Investigation: Junchen Liu and Faisal Arif Ahmad; Writing—original draft preparation: Junchen Liu; Writing—review and editing: Junchen Liu, Faisal Arif Ahmad, Khairulmizam Samsudin, Fazirulhisyam Hashim, and Mohd Zainal Abidin Ab Kadir; and Supervision: Faisal Arif Ahmad, Khairulmizam Samsudin, Fazirulhisyam Hashim, and Mohd Zainal Abidin Ab Kadir. All authors have read and agreed to the published version of the manuscript.

REFERENCES

- [1] F. A. Ahmad, J. Liu, F. Hashim, and K. Samsudin, “Short-term load forecasting utilizing a combination model: A brief review,” *Int. J. Technol.*, vol. 15, no. 1, pp. 121–129, Jan. 2024, doi: [10.14716/ijtech.v15i1.5543](https://doi.org/10.14716/ijtech.v15i1.5543).
- [2] J. Liu, F. A. Ahmad, K. Samsudin, F. Hashim, and M. Z. A. A. Kadir, “Performance evaluation of activation functions in deep residual networks for short-term load forecasting,” *IEEE Access*, vol. 13, pp. 78618–78633, 2025, doi: [10.1109/ACCESS.2025.3565798](https://doi.org/10.1109/ACCESS.2025.3565798).
- [3] S. R. U. Hassnain and A. Khan, “Short-term load forecasting using particle swarm optimization based ANN approach,” in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2007, pp. 1476–1481, doi: [10.1109/IJCNN.2007.4371176](https://doi.org/10.1109/IJCNN.2007.4371176).
- [4] S. Ruzic, A. Vuckovic, and N. Nikolic, “Weather sensitive method for short term load forecasting in electric power utility of Serbia,” *IEEE Trans. Power Syst.*, vol. 18, no. 4, pp. 1581–1586, Nov. 2003, doi: [10.1109/TPWRS.2003.811172](https://doi.org/10.1109/TPWRS.2003.811172).
- [5] K.-B. Song, Y.-S. Baek, D. H. Hong, and G. Jang, “Short-term load forecasting for the holidays using fuzzy linear regression method,” *IEEE Trans. Power Syst.*, vol. 20, no. 1, pp. 96–101, Feb. 2005, doi: [10.1109/TPWRS.2004.835632](https://doi.org/10.1109/TPWRS.2004.835632).
- [6] W. Charytoniuk, M. S. Chen, and P. Van Olinda, “Nonparametric regression based short-term load forecasting,” *IEEE Trans. Power Syst.*, vol. 13, no. 3, pp. 725–730, Mar. 1998, doi: [10.1109/59.708572](https://doi.org/10.1109/59.708572).
- [7] J. W. Taylor, “Short-term electricity demand forecasting using double seasonal exponential smoothing,” *J. Oper. Res. Soc.*, vol. 54, no. 8, pp. 799–805, Aug. 2003, doi: [10.1057/palgrave.jors.2601589](https://doi.org/10.1057/palgrave.jors.2601589).
- [8] J. W. Taylor, “Triple seasonal methods for short-term electricity demand forecasting,” *Eur. J. Oper. Res.*, vol. 204, no. 1, pp. 139–152, Jul. 2010, doi: [10.1016/j.ejor.2009.10.003](https://doi.org/10.1016/j.ejor.2009.10.003).
- [9] E. E. Elattar, J. Goulermas, and Q. H. Wu, “Electric load forecasting based on locally weighted support vector regression,” *IEEE Trans. Syst., Man, Cybern., C Appl. Rev.*, vol. 40, no. 4, pp. 438–447, Jul. 2010, doi: [10.1109/TSMCC.2010.2040176](https://doi.org/10.1109/TSMCC.2010.2040176).
- [10] E. Ceperic, V. Ceperic, and A. Baric, “A strategy for short-term load forecasting by support vector regression machines,” *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4356–4364, Nov. 2013, doi: [10.1109/TPWRS.2013.2269803](https://doi.org/10.1109/TPWRS.2013.2269803).
- [11] G. Zhang and J. Guo, “A novel method for hourly electricity demand forecasting,” *IEEE Trans. Power Syst.*, vol. 35, no. 2, pp. 1351–1363, Mar. 2020, doi: [10.1109/TPWRS.2019.2941277](https://doi.org/10.1109/TPWRS.2019.2941277).
- [12] D. Alberg and M. Last, “Short-term load forecasting in smart meters with sliding window-based ARIMA algorithms,” *Vietnam J. Comput. Sci.*, vol. 5, nos. 3–4, pp. 241–249, Sep. 2018, doi: [10.1007/s40595-018-0119-7](https://doi.org/10.1007/s40595-018-0119-7).
- [13] M. Rejc and M. Pantos, “Short-term transmission-loss forecast for the Slovenian transmission power system based on a fuzzy-logic decision approach,” *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1511–1521, Aug. 2011, doi: [10.1109/TPWRS.2010.2096829](https://doi.org/10.1109/TPWRS.2010.2096829).
- [14] M. Ali, M. Adnan, M. Tariq, and H. V. Poor, “Load forecasting through estimated parametrized based fuzzy inference system in smart grids,” *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 1, pp. 156–165, Jan. 2021, doi: [10.1109/TFUZZ.2020.2986982](https://doi.org/10.1109/TFUZZ.2020.2986982).
- [15] H. S. Hippert, C. E. Pedreira, and R. C. Souza, “Neural networks for short-term load forecasting: A review and evaluation,” *IEEE Trans. Power Syst.*, vol. 16, no. 1, pp. 44–55, Jan. 2001, doi: [10.1109/59.910780](https://doi.org/10.1109/59.910780).

- [16] C. Kuster, Y. Rezgui, and M. Mourshed, "Electrical load forecasting models: A critical systematic review," *Sustain. Cities Soc.*, vol. 35, pp. 257–270, Nov. 2017, doi: [10.1016/j.scs.2017.08.009](#).
- [17] L. Hernández, C. Baladrón, J. M. Aguiar, B. Carro, A. Sánchez-Esguevillas, and J. Lloret, "Artificial neural networks for short-term load forecasting in microgrids environment," *Energy*, vol. 75, pp. 252–264, Oct. 2014, doi: [10.1016/j.energy.2014.07.065](#).
- [18] C. Cecati, J. Kolbusz, P. Różycki, P. Siano, and B. M. Wilamowski, "A novel RBF training algorithm for short-term electric load forecasting and comparative studies," *IEEE Trans. Ind. Electron.*, vol. 62, no. 10, pp. 6519–6529, Oct. 2015, doi: [10.1109/TIE.2015.2424399](#).
- [19] Y. Chen, P. B. Luh, C. Guan, Y. Zhao, L. D. Michel, M. A. Coolbeth, P. B. Friedland, and S. J. Rourke, "Short-term load forecasting: Similar day-based wavelet neural networks," *IEEE Trans. Power Syst.*, vol. 25, no. 1, pp. 322–330, Feb. 2010, doi: [10.1109/TPWRS.2009.2030426](#).
- [20] Y. Zhao, P. B. Luh, C. Bomgardner, and G. H. Beeler, "Short-term load forecasting: Multi-level wavelet neural networks with holiday corrections," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Jul. 2009, pp. 1–7, doi: [10.1109/PES.2009.5275304](#).
- [21] I. Goodfellow, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [22] Y. Eren and İ. Küçükdemir, "A comprehensive review on deep learning approaches for short-term load forecasting," *Renew. Sustain. Energy Rev.*, vol. 189, Jan. 2024, Art. no. 114031, doi: [10.1016/j.rser.2023.114031](#).
- [23] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," *IEEE Trans. Power Syst.*, vol. 27, no. 1, pp. 134–141, Feb. 2012, doi: [10.1109/TPWRS.2011.2162082](#).
- [24] W. He, "Load forecasting via deep neural networks," *Proc. Comput. Sci.*, vol. 122, pp. 308–314, Jan. 2017, doi: [10.1016/j.procs.2017.11.374](#).
- [25] L. Li, K. Ota, and M. Dong, "Everything is image: CNN-based short-term electrical load forecasting for smart grid," in *Proc. 14th Int. Symp. Pervasive Syst., Algorithms Netw. 11th Int. Conf. Frontier Comput. Sci. Technol. 3rd Int. Symp. Creative Comput. (ISPAN-FCST-ISCC)*, Jun. 2017, pp. 344–351, doi: [10.1109/ISPAN-FCST-ISCC.2017.78](#).
- [26] X. Dong, L. Qian, and L. Huang, "Short-term load forecasting in smart grid: A combined CNN and K-means clustering approach," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Feb. 2017, pp. 119–125, doi: [10.1109/BIGCOMP.2017.7881726](#).
- [27] C. Tian, J. Ma, C. Zhang, and P. Zhan, "A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network," *Energies*, vol. 11, no. 12, p. 3493, Dec. 2018, doi: [10.3390/en11123493](#).
- [28] L. Han, Y. Peng, Y. Li, B. Yong, Q. Zhou, and L. Shu, "Enhanced deep networks for short-term and medium-term load forecasting," *IEEE Access*, vol. 7, pp. 4045–4055, 2019, doi: [10.1109/ACCESS.2018.2888978](#).
- [29] M. Cai, M. Pipattanasomporn, and S. Rahman, "Day-ahead building-level load forecasts using deep learning vs. Traditional time-series techniques," *Appl. Energy*, vol. 236, pp. 1078–1088, Feb. 2019, doi: [10.1016/j.apenergy.2018.12.042](#).
- [30] Z. Deng, B. Wang, Y. Xu, T. Xu, C. Liu, and Z. Zhu, "Multi-scale convolutional neural network with time-cognition for multi-step short-term load forecasting," *IEEE Access*, vol. 7, pp. 88058–88071, 2019, doi: [10.1109/ACCESS.2019.2926137](#).
- [31] Z. Kong, C. Zhang, H. Lv, F. Xiong, and Z. Fu, "Multimodal feature extraction and fusion deep neural networks for short-term load forecasting," *IEEE Access*, vol. 8, pp. 185373–185383, 2020, doi: [10.1109/ACCESS.2020.3029828](#).
- [32] Y. Shi, S. Wang, L. Zhang, F. Yang, and X. Luo, "A novel short-term power load forecasting method based on high-dimensional meteorological data dimensionality reduction and hybrid deep neural network," *J. Energy Eng.*, vol. 149, no. 6, Dec. 2023, Art. no. 04023049, doi: [10.1061/jleed9.eyeng-5009](#).
- [33] M. Jurado, M. Samper, and R. Rosés, "An improved encoder–decoder-based CNN model for probabilistic short-term load and PV forecasting," *Electr. Power Syst. Res.*, vol. 217, Apr. 2023, Art. no. 109153, doi: [10.1016/j.epr.2023.109153](#).
- [34] A. Baul, G. C. Sarker, P. Sikder, U. Mozumder, and A. Abdelgawad, "Data-driven short-term load forecasting for multiple locations: An integrated approach," *Big Data Cognit. Comput.*, vol. 8, no. 2, p. 12, Jan. 2024, doi: [10.3390/bdcc8020012](#).
- [35] Ö. F. Ertugrul, "Forecasting electricity load by a novel recurrent extreme learning machines approach," *Int. J. Electr. Power Energy Syst.*, vol. 78, pp. 429–435, Jun. 2016, doi: [10.1016/j.ijepes.2015.12.006](#).
- [36] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo, and Y. Xu, "Short-term residential load forecasting based on resident behaviour learning," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 1087–1088, Jan. 2018, doi: [10.1109/TPWRS.2017.2688178](#).
- [37] A. Narayan and K. W. Hipel, "Long short term memory networks for short-term electric load forecasting," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2017, pp. 2573–2578, doi: [10.1109/SMC.2017.8123012](#).
- [38] D. L. Marino, K. Amarasinghe, and M. Manic, "Building energy load forecasting using deep neural networks," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc.*, Oct. 2016, pp. 7046–7051, doi: [10.1109/IECON.2016.7793413](#).
- [39] P. Bento, J. Pombo, S. Mariano, and M. d. R. Calado, "Short-term load forecasting using optimized LSTM networks via improved bat algorithm," in *Proc. Int. Conf. Intell. Syst. (IS)*, Sep. 2018, pp. 351–357, doi: [10.1109/IS.2018.8710498](#).
- [40] D. Gan, Y. Wang, N. Zhang, and W. Zhu, "Enhancing short-term probabilistic residential load forecasting with quantile long–short-term memory," *J. Eng.*, vol. 2017, no. 14, pp. 2622–2627, Jan. 2017, doi: [10.1049/joe.2017.0833](#).
- [41] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Appl. Energy*, vol. 235, pp. 10–20, Feb. 2019, doi: [10.1016/j.apenergy.2018.10.078](#).
- [42] S. Wang, X. Wang, S. Wang, and D. Wang, "Bi-directional long short-term memory method based on attention mechanism and rolling update for short-term load forecasting," *Int. J. Electr. Power Energy Syst.*, vol. 109, pp. 470–479, Jul. 2019, doi: [10.1016/j.ijepes.2019.02.022](#).
- [43] C. Cai, Y. Tao, T. Zhu, and Z. Deng, "Short-term load forecasting based on deep learning bidirectional LSTM neural network," *Appl. Sci.*, vol. 11, no. 17, p. 8129, Sep. 2021, doi: [10.3390/app11178129](#).
- [44] X. Tang, Y. Dai, Q. Liu, X. Dang, and J. Xu, "Application of bidirectional recurrent neural network combined with deep belief network in short-term load forecasting," *IEEE Access*, vol. 7, pp. 160660–160670, 2019, doi: [10.1109/ACCESS.2019.2950957](#).
- [45] Q. Liu, J. Cao, J. Zhang, Y. Zhong, T. Ba, and Y. Zhang, "Short-term power load forecasting in FGSM-Bi-LSTM networks based on empirical wavelet transform," *IEEE Access*, vol. 11, pp. 105057–105068, 2023, doi: [10.1109/ACCESS.2023.3316516](#).
- [46] M. Liu, Y. Li, J. Hu, X. Wu, S. Deng, and H. Li, "A new hybrid model based on SCINet and LSTM for short-term power load forecasting," *Energies*, vol. 17, no. 1, p. 95, Dec. 2023, doi: [10.3390/en17010095](#).
- [47] W. Wu, W. Liao, J. Miao, and G. Du, "Using gated recurrent unit network to forecast short-term load considering impact of electricity price," *Energy Proc.*, vol. 158, pp. 3369–3374, Feb. 2019, doi: [10.1016/j.egypro.2019.01.950](#).
- [48] B.-S. Kwon, R.-J. Park, and K.-B. Song, "Short-term load forecasting based on deep neural networks using LSTM layer," *J. Electr. Eng. Technol.*, vol. 15, no. 4, pp. 1501–1509, Jul. 2020, doi: [10.1007/s42835-020-00424-7](#).
- [49] S. Atef, K. Nakata, and A. B. Eltawil, "A deep bi-directional long-short term memory neural network-based methodology to enhance short-term electricity load forecasting for residential applications," *Comput. Ind. Eng.*, vol. 170, Aug. 2022, Art. no. 108364, doi: [10.1016/j.cie.2022.108364](#).
- [50] S. Muzaffar and A. Afshari, "Short-term load forecasts using LSTM networks," *Energy Proc.*, vol. 158, pp. 2922–2927, Feb. 2019, doi: [10.1016/j.egypro.2019.01.952](#).
- [51] A. Haque and S. Rahman, "Short-term electrical load forecasting through heuristic configuration of regularized deep neural network," *Appl. Soft Comput.*, vol. 122, Jun. 2022, Art. no. 108877, doi: [10.1016/j.asoc.2022.108877](#).
- [52] Z. Zhao, C. Xia, L. Chi, X. Chang, W. Li, T. Yang, and A. Y. Zomaya, "Short-term load forecasting based on the transformer model," *Information*, vol. 12, no. 12, p. 516, Dec. 2021, doi: [10.3390/info12120516](#).
- [53] S. Huang, J. Zhang, Y. He, X. Fu, L. Fan, G. Yao, and Y. Wen, "Short-term load forecasting based on the CEEMDAN-sample entropy-BPNN-transformer," *Energies*, vol. 15, no. 10, p. 3659, May 2022, doi: [10.3390/en15103659](#).
- [54] P. Ran, K. Dong, X. Liu, and J. Wang, "Short-term load forecasting based on CEEMDAN and transformer," *Electr. Power Syst. Res.*, vol. 214, Jan. 2023, Art. no. 108885, doi: [10.1016/j.epr.2022.108885](#).

- [55] B. Jiang, Y. Liu, H. Geng, H. Zeng, and J. Ding, "A transformer based method with wide attention range for enhanced short-term load forecasting," in *Proc. 4th Int. Conf. Smart Power Internet Energy Syst. (SPIES)*, Beijing, China, 2022, pp. 1684–1690, doi: [10.1109/SPIES55999.2022.10082249](https://doi.org/10.1109/SPIES55999.2022.10082249).
- [56] G. Zhang, C. Wei, C. Jing, and Y. Wang, "Short-term electrical load forecasting based on time augmented transformer," *Int. J. Comput. Intell. Syst.*, vol. 15, no. 1, p. 67, Aug. 2022, doi: [10.1007/s44196-022-00128-y](https://doi.org/10.1007/s44196-022-00128-y).
- [57] A. Upadhyay, D. Garg, and M. Singh, "Short term load forecasting for smart grids using apache spark and a modified transformer model," *Comput. Informat.*, vol. 42, no. 1, pp. 75–97, 2023, doi: [10.31577/cai_2023_1_75](https://doi.org/10.31577/cai_2023_1_75).
- [58] H. Liao and K. K. Radhakrishnan, "Short-term load forecasting with temporal fusion transformers for power distribution networks," in *Proc. IEEE Sustain. Power Energy Conf. (iSPEC)*, Dec. 2022, pp. 1–5, doi: [10.1109/iSPEC54162.2022.10033079](https://doi.org/10.1109/iSPEC54162.2022.10033079).
- [59] P. C. Huy, N. Q. Minh, N. D. Tien, and T. T. Q. Anh, "Short-term electricity load forecasting based on temporal fusion transformer model," *IEEE Access*, vol. 10, pp. 106296–106304, 2022, doi: [10.1109/ACCESS.2022.3211941](https://doi.org/10.1109/ACCESS.2022.3211941).
- [60] E. Giacomazzi, F. Haag, and K. Hopf, "Short-term electricity load forecasting using the temporal fusion transformer: Effect of grid hierarchies and data sources," in *Proc. 14th ACM Int. Conf. Future Energy Syst.*, Jun. 2023, pp. 353–360, doi: [10.1145/3575813.3597345](https://doi.org/10.1145/3575813.3597345).
- [61] Q. Zhang, J. Chen, G. Xiao, S. He, and K. Deng, "TransformGraph: A novel short-term electricity net load forecasting model," *Energy Rep.*, vol. 9, pp. 2705–2717, Dec. 2023, doi: [10.1016/j.egyr.2023.01.050](https://doi.org/10.1016/j.egyr.2023.01.050).
- [62] C. Xu and G. Chen, "Interpretable transformer-based model for probabilistic short-term forecasting of residential net load," *Int. J. Electr. Power Energy Syst.*, vol. 155, Jan. 2024, Art. no. 109515, doi: [10.1016/j.ijepes.2023.109515](https://doi.org/10.1016/j.ijepes.2023.109515).
- [63] S. Cen and C. G. Lim, "Multi-task learning of the PatchTCN-TST model for short-term multi-load energy forecasting considering indoor environments in a smart building," *IEEE Access*, vol. 12, pp. 19553–19568, 2024, doi: [10.1109/ACCESS.2024.3355448](https://doi.org/10.1109/ACCESS.2024.3355448).
- [64] W. Lu and X. Chen, "Short-term load forecasting for power systems with high-penetration renewables based on multivariate data slicing transformer neural network," *Frontiers Energy Res.*, vol. 12, Jan. 2024, Art. no. 1355222, doi: [10.3389/fenrg.2024.1355222](https://doi.org/10.3389/fenrg.2024.1355222).
- [65] Z. Lin, L. Xie, and S. Zhang, "A compound framework for short-term gas load forecasting combining time-enhanced perception transformer and two-stage feature extraction," *Energy*, vol. 298, Jul. 2024, Art. no. 131365, doi: [10.1016/j.energy.2024.131365](https://doi.org/10.1016/j.energy.2024.131365).
- [66] B. Fang, L. Xu, Y. Luo, Z. Luo, and W. Li, "A method for short-term electric load forecasting based on the FMLP-iTransformer model," *Energy Rep.*, vol. 12, pp. 3405–3411, Dec. 2024, doi: [10.1016/j.egyr.2024.09.023](https://doi.org/10.1016/j.egyr.2024.09.023).
- [67] V. A. N. Valencia and J. E. Sanchez-Galan, "Use of attention-based neural networks to short-term load forecasting in the republic of Panama," in *Proc. IEEE 40th Central Amer. Panama Conv. (CONCAPAN)*, Nov. 2022, pp. 1–6, doi: [10.1109/CONCAPAN48024.2022.9997752](https://doi.org/10.1109/CONCAPAN48024.2022.9997752).
- [68] M. L. Santos, S. D. García, X. García-Santiago, A. Ogando-Martínez, F. E. Camarero, G. B. Gil, and P. C. Ortega, "Deep learning and transfer learning techniques applied to short-term load forecasting of data-poor buildings in local energy communities," *Energy Buildings*, vol. 292, Aug. 2023, Art. no. 113164, doi: [10.1016/j.enbuild.2023.113164](https://doi.org/10.1016/j.enbuild.2023.113164).
- [69] S. Li, W. Zhang, and P. Wang, "TS2ARCformer: A multi-dimensional time series forecasting framework for short-term load prediction," *Energies*, vol. 16, no. 15, p. 5825, Aug. 2023, doi: [10.3390/en16155825](https://doi.org/10.3390/en16155825).
- [70] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, "Short-term load forecasting with deep residual networks," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3943–3952, Jul. 2019, doi: [10.1109/TSG.2018.2844307](https://doi.org/10.1109/TSG.2018.2844307).
- [71] Q. Xu, X. Yang, and X. Huang, "Ensemble residual networks for short-term load forecasting," *IEEE Access*, vol. 8, pp. 64750–64759, 2020, doi: [10.1109/ACCESS.2020.2984722](https://doi.org/10.1109/ACCESS.2020.2984722).
- [72] Y. Tian, S. Yu, M. Wen, K. Zhang, and Y. Chen, "Short-term load forecasting scheme based on improved deep residual network and LSTM," in *Proc. CIRED Berlin Workshop (CIRED)*, vol. 2020, Sep. 2020, pp. 117–120, doi: [10.1049/OAP-CIRED.2021.0257](https://doi.org/10.1049/OAP-CIRED.2021.0257).
- [73] V. Y. Kondaiah and B. Saravanan, "Short-term load forecasting with deep learning," in *Proc. Innov. Power Adv. Comput. Technol. (i-PACT)*, Nov. 2021, pp. 1–5, doi: [10.1109/i-PACT52855.2021.9696634](https://doi.org/10.1109/i-PACT52855.2021.9696634).
- [74] A. Ding, T. Liu, and X. Zou, "Integration of ensemble GoogLeNet and modified deep residual networks for short-term load forecasting," *Electronics*, vol. 10, no. 20, p. 2455, Oct. 2021, doi: [10.3390/electronics10202455](https://doi.org/10.3390/electronics10202455).
- [75] Z. Sheng, H. Wang, G. Chen, B. Zhou, and J. Sun, "Convolutional residual network to short-term load forecasting," *Appl. Intell.*, vol. 51, no. 4, pp. 2485–2499, Apr. 2021, doi: [10.1007/s10489-020-01932-9](https://doi.org/10.1007/s10489-020-01932-9).
- [76] H. Li, P. Zhang, and C. Li, "Short-term load forecasting for distribution substations based on residual neural networks and long short-term memory neural networks with attention mechanism," *J. Phys., Conf. Ser.*, vol. 2030, no. 1, Sep. 2021, Art. no. 012087, doi: [10.1088/1742-6596/2030/1/012087](https://doi.org/10.1088/1742-6596/2030/1/012087).
- [77] W. Chen, G. Han, H. Zhu, L. Liao, and W. Zhao, "Deep ResNet-based ensemble model for short-term load forecasting in protection system of smart grid," *Sustainability*, vol. 14, no. 24, p. 16894, Dec. 2022, doi: [10.3390/su142416894](https://doi.org/10.3390/su142416894).
- [78] V. Y. Kondaiah and B. Saravanan, "A modified deep residual network for short-term load forecasting," *Frontiers Energy Res.*, vol. 10, Oct. 2022, Art. no. 1038819, doi: [10.3389/fenrg.2022.1038819](https://doi.org/10.3389/fenrg.2022.1038819).
- [79] Z. Sheng, Z. An, H. Wang, G. Chen, and K. Tian, "Residual LSTM based short-term load forecasting," *Appl. Soft Comput.*, vol. 144, Sep. 2023, Art. no. 110461, doi: [10.1016/j.asoc.2023.110461](https://doi.org/10.1016/j.asoc.2023.110461).
- [80] F. M. Bianchi, E. De Santis, A. Rizzi, and A. Sadeghian, "Short-term electric load forecasting using echo state networks and PCA decomposition," *IEEE Access*, vol. 3, pp. 1931–1943, 2015, doi: [10.1109/ACCESS.2015.2485943](https://doi.org/10.1109/ACCESS.2015.2485943).
- [81] H. H. Bian, Q. Wang, and L. Tian, "Research on short-term load forecasting based on PCA-GM," in *Proc. 2nd EAI Int. Conf. Multimedia Technol. Enhanced Learn. (ICMTEL)*, Apr. 2020, pp. 169–177, doi: [10.1007/978-3-030-51103-6_15](https://doi.org/10.1007/978-3-030-51103-6_15).
- [82] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [83] H. Abdi and L. J. Williams, "Principal component analysis," *WIREs Comput. Statist.*, vol. 2, no. 4, pp. 433–459, Jul. 2010, doi: [10.1002/wics.101](https://doi.org/10.1002/wics.101).
- [84] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—A novel pooling deep RNN," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 5271–5280, Sep. 2018, doi: [10.1109/TSG.2017.2686012](https://doi.org/10.1109/TSG.2017.2686012).
- [85] G. Huang, Y. Li, G. Pleiss, Z. Liu, J. E. Hopcroft, and K. Q. Weinberger, "Snapshot ensembles: Train 1, get m for free," 2017, arXiv:1704.00109.
- [86] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, arXiv:1412.6980.
- [87] M. G. Johnston and C. Faulkner, "A bootstrap approach is a superior statistical method for the comparison of non-normal data with differing variances," *New Phytologist*, vol. 230, no. 1, pp. 23–26, Apr. 2021, doi: [10.1111/nph.17159](https://doi.org/10.1111/nph.17159).
- [88] C. John, E. J. Ekpenyong, and C. C. Nworu, "Imputation of missing values in economic and financial time series data using five principal component analysis (PCA) approaches," *Cbn, J. Appl. Statist.*, vol. 10, no. 1, pp. 51–73, Aug. 2019, doi: [10.33429/cjas.10119.3/6](https://doi.org/10.33429/cjas.10119.3/6).
- [89] G. Tan, W.-H. Chen, J. Yang, X.-T. Tran, and Z. Li, "Dual control for autonomous airborne source search with Nesterov accelerated gradient descent: Algorithm and performance analysis," *Neurocomputing*, vol. 630, May 2025, Art. no. 129729.



JUNCHEN LIU received the Bachelor of Engineering degree in electrical engineering and automation from Nanchang Institute of Technology, Nanchang, China, in 2018, and the master's degree in information systems and technology from Irkutsk National Research Technical University, Irkutsk, Russia, in 2021. He is currently pursuing the Ph.D. degree with the Department of Computer and Communication Systems Engineering, Faculty of Engineering, Universiti Putra Malaysia. His research interests include artificial intelligence, short-term load forecasting in power systems, and the intersection of artificial intelligence and electrical engineering.



FAISUL ARIF AHMAD (Member, IEEE) received the B.Eng. degree in information engineering from Muroran Institute of Technology, Muroran, Japan, in 2001, the M.Eng. degree in electrical engineering from Universiti Teknologi Malaysia, in 2009, and the Ph.D. degree from Universiti Putra Malaysia (UPM), in 2016. He was an Engineer with Panasonic AVC Network Johor, Malaysia (formerly known as Matsushita Audio Video Sdn. Bhd.). He is currently a Senior Lecturer with the Department of Computer and Communication Systems Engineering, Faculty of Engineering, UPM, where he has served as a Tutor and was appointed as a Senior Lecturer, in February 2017. His research interests include robotic intelligent systems, swarm intelligence systems, embedded and real-time systems, and artificial intelligence systems.



FAZIRULHISYAM HASHIM (Member, IEEE) received the B.Eng. degree in computer and communication systems from Universiti Putra Malaysia, in 2002, the M.S. degree in computer and communication engineering from Universiti Sains Malaysia, in 2006, and the Ph.D. degree in wireless communication network engineering from The University of Sydney, Australia, in 2010. He is currently an Associate Professor with the Department of Computer and Communication Systems Engineering, Universiti Putra Malaysia. His research interests include mobile networks, network and computer security, wireless sensor networks, software-defined networking, network function virtualization, blockchain, vehicular communication, and emerging networks (5G/6G).



KHAIRULMIZAM SAMSUDIN received the B.Sc. degree in computer systems and communications from Universiti Putra Malaysia, in 1998, and the Ph.D. degree from the University of Glasgow, U.K., in 2002. He is currently a Senior Lecturer with the Computer and Communication Systems Engineering Department, Universiti Putra Malaysia. His research interests include device modeling, embedded systems, robotics, security, web services, and distributed computing.



MOHD ZAINAL ABIDIN AB KADIR (Senior Member, IEEE) received the B.Eng. degree in electrical and electronic engineering from Universiti Putra Malaysia and the Ph.D. degree in high-voltage engineering from The University of Manchester, U.K. He is currently a Strategic Hire Professor with the Institute of Power Engineering (IPE), Universiti Tenaga Nasional (UNITEN), and a Professor with the Faculty of Engineering, Universiti Putra Malaysia. He is also the Founding Director of the Centre for Electromagnetic and Lightning Protection Research (CELP), Universiti Putra Malaysia. He is also an Advisory Board Member of the National Lightning Safety Institute (NLSI), USA, and a Research Advisor of African Centre for Lightning and Electromagnetic (ACLE). To date, he has authored or co-authored over 450 journal articles and conference papers. His research interests include high-voltage engineering, lightning protection, electromagnetic compatibility, power system transients, and renewable energy. He is the Chairperson of the National Mirror Committee of IEC TC 81 (Lightning Protection) and the Local Convener of MNC-CIGRE C4 on System Technical Performance. He is a Professional Engineer (P.Eng.), a Chartered Engineer (C.Eng.), and a Professional Technologist (P.Tech.). He is an IEEE Power and Energy Society (PES) Distinguished Lecturer in the field of lightning and high-voltage engineering.

...