Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 10, No. 3, 357-371
2026
Publisher: Learning Gate
DOI: 10.55214/2576-8484.v10i3.12398

Enhancing LSTM prediction accuracy through hybrid feature engineering: An LSTM-based model for load forecasting of industry in Thailand

Siamrat Phonkaphon[1], Ekawit Songkohb[2], Pramuk Unahalekhaka[1]*
[1]Department of Electrical Engineering, Faculty of Engineering and Architecture, Rajamangala University of Technology Suvarnabhumi (RUS), Nonthaburi 11000, Thailand; siamrat.p@rmutsb.ac.th (S.P.) pramuk.u@rmutsb.ac.th (P.U.).
[2]Department of Industrial Engineering, Faculty of Engineering and Architecture, Rajamangala University of Technology Suvarnabhumi (RUS), Phra Nakhon Si Ayutthaya Hantra 13000, Thailand.

Abstract: This study aims to improve the accuracy of electrical load forecasting in the industrial sector by enhancing the performance of the Long Short-Term Memory (LSTM) model and comparing three deep learning methods: the standard LSTM model, the hybrid CNN–LSTM model, and the Feature Engineering LSTM (FE-LSTM) model. This research uses historical industrial electricity consumption data, environmental temperature data, and data on working days and holidays. This data will be used to forecast electricity load for all three models and evaluate the models' performance using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R^2 value. The research results show that the FE-LSTM model outperforms the other models, with the highest R^2 value of 0.9447, and can reduce the RMSE from 13,220.08 W (standard LSTM) to 8,231.62 W, indicating an improvement in electrical load forecasting by 37.7%. The study concludes that the integration of hybrid feature engineering techniques, such as lagged features, moving averages, and calendar variables, significantly enhances forecasting accuracy. The proposed FE-LSTM model is suitable for short-term industrial load forecasting and can support energy management planning, cost optimization, and operational decision-making in the industrial sector.
Keywords: *Feature engineering, Hybrid model, Load forecasting, LSTM model.*

1. Introduction

Electricity is a fundamental factor that plays a crucial role in human life and the economic development of a country. The continuously increasing demand for electricity results from the expansion of the industrial sector, urban development, population growth, and changing energy consumption behaviors [1-4]. Therefore, efficient energy management is a key mission for energy agencies, particularly the process of electricity consumption forecasting, which is vital for production planning, grid management, and maintaining the overall stability of the electricity system.

Accurate energy consumption forecasting enables efficient advance planning in areas such as production capacity planning, scheduling of power plant operations, fuel cost management, and the development of smart electrical grids that can respond to continuously changing energy demands [5-7]. Additionally, accurate forecasting helps reduce risks to the electrical system from economic and climatic fluctuations and supports the transition to a sustainable, clean energy system in the future [8-10].

Energy consumption forecasting can be divided into three levels: short-term, medium-term, and long-term. Short-term forecasting, which covers periods from a few hours to several days, is crucial for controlling production and allocating sufficient power supply to meet demand. Conversely, medium-term and long-term forecasting are often used to support investment planning and energy policy formulation [11]. In the past, traditional statistical methods such as Autoregressive Integrated Moving Average (ARIMA), exponential smoothing, and regression models were widely used for energy

History: Received: 12 January 2026; Revised: 16 February 2026; Accepted: 20 February 2026; Published: 11 March 2026
* Correspondence: pramuk.u@rmutsb.ac.th

forecasting. However, although these methods have advantages in terms of simplicity and the ability to clearly explain outcomes, they are limited in handling time series data with trends, seasonality, and nonlinear relationships, resulting in reduced accuracy when the data is highly complex [12-14].

In the past decade, deep learning technologies and artificial intelligence methods, such as ANN, CNN, LSTM, and XGBoost, have provided higher forecasting accuracy, with their strengths lying in the ability to learn complex data patterns and adapt well to new data. In particular, LSTM models are capable of effectively learning and remembering temporal relationships both in the short and long term [15-17]. This results in superior abilities to analyze complex time series data patterns and adapt to the volatility of new datasets compared to traditional statistical models. However, using an LSTM model alone for forecasting electricity consumption in real situations may not be sufficient, as energy consumption data does not rely solely on historical data but is also influenced by other external factors such as weather, day of the week, holidays, and seasonal factors [18-21]. A feature enhancement framework based on derivative memory long short-term memory (FEDM-LSTM model) has been presented, using important data from previous production steps, such as temperature, pressure, and level to provide results in the current step. The simulation results show that the FEDM-LSTM model has an accuracy of up to 91.17% [22]. The SFE-LSTM model is an online prediction model that enhances short-term features for key quality variables in industrial processes by using data such as temperature, rainfall, humidity, and atmospheric pressure. The simulation results yield the lowest RMSE, which is better than the traditional LSTM model [23].

For this reason, this research developed the Feature-Enhanced Long Short-Term Memory (FE-LSTM) model approach, which enhances data features before entering the LSTM model by adding variables from lag features, moving averages, and calendar features. This helps the model learn temporal and seasonal patterns more effectively. The approach focuses on improving efficiency through a structured feature enhancement process.

2. Materials and Methods

2.1. Data Collection

In this study, hourly electricity consumption data were collected from an automotive parts manufacturing plant in northeastern Thailand, which operates continuously and is open from Monday to Saturday. The data was recorded 24 hours a day during 2021–2022. A total of 17,520 data sets were collected using an online energy monitoring system as a tool for measuring and storing electrical energy data, along with temperature data in the area and variables indicating working days and holidays, to analyze the temporal patterns of energy demand. The results of the hourly load pattern analysis (00:00–23:00), as shown in Figure 1, reveal that energy consumption behavior can be categorized by time as follows.

- Time period 00:00–06:00. Electricity consumption is low and relatively stable, reflecting the system's base load that must operate continuously.
- From 07:00 to 09:00. The demand for electricity increases rapidly, corresponding with the start of the production process and the operation of machinery.
- The period from 10:00 AM to 3:00 PM is the time with the highest energy consumption of the day, with peaks around 11:00 AM and from 1:00 PM to 2:00 PM, reflecting the full operation of the production line.
- Time period 16:00–18:00. Electricity consumption decreases at the end of the work shift before returning to the base load level during the night.

When comparing weekdays and weekends, it is found that weekends have significantly lower load levels, especially Sundays, reflecting a reduction or cessation of production activities. Additionally, high external temperatures during the day are directly related to an increase in electrical load, particularly from the operation of air conditioning and cooling systems.

ISSN: 2576-8484
Vol. 10, No. 3: 357-371, 2026
DOI: 10.55214/2576-8484.v10i3.12398

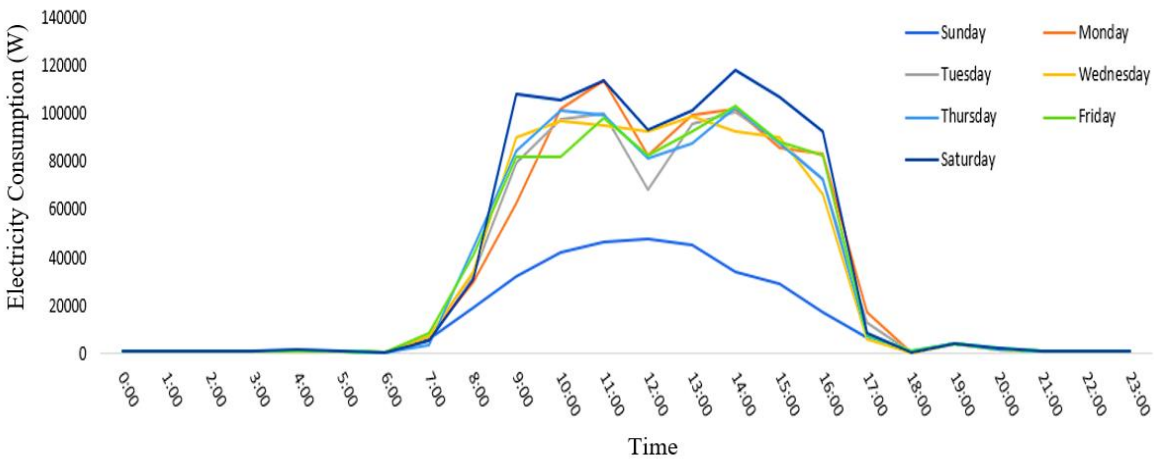

Figure 1.
Weekly Comparison of Hourly Electricity Consumption.

The hourly electricity consumption data will be collected through an online energy monitoring system. The dataset includes the Date-Time variable, the Day of the week variable (Day: 0–6), the hourly temperature (temp, °C), and the electricity consumption (consumption, W). This data will be used to analyze the temporal characteristics of energy demand, with the data recording format shown in Table 1.

Table 1.
Sample of Hourly Electricity Consumption and Environmental Data.

Date-Time	Day	Temp (°C)	Consumption(W)
1/1/2021 0:00	2	16.2	917.5053
1/1/2021 1:00	2	16.1	23,274.58
1/1/2021 2:00	2	21.2	26,956.14
1/1/2021 3:00	2	20.3	2,503.99
…	…	…	…
…	…	…	…
31/12/2022 20:00	5	24.6	21,845.74
31/12/2022 21:00	5	27.3	25,903.44
31/12/2022 22:00	5	22.1	3,154.67
31/12/2022 23:00	5	19.3	1,045.88

Note:

- "Day" (0–6) refers to the order of days in the week:
  0 = Sunday, 1 = Monday, 2 = Tuesday,
  3 = Wednesday, 4 = Thursday, 5 = Friday,
  6 = Saturday
- Temp (°C) is the hourly temperature, measured in degrees Celsius (°C).
- Consumption is measured in W.

The min–max normalization method is applied to prepare the data before entering the modeling process. This method is used to adjust data values to a range between 0 and 1, which helps reduce scale differences between variables and increases the consistency of the original data. Scaling the data in this manner helps the model learn data patterns more efficiently. Additionally, the issue of bias caused by data values with a wide range is reduced, resulting in improved accuracy and stability of predictions. The equation used for min-max data normalization is shown in the equation (1).

$$\widetilde{X_t} = \frac{(X_t - X_{min})}{(X_{max} - X_{min})} \qquad (1)$$

$$X_t = \widetilde{X_t}(X_{max} - X_{min}) + X_{min} \quad (2)$$

Where:

$\widetilde{X_t}$ represents the normalized value.

X_t is the original value at time step t.

X_{min} and X_{max} denote the minimum and maximum values of the variable computed from the training dataset.

After the load forecasting process is completed, the scaled data is converted back to its actual values using the formula presented in equation (2).

2.2. Time Series Model

2.2.1. Long Short-Term Memory (LSTM model)

The LSTM model is a type of artificial neural network developed by Sepp Hochreiter and Jürgen Schmidhuber to address the vanishing gradient problem in Recurrent Neural Networks (RNNs), involving a forget gate and an input gate, which make it inefficient to learn long-term data sequences. The structure of the LSTM model incorporates forget gates, input gates, and output gate mechanisms along with the cell state to control the flow of data over time, resulting in better learning of long-term temporal relationships. The structure of the LSTM model is shown in Figure 2. Previous research has shown that the LSTM model can significantly improve the accuracy of electrical load forecasting, especially when applying time delay adjustment techniques [24]. It also demonstrates high efficiency in forecasting household load, which is highly variable [25].

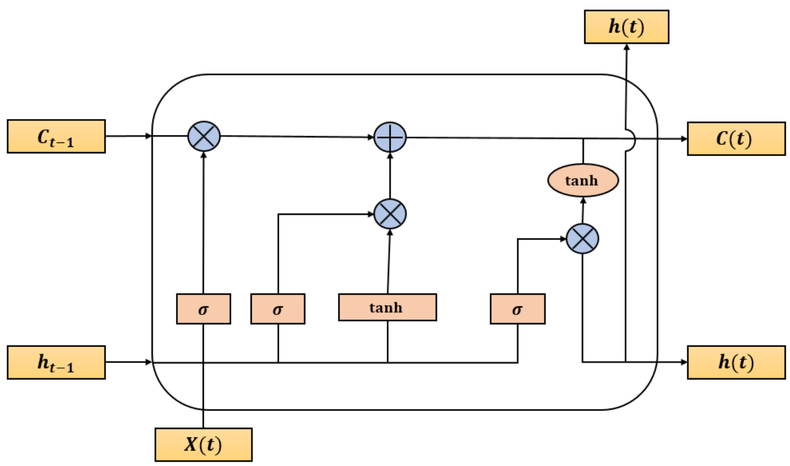

Figure 2.
Structure of LSTM model.

2.2.1.1. Forget Gate Layer

The forget gate is responsible for determining whether the incoming data to the cell should be retained or discarded. The decision to retain the data is evaluated based on the incoming input to that node, along with the results from the previous node's calculations, as shown in the equation (3).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f \quad (3)$$

Where:

f_t represents the forget gate at t.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 10, No. 3: 357-371, 2026
DOI: 10.55214/2576-8484.v10i3.12398

W_f represents the weight matrix between the forget gate and the input gate.
h_{t-1} represents the hidden state of the previous moment.
x_t represents the input of the current moment.
b_f represents connection bias at t.

2.2.1.2. Input Gate Layer

The input gate receives new input data and records it into each node. The operation is divided into two parts. The first part updates the cell state; the input gate decides whether to update the cell state or not. The second part applies the tanh function to the state. If the input gate decides to update the cell state, as shown in the equation (4-6).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$

$$\widetilde{C_t} = tanh\,(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{5}$$

$$C_t = (f_t \cdot C_{t-1}) + (i_t \cdot \widetilde{C_t}) \tag{6}$$

Where:
i_t represents the input gate at t.
W_i represents the hidden layer weights of state vector.
b_i represents respective corresponding doors on the bias vector.
$\widetilde{C_t}$ represents the information candidate state.
W_C represents the hidden layer weights of state vector.
b_C represents respective corresponding doors on the bias vector.
C_t represents the memory cell of the current moment.
C_{t-1} represents the memory cell (cell state) of the previous moment.

2.2.1.3. Output Gate Layer

The output gate controls the flow of information from the memory to the output, considering the state of the cell that has been computed. This value is passed through the tanh function, and the result from the tanh function is used in the next sequence, as shown in the equation (7,8).

$$O_t = \sigma(W_O \cdot [h_{t-1}, x_t] + b_O) \tag{7}$$

$$h_t = O_t \cdot \tanh\,(C_t) \tag{8}$$

Where:
O_t represents the output gate at t.
W_O represents the hidden layer weights of the state vector.
b_O represents the respective corresponding doors on the bias vector.
h_t represents denotes the hidden state of the current moment.

2.3. One-Dimensional Convolutional Neural Network (1D-CNN)

Convolutional Neural Networks (CNNs) are multi-layer neural networks that consist of at least one convolutional layer followed by a pooling layer, and a fully connected layer for deep learning. In the case of time series data, a one-dimensional structure (1D-CNN) can be applied to efficiently extract specific features from data along the time axis, especially for multi-series time series data, which requires considering the relationships of multiple datasets simultaneously over different time periods. The architecture of a 1D-CNN is illustrated in Figure 3. However, while 1D-CNNs excel at extracting spatial features or local patterns, they have limitations in learning long-term temporal dependencies.

Therefore, 1D-CNNs are combined with LSTM models, which have a memory cell structure and gating mechanisms to learn long-term temporal dependencies. This combination allows the model to simultaneously extract spatial features from CNN and learn temporal dynamics from the LSTM model, resulting in higher accuracy in forecasting time series data, such as electrical load [26].

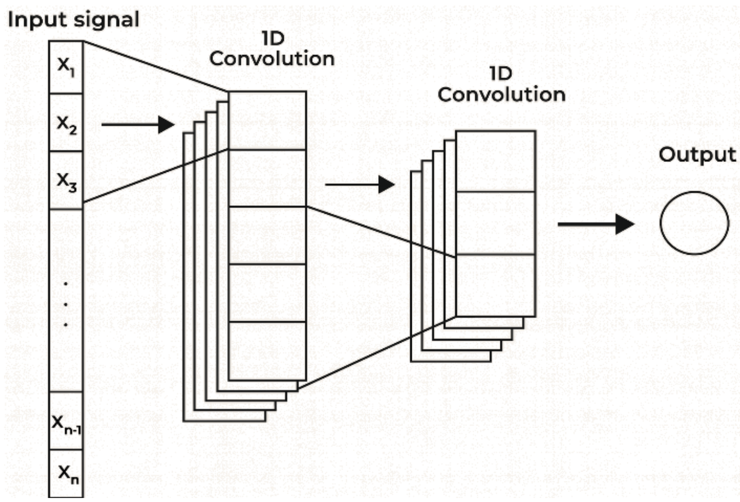

Figure 3.
Structure of One-dimensional convolutional neural network.

2.3.1. Feature Engineering Techniques

In the feature engineering process for time series data analysis, it refers to transforming time-related data into an appropriate format so that the model can learn and utilize temporal patterns and relationships effectively for forecasting. In time series forecasting, there are two important time components: the passage of time and periodicity. Therefore, it is necessary to create features that can reflect both dimensions appropriately before entering the model [27].

2.3.1.1. Lag Features

Lag features involve using past observation values as explanatory variables for the present, where the value at time t is often influenced by previous values, such as t−1, t−2, and so on, called lag 1 and lag 2, respectively. Creating lag features helps models learn temporal dependencies and short-term data dynamics effectively. This technique is employed in both statistical and machine learning models to improve the ability to capture recurring patterns and continuity in data.

2.3.1.2. Moving Average Features

Moving Average Features involve creating features by calculating the average of historical data within a specified time window, such as 3-day or 7-day averages. This method helps reduce short-term volatility (noise) and clarifies the data's trend.

2.3.1.3. Calendar Features

Calendar features are characteristics derived from date and time data to reflect the seasonality of the data, such as days of the week, months, quarters, holidays, or working days. This technique helps the model understand recurring seasonal patterns. For example, in forecasting electricity load, electricity consumption may differ between weekdays and weekends or between summer and winter. Adding calendar features allows the model to capture time-contextual patterns more accurately.

ISSN: 2576-8484
Vol. 10, No. 3: 357-371, 2026
DOI: 10.55214/2576-8484.v10i3.12398

2.4. Evaluation Metrics

Mean Absolute Error (MAE) is the average of absolute errors. It is used to measure model performance and is calculated as the average of the absolute differences between predicted and actual values, as shown in equation (9).

Root Mean Square Error (RMSE) is the square root of the mean of the squared deviations. It is a commonly used statistical measure to evaluate the difference between predicted values and actual values, as shown in equation (10).

R-squared is a statistical measure used to assess how well the forecasting model matches the actual data. It is commonly referred to as the "coefficient of determination." Generally, R^2 ranges from 0 to 1; the closer the value is to 1, the more accurate the model is. Conversely, a value close to 0 indicates that the model is unreliable in predicting the data, as shown in equation (11).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|A_t - F_t| \tag{9}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(A_t - F_t)^2} \tag{10}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(A_t - F_t)^2}{\sum_{i=1}^{n}(A_t - \bar{A}_t)^2} \tag{11}$$

Where:

n is the number of observations

A_t is the actual observed value at time t.

F_t is the forecasted value for the time period or observation at t.

$\bar{A}$ is the average of the actual values in all observations.

2.5. Research Tools

The development and training of the model were conducted on the Google Colaboratory (Google Colab) platform, a cloud-based processing environment supporting the Python language. The main libraries used in model development include TensorFlow/Keras for building neural networks, NumPy and Pandas for data handling, Scikit-learn for data scaling, and Matplotlib for graph visualization.

3. Results and Discussion

3.1. Long Short-Term Memory (LSTM Model)

The comparison of the predicted and actual electricity consumption values is shown in Figure 4. After training, the LSTM model made predictions using the test dataset from August 8, 2022, to December 31, 2022.

DOI: 10.55214/2576-8484.v10i3.12398

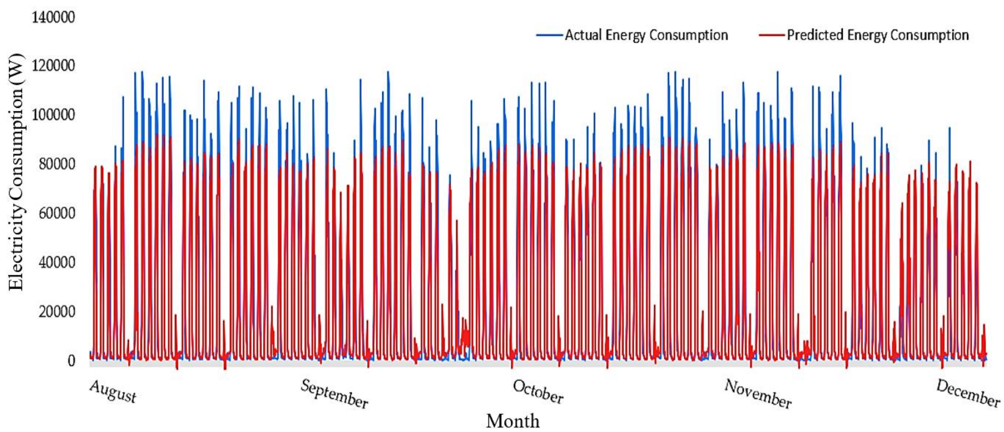

Figure 4.
Time-Series Comparison Between Actual and LSTM-Predicted Electricity Consumption.

The results were transformed back into actual measurement units through the inverse transformation process to ensure they reflect practical accuracy and can be directly compared with actual energy consumption values. The relevant parameters and characteristics of the variables are shown in Table 2 to provide a comprehensive evaluation of the model's performance across multiple dimensions. This research used error metrics including RMSE, MAE, and R^2. The experimental results showed that the LSTM model yielded an RMSE of 13,220.08 W, an MAE of 7,064.33 W, and an R^2 of 0.8578.

Table 2.
Summary of LSTM parameters.

Section	**Details**
Dataset Information	
Total Records	17,520 data sets
Train-Test Split	80:20
Train Data	14,016 data sets
Test Data	3,504 data sets
Data Preprocessing Module	
Cleaning Process	Remove missing and outlier values.
Scaling Method	Min-Max Scaler ([0, 1])
Separate Scalers	Features (X) and Target (y) are scaled independently
Training Config.	
Optimizer	Adam
Learning Rate	0.0005
Loss Function	MSE (Mean Squared Error)
Epochs	50
Batch Size	64
Hyperparameter Tuning (Random Search)	
Learning Rate Candidates	0.01, 0.001, 0.0001
LSTM Units (Layer 1)	32, 64, 128
LSTM Units (Layer 2)	16, 32, 64
Dropout Rate	0.1, 0.2, 0.3
Batch Size Options	32, 64, 128
Epoch Options	10, 20, 50

Additionally, to further study the temporal behavior of the model, Figure 5 shows a comparison between actual and predicted values on a weekly basis, which clearly reveals the cyclical patterns of energy usage, such as the differences between weekdays and weekends.

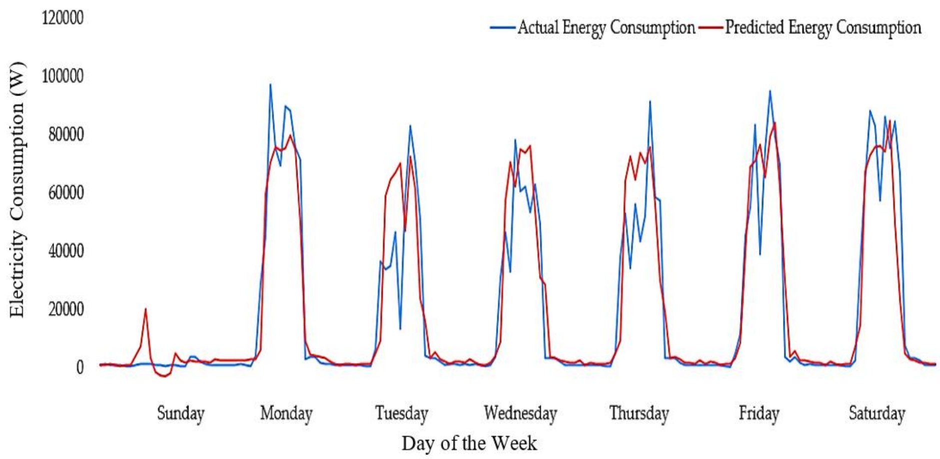

Figure 5.
Weekly Comparison of Actual and LSTM-Predicted Hourly Electricity Consumption.

3.2. Hybrid CNN-LSTM Model

This study focuses on evaluating the performance of the hybrid CNN-LSTM model, which combines the feature extraction capabilities of Convolutional Neural Networks (CNN) and LSTM models. The study identified relevant parameters, with variable characteristics detailed in Table 3.

Table 3.
Summary of Hybrid CNN-LSTM parameters.

Section	Details
Dataset Information	
Total Records	17,520 data sets
Train-Test Split	80:20
Train Data	14,016 data sets
Test Data	3,504 data sets
Feature Engineering Framework	
Lag Features	Energy consumption temperature from previous days.
Calendar Features	Hour of the day (0–23) Day of the week (0–6) Weekend/Holiday indicator (0 = weekday, 1 = weekend /holiday)
Data Preprocessing Module	
Cleaning Process	Remove missing and outlier values.
Scaling Method	Min-Max Scaler ([0, 1])
Separate Scalers	Features (X) and Target (y) are scaled independently
Training Config.	
Optimizer	Adam
Learning Rate	0.0005
Loss Function	MSE (Mean Squared Error)
Epochs	50
Batch Size	64
Hyperparameter Tuning (Random Search)	
Learning Rate Candidates	0.01, 0.001, 0.0001
CNN Filters	32, 64, 128

DOI: 10.55214/2576-8484.v10i3.12398

Section	Details
LSTM Units	50, 100, 150
Dropout Rate	0.1, 0.2, 0.3
Batch Size Options	32, 64, 128
Epoch Options	30, 50, 100

From the performance tests on the test dataset (from August 8, 2022, to December 31, 2022), it was found that the hybrid CNN-LSTM model achieved an RMSE of 11,523.4 W, an MAE of 6,133.34 W, and a coefficient of determination (R^2) of 0.8921. These results demonstrate a significant increase in accuracy compared to the standard LSTM model (RMSE: 13,220.08 W). The R^2 value approaching 1 indicates that the model can explain the variability in electricity consumption data.

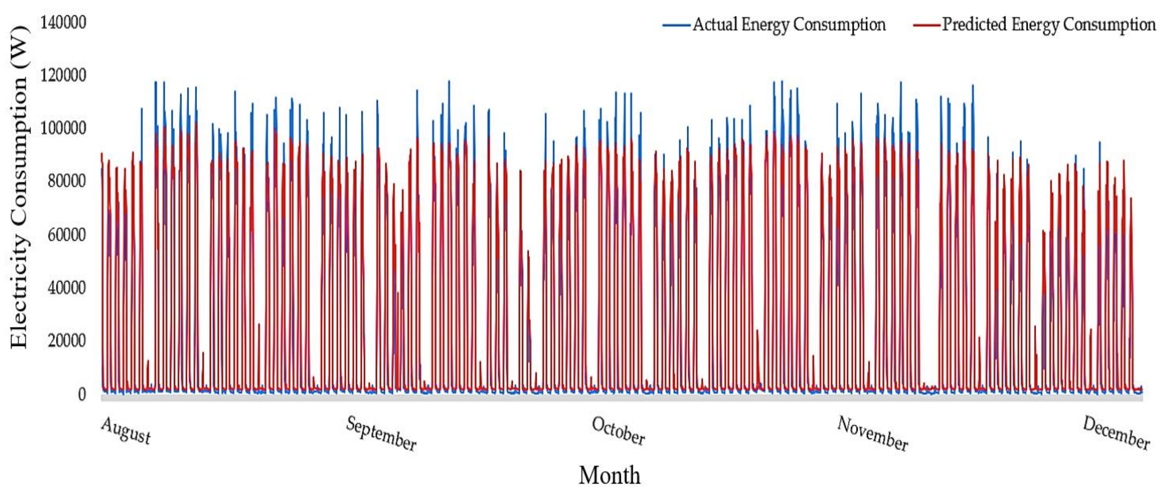

Figure 6.
Time-Series Comparison Between Actual and Hybrid CNN-LSTM Predicted Electricity Consumption.

The comparison between the predicted electricity usage and the actual electricity usage is shown in Figure 6, while Figure 7 displays the weekly forecast results. The simulation results confirm that the model can depict the characteristics of electricity usage behavior on weekdays and weekends.

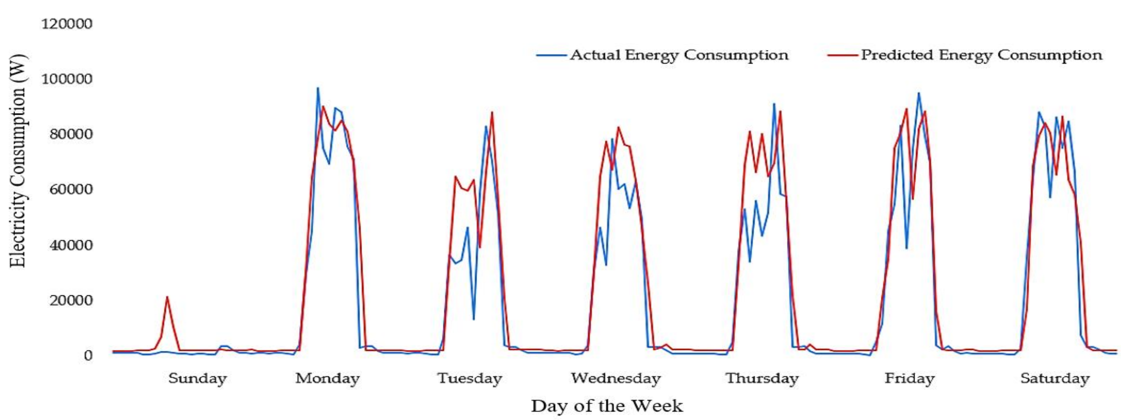

Figure 7.
Weekly Comparison of Actual and hybrid CNN-LSTM Predicted Hourly Electricity Consumption.

3.3. Feature Engineering Lstm (Fe-Lstm Model)

The FE-LSTM model has incorporated the process of temporal feature engineering and calendar feature engineering to enhance the accuracy of electrical load forecasting. The generated features include:

- Lag values at 7, 14, and 30 days reflect both short-term temporal relationships and seasonal patterns.
- 7-day, 14-day, and 30-day backward moving average values help reduce short-term volatility and highlight the main trend of the data. These moving averages emphasize the primary trend of the data.
- Calendar features include the hour of the day, day of the week, and holidays to reflect energy consumption behavior over time. Calendar characteristics include hours of the day, days of the week, and holidays to reflect energy consumption behavior over time.

This study identified the relevant parameters, with details of the variable characteristics shown in Table 4.

The model predicted values from the test dataset (from August 8, 2022, to December 31, 2022) and converted them back to real units using the inverse transformation process. The experimental results showed that the FE-LSTM model had the highest performance, with an RMSE of 8,231.62 W, an MAE of 4,244.48 W, and an R^2 value of 0.9447.

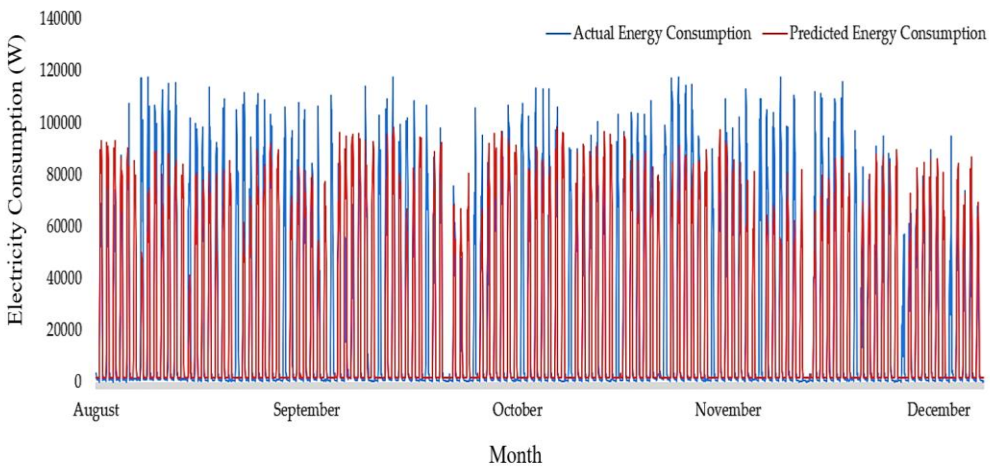

Figure 8.
Time-Series Comparison Between Actual and FE-LSTM Predicted Electricity Consumption.

The comparison of the predicted and actual electricity usage is shown in Figure 8. Additionally, to further study the temporal behavior of the model, Figure 9 shows the comparison between the actual and predicted values on a weekly basis, clearly illustrating the patterns of energy usage behavior, such as the differences between weekdays and weekends.

Table 4.
Summary of FE-LSTM parameter.

Section	Details
Dataset Information	
Total Records	17,520 data sets
Train-Test Split	80:20
Train Data	14,016 data sets
Test Data	3,504 data sets
Feature Engineering Framework	
Objective	To enhance model performance by incorporating temporal and contextual information.
Lag Features (8 features)	Historical consumption and temperature values at 7, 14, and 30 days before the target date.
Moving Average Features (6 features)	Consumption MA [7,14,30] and Temp MA [7,14,30] to smooth fluctuations and reduce short-term noise.
Calendar Features	Hour of the day (0–23) Day of the week (0–6) Weekend/Holiday indicator (0 = weekday, 1 = weekend /holiday)
Data Preprocessing Module	
Cleaning Process	Remove missing and outlier values.
Scaling Method	Min-Max Scaler ([0, 1])
Separate Scalers	Features (X) and Target (y) are scaled independently
Training Config.	
Optimizer	Adam
Learning Rate	0.0005
Loss Function	MSE (Mean Squared Error)
Epochs	50
Batch Size	64
Hyperparameter Tuning (Random Search)	
Learning Rate Candidates	0.01, 0.001, 0.0001
LSTM Units	50, 100, 150
Dropout Rate	0.1, 0.2, 0.3
Batch Size Options	32, 64, 128
Epoch Options	30, 50, 100

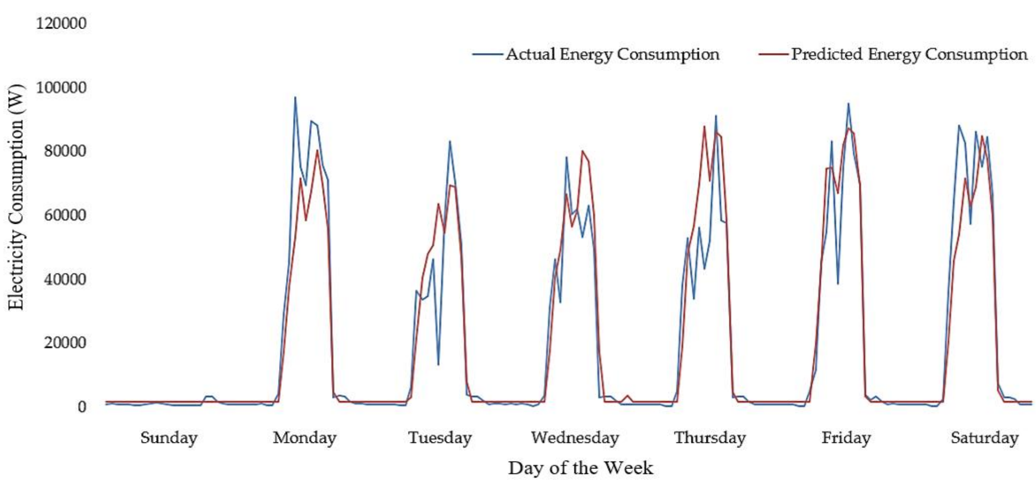

Figure 9.
Weekly Comparison of Actual and FE-LSTM Predicted Hourly Electricity Consumption.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 10, No. 3: 357-371, 2026
DOI: 10.55214/2576-8484.v10i3.12398

3.4. Model Comparison

To systematically evaluate the performance of each model, the results of the standard LSTM model, the hybrid CNN-LSTM model, and the FE-LSTM model were compared using error metrics including RMSE and MAE, as well as the coefficient of determination (R^2) to analyze the forecasting accuracy in real-world electrical load data. Table 5 shows the performance comparison of the three models, which found that adding temporal and calendar features to the FE-LSTM model improved forecasting performance compared to the other models.

Table 5.
Performance Comparison of LSTM, Hybrid CNN–LSTM, and FE-LSTM Models.

Model	RMSE (W)	MAE (W)	R^2
LSTM	13,220.08	7,064.33	0.8578
Hybrid CNN-LSTM	11,523.40	6,133.34	0.8921
FE-LSTM	8,231.62	4,244.48	0.9447

The data in Table 5 shows that the FE-LSTM model has the highest performance with the lowest RMSE value of 8,231.62 W, indicating that feature engineering properties enable the model to comprehensively learn energy usage patterns compared to other methods. Meanwhile, the hybrid CNN-LSTM model (RMSE: 11,523.40 W) performs better than the standard LSTM model because the structure of the CNN-LSTM model can effectively extract spatial features, but it still provides lower forecasting accuracy than the FE-LSTM model. From the simulation results, it can be concluded that feature creation in the FE-LSTM model enhances load forecasting efficiency by 37.7% compared to the baseline model. Therefore, the FE-LSTM model is a highly accurate method suitable for predicting actual electricity usage in the industrial sector.

4. Conclusions

This paper presents enhancing LSTM prediction accuracy through hybrid feature engineering: an LSTM-based model for load forecasting of industry in Thailand. The study found that the standard LSTM model, while capable of learning temporal relationships well, showed that electricity consumption behavior is significantly correlated with temporal and environmental factors. To address these limitations, this research introduced the FE-LSTM model, which incorporates feature engineering processes such as lag features, moving averages, and calendar features into the LSTM structure. The addition of temporal and calendar features allows the model to comprehensively reflect energy consumption behavior. The experimental results show that the FE-LSTM model is the most effective, reducing the RMSE error from 13,220.08 W (standard LSTM model) to only 8,231.62 W, which represents a 37.7% improvement in the accuracy of electrical load forecasting. When compared to the hybrid CNN-LSTM model, although the hybrid CNN-LSTM model can effectively extract sequential features better than the LSTM model, it still yields lower forecasting accuracy than the FE-LSTM model.

In summary, the FE-LSTM model incorporates feature engineering processes such as lag features, moving averages, and calendar features, which are factors that represent electricity consumption behavior, into the LSTM structure. The feature engineering process is highly effective, allowing for the capture of relationships in electricity consumption behavior throughout the day. The differences in electricity usage on weekdays and weekends enhance the accuracy of electricity consumption forecasts, making it suitable for short-term load forecasting in the industrial sector.

Transparency:

The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.

References

[1] International Energy Agency, *Global energy review 2025*. Paris, France: IEA, 2025.

[2] F. J. Zarco-Soto, I. M. Zarco-Soto, S. S. S. Ali, and P. J. Zarco-Perinan, "Energy consumption in buildings: A compilation of current studies," *Energy Reports*, vol. 13, pp. 1293-1307, 2025. https://doi.org/10.1016/j.egyr.2024.12.069

[3] H. Terbrack, T. Claus, and F. Herrmann, "Energy-oriented production planning in industry: A systematic literature review and classification scheme," *Sustainability*, vol. 13, no. 23, p. 13317, 2021. https://doi.org/10.3390/su132313317

[4] A. Ortiz-Peña, A. Honrubia-Escribano, and E. Gómez-Lázaro, "Electricity consumption and efficiency measures in public buildings: A comprehensive review," *Energies*, vol. 18, no. 3, p. 609, 2025. https://doi.org/10.3390/en18030609

[5] C. Ragupathi, S. Dhanasekaran, N. Vijayalakshmi, and A. O. Salau, "Prediction of electricity consumption using an innovative deep energy predictor model for enhanced accuracy and efficiency," *Energy Reports*, vol. 12, pp. 5320-5337, 2024. https://doi.org/10.1016/j.egyr.2024.11.018

[6] R. Morcillo-Jimenez, J. Mesa, J. Gómez-Romero, M. A. Vila, and M. J. Martin-Bautista, "Deep learning for prediction of energy consumption: An applied use case in an office building: R. Morcillo-Jimenez et al," *Applied Intelligence*, vol. 54, no. 7, pp. 5813-5825, 2024. https://doi.org/10.1007/s10489-024-05451-9

[7] T. C. Brito and M. A. Brito, "Forecasting of energy consumption: Artificial intelligence methods," presented at the 17th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-4). IEEE, 2022.

[8] N. A.-H. K. Hussein and M. b. Abouessaouab, "Optimizing energy efficiency in smart grids using machine learning algorithms: A case study in electrical engineering," *Khwarizmia*, vol. 2023, pp. 113-120, 2023. https://doi.org/10.70470/KHWARIZMIA/2023/011

[9] W. Softah, L. Tafakori, and H. Song, "Analyzing and predicting residential electricity consumption using smart meter data: A copula-based approach," *Energy and Buildings*, vol. 332, p. 115432, 2025. https://doi.org/10.1016/j.enbuild.2025.115432

[10] L. M. N. Gabriel, J. A. Adebisi, L. N. P. Ndjuluwa, and D. K. Chembe, "Investigation of smart grid technologies deployment for energy reliability enhancement in electricity distribution networks," *Franklin Open*, vol. 10, p. 100227, 2025. https://doi.org/10.1016/j.fraope.2025.100227

[11] K. Zor, O. Timur, Ö. Çelik, H. Yıldırım, and A. Teke, "Interpretation of error calculation methods in the context of energy forecasting," presented at the In 12th Conference on Sustainable Development of Energy, Water and Environment Systems, Dubrovnik, Croatia (Vol. 722, pp. 1-9), 2017.

[12] K. Misiurek, T. Olkuski, and J. Zyśk, "Review of methods and models for forecasting electricity consumption," *Energies*, vol. 18, no. 15, p. 4032, 2025. https://doi.org/10.3390/en18154032

[13] S. S. W. Fatima and A. Rahimi, "A review of time-series forecasting algorithms for industrial manufacturing systems," *Machines*, vol. 12, no. 6, p. 380, 2024. https://doi.org/10.3390/machines12060380

[14] S. Tulli, "Comparative analysis of traditional and AI-based demand forecasting models," *International Journal of Emerging Trends in Science and Technology*, vol. 7, no. 03, pp. 6842-6847, 2020.

[15] F. Hosseini, C. Prieto, and C. Álvarez, "Hyperparameter optimization of regional hydrological LSTMs by random search: A case study from Basque Country, Spain," *Journal of Hydrology*, vol. 643, p. 132003, 2024. https://doi.org/10.1016/j.jhydrol.2024.132003

[16] A.-L. Toba, S. Kulkarni, W. Khallouli, and T. Pennington, "Long-term traffic prediction using deep learning long short-term memory," *Smart Cities*, vol. 8, no. 4, p. 126, 2025. https://doi.org/10.3390/smartcities8040126

[17] R. B. Preeti and R. P. Singh, "Financial and non-stationary time series forecasting using LSTM recurrent neural network for short and long horizon," presented at the Proceedings of the 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 2019.

[18] D. Mailepessov, J. Aik, and W. J. Seow, "A time series analysis of the short-term association between climatic variables and acute respiratory infections in Singapore," *International Journal of Hygiene and Environmental Health*, vol. 234, p. 113748, 2021. https://doi.org/10.1016/j.ijheh.2021.113748

[19] P. Zeng and M. Jin, "Peak load forecasting based on multi-source data and day-to-day topological network," *IET Generation, Transmission & Distribution*, vol. 12, no. 6, pp. 1374-1381, 2018. https://doi.org/10.1049/iet-gtd.2017.0201Digital

[20] M. Borunda *et al.*, "An intelligent method for day-ahead regional load demand forecasting via machine-learning analysis of energy consumption patterns across daily, weekly, and annual scales," *Applied Sciences*, vol. 15, no. 9, p. 4717, 2025. https://doi.org/10.3390/app15094717

[21] J. Son, J. Cha, H. Kim, and Y.-M. Wi, "Day-ahead short-term load forecasting for holidays based on modification of similar days' load profiles," *IEEE Access*, vol. 10, pp. 17864-17880, 2022. https://doi.org/10.1109/ACCESS.2022.3150344

[22] X. Zhu, J. Xu, Z. Fu, S. K. Damarla, P. Wang, and K. Hao, "Novel dynamic data-driven modeling based on feature enhancement with derivative memory LSTM for complex industrial process," *Neurocomputing*, vol. 626, p. 129619, 2025. https://doi.org/10.1016/j.neucom.2025.129619

[23] Y. Ouyang, S. Xie, and Y. Xie, "A short-term feature-enhanced lstm-based method for predicting quality variables in industrial processes," presented at the In 2025 37th Chinese Control and Decision Conference (CCDC) (pp. 731-736). IEEE, 2025.

[24] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, "Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches," *Energies*, vol. 11, no. 7, p. 1636, 2018. https://doi.org/10.3390/en11071636

[25] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Transactions on Smart grid*, vol. 10, no. 1, pp. 841-851, 2017. https://doi.org/10.1109/TSG.2017.2753802

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

[27] M. Joseph, *Modern time series forecasting with python: Explore industry-ready time series forecasting using modern machine learning and deep learning*. Birmingham, U.K: Packt Publishing, 2022.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 10, No. 3: 357-371, 2026
DOI: 10.55214/2576-8484.v10i3.12398