iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VII **Month of publication:** July 2026

DOI: https://doi.org/10.22214/ijraset.2026.84342

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

Data Driven Electrical Load Forecasting: A Comapartive Study of Normalization Techniques

Kuldeep Atmdarshi[1], Seema Pal[2]
Electrical Engineering Department, Jabalpur Engineering College, Jabalpur, India

Abstract: Short-term electrical load forecasting is of great importance for power system operation, efficient energy management, and planning.
For better performance of deep learning models for short-term load forecasting, the quality of the data must be high. In the preprocessing stage, data normalization plays an important role.
The present study performs the analysis and comparison of various data normalization techniques, namely Min-Max Scaling, Mean Normalization, Z-Score Standardization, and Gaussian Normalization, for the forecasting of short-term electrical load using an LSTM model. The performance metrics Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R^2) are used to compare the performance of the normalization methods. The results show the impact of normalization techniques on the load prediction performance of the LSTM model.
Keywords: LSTM, Load Forecasting, Normalization, Min-Max Scaling, Z-Score, Gaussian Transformation, Time Series.

I. INTRODUCTION

Electricity demand forecasting is a vital aspect in the planning, operation, and administration of contemporary power systems. Accurate short-term load forecasting helps utility companies schedule power generation efficiently, preserve system dependability, reduce operational costs, and eliminate unnecessary reserve capacity[1]. Inaccurate projections may lead to either over-generation, resulting in increased operational expenses, or under-generation, which threatens grid stability and may cause power shortages.

In the country, population growth, industrial development, urbanisation, and the extensive use of electrical equipment have led to an increasing demand for electricity. Developing country is the largest electricity-consuming region and accounts for the majority of electrical consumption. The widespread use of cooling systems in the country's tropical climate has a significant impact on electricity consumption patterns, as ambient temperature is a major weather-related factor. In general, higher temperatures increase the use of air-conditioning systems, leading to higher electricity demand [2]. Furthermore, electricity load demand demonstrates strong daily and weekly seasonal trends due to variations in residential, commercial, and industrial activities across different times of the day and days of the week [3].

These characteristics make short-term electricity load forecasting a challenging task. Traditional statistical forecasting approaches, such as linear regression, autoregressive integrated moving average (ARIMA), and exponential smoothing, have been widely used for load forecasting.

However, these methods may have limited ability to capture complex nonlinear relationships and temporal dependencies in electricity demand data [4].

Recently, deep learning algorithms have gained considerable attention because of their effectiveness in modelling nonlinear and sequential patterns. Among these techniques, Long Short-Term Memory (LSTM) networks have demonstrated excellent performance in time-series forecasting applications [5]. It is a specialised type of recurrent neural network that can learn long-term dependencies from historical data while overcoming the vanishing gradient problem commonly found in conventional recurrent neural networks [6]. Since electricity demand depends heavily on historical load values, time-related variables, and weather conditions, LSTM is particularly well suited for short-term load forecasting tasks [7].

II. METHODOLOGY

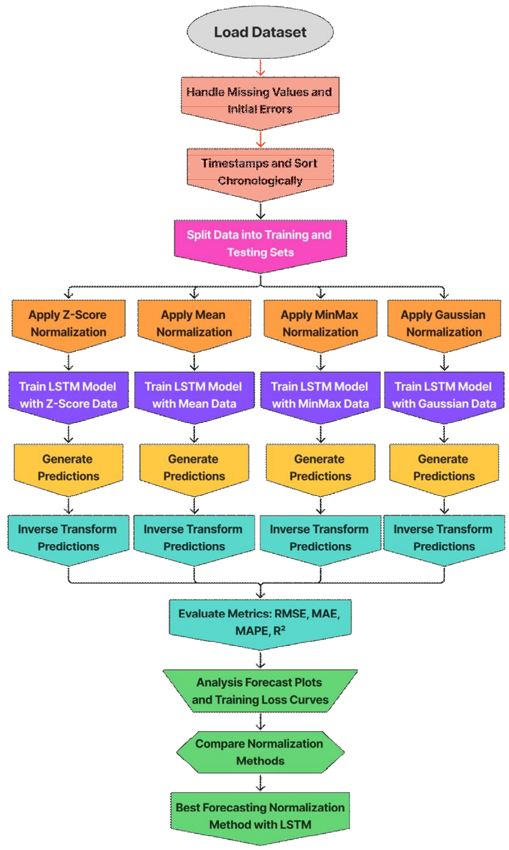

Fig. 1 Methodology of Model

This paper presents short-term hourly electricity load forecasting using of different normalization techniques for LSTM networks for shown in fig. 1. The dataset consists of hourly electricity load demand and temperature data collected of 1-year historical data. To improve the predictive capability of the model, additional time-related features such as hour of the day, day of the week, and lagged load variables are included as input features. Furthermore, multiple data normalisation methods are investigated to evaluate their impact on forecasting performance. The primary objective of this study is to compare the performance of four normalisation techniques—Min-Max normalisation, Mean normalisation, Z-score normalisation, and Gaussian function normalisation—when used with an LSTM forecasting model. The performance of each normalisation method is evaluated using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). Through this comparison, the study aims to identify the most suitable normalisation technique for improving the forecasting accuracy of LSTM models in short-term electricity load prediction.

A. Data Description

The dataset utilised for this analysis is hourly energy load demand data for one year which includes the date and time stamp, ambient temperature, and the load demand in megawatts (MW). The timestamp variable contains useful time information which may be used to calculate other properties such as hour of day, day of week, month, and whether the day is a weekend or weekday. These temporal indicators are especially important since electricity demand patterns tend to follow the human activity schedules, working hours and seasonal behaviour [8].

The ambient temperature is included as an exogenous variable due to its large effect on electricity usage in tropical countries. Hot and humid weather usually mean more usage of air-conditioning systems, refrigeration equipment and other cooling appliances, which leads to higher electricity demand. On the other hand, lower temperatures may lead to fewer cooling demands and hence lower system load. Thus, temperature is anticipated to have a substantial positive association with electricity consumption [9].

The load profile of the power in the dataset has specific cyclical properties, especially daily and weekly seasonality. The daily seasonality is due to the fact that electricity consumption is often lower during the late-night hours and greater during business hours and the evening hours. Weekly seasonality is also observed as electricity demand on weekdays is usually different from that on weekends due to variations in industrial, commercial and residential activity [10]. Forecasting model incorporates time related variables such as hour of day and day of week to better capture these repeating trends. These qualities enable the model to learn the regular periodic behaviour of electricity demand [11].

Moreover, load factors with time lags are considered as input characteristics since present power demand is typically largely influenced by past demand values. For example, the power load in a certain hour could be strongly correlated with the load in the previous hour, the same hour in the previous day, or the same hour in the previous week [12].

Including such lagged variables can considerably enhance forecasting accuracy, since it allows the model to capture temporal interdependence and autocorrelation in the load series.

B. Data Preprocessing

The raw hourly electricity load and ambient temperature data were first subjected to a data-cleaning process to ensure consistency, reliability, and suitability for model training. During the initial inspection, missing values and anomalous observations were identified in both the electricity load and temperature series. Missing data may arise due to sensor failures, communication errors, or incomplete data recording. If not properly addressed, these issues can negatively affect the performance and accuracy of forecasting models [13].

Linear interpolation was employed to replace the missing values. This technique estimates each missing observation by fitting a straight line between the nearest preceding and succeeding valid observations. Linear interpolation is widely used in time-series analysis because it preserves data continuity and maintains the overall trend without introducing abrupt changes into the dataset [14].

Then, the outliers and anomalous observations were recognised by utilising the Interquartile Range (IQR) approach. In this method, the first quartile (Q_1) and third quartile (Q_3) of data distribution are calculated and the interquartile range is determined as:

$$IQR = Q_3 - Q_1 I \qquad \dots (1)$$

Outliers were defined as observations below Q_1-1.5×IQR or above Q_3+1.5×IQR. To avoid distortion of the learning process of the forecasting model, the aberrant values were eliminated or replaced using interpolation [15].

C. Normalization Methods

Before training the LSTM model, we normalised the input data to ensure that all characteristics were evaluated on the same scale. Our dataset contains variables with different units and scales, such as temperature and power demand. If the raw values are fed directly into the network, features with larger numerical values can dominate the learning process, leading to unstable weight updates, slower training, and reduced prediction accuracy. Standardising the data enables the LSTM to learn more effectively and helps maintain numerical stability during training[16] .

We examined and compared four alternative normalisation methods for the input variables: Min-Max, Mean, Z-score, and Gaussian normalisation, to determine the most suitable approach. Each method transforms the raw data differently and can significantly influence the model's convergence speed, handling of outliers, and predictive performance.

In this paper, we compare all four normalisation techniques and identify the method that provides the best prediction performance when used with the LSTM model.

1) Min–Max Normalization

Min-max normalization is a data preparation method that rescales the data features to a specified range, from 0 to 1. It is frequently used in machine learning and deep learning models to ensure that the variables with varying units or magnitudes contribute equally during the training process and do not bias the learning process towards larger-valued features [17].

The normalized value x' of an original data point x is computed as:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \qquad \dots (2)$$

where:

x is the original value,
x_{min} is the minimum value in the dataset,
x_{max} is the maximum value in the dataset,
x' is the normalized value.

After Min-Max normalisation, all values are scaled to the interval $[0, 1]$. In some circumstances, the procedure can be extended to transform values into another range, such as $[-1, 1]$, depending on the requirements of the learning algorithm. It preserves the relationships among the original data values and is widely used in neural networks and LSTM-based load forecasting because appropriate feature scaling improves convergence and training stability[18].
However, Min-Max normalisation is sensitive to outliers. Since the transformation depends directly on the minimum and maximum values of the dataset, extreme observations can compress most of the data into a narrow interval, reducing the model's ability to learn meaningful patterns [19].

2) Mean Normalization

Feature scaling techniques are used to modify numerical variables to be centred around zero and usually have a small range. Mean normalization repositions the data around the feature's mean, and its range, whereas min-max normalization scales values to lie within predefined ranges. This strategy is widely used in machine learning and statistical analysis, to increase convergence and to ensure that variables with varied scales contribute evenly to the model [20].
The formula for mean normalization is:

$$x' = \frac{(x-\bar{x})}{(x_{max}-x_{min})} \qquad \dots (3)$$

where:
- x' is the original value,
- x is the mean of the feature values,
- x_{max} is the maximum value in the dataset,
- x_{min} is the minimum value in the dataset,
- x' is the normalized value.

The mean normalisation of the transformed data sets all the data to be around zero, where most of the data are in the range $[-1,1]$ based on the distribution of the original dataset. Mean normalisation is particularly useful when you wish to standardise variables but yet want to maintain the relative distances between observations. It is often employed in optimization-based machine learning methods such as gradient descent because it reduces numerical instability and accelerates convergence during model training.
Appropriately centred input data can improve the model learning efficiency and performance of deep learning based load forecasting .
But the difficulty with mean normalisation is that it is sensitive to outliers in the data set. As the scaling is based on the range (x_{max}-x_{min}), outliers can cause most normalised values to cluster around zero. In this case, other approaches such as z-score normalisation could be a more robust solution as it uses standard deviation instead of the range [21].

3) Z-Score Normalization

Z-score normalisation, often called as standardisation, is a feature scaling method which scales the data such that the mean is 0 and standard deviation is 1. It is popularly employed in statistical analysis and machine learning, especially in the case when data contains outliers or methods assume input features are regularly distributed [22].

$$z = \frac{x-\mu}{\sigma} \qquad \dots (4)$$

where:
- x is the original value,
- μ is the mean of the dataset,
- σ is the standard deviation of the dataset,
- z is the standardized value.

After normalisation, positive z-scores indicate values above the mean, and negative z-scores indicate values below the mean. Z = 0 means the observation is equal to the mean. Z-score normalisation is very beneficial for machine learning methods such as logistic regression, support vector machines, principal component analysis, and neural networks. Compared with min–max normalisation, it is less sensitive to outliers because it is based on the mean and standard deviation instead of extreme values. In deep learning based load forecasting, standardisation has been demonstrated to improve both training stability and model generalisation performance [23].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

4) Gaussian Function Normalization

Gaussian function normalisation is a scaling method that uses a Gaussian (normal distribution) function to transform data so that the transformed data are between 0 and 1. This approach is particularly useful for data that are nearly normally distributed and when a smooth nonlinear scaling is sought[24].

The Gaussian normalization formula is:

$$x' = e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad \dots (5)$$

where:

x is the original value,
μ is the mean of the dataset,
σ is the standard deviation,
x' is the normalized value.

The effect of this transformation is to translate values adjacent to the mean to values near to 1 and values further from the mean to values approaching 0. This attribute allows for the Gaussian normalisation to emphasise the central data points and de-emphasise the impact of the outliers. Therefore, it is frequently utilised in pattern recognition, signal processing, fuzzy systems and neural network based models. A key benefit of Gaussian normalisation is that it delivers a smooth, continuous transformation while reducing the effect of outliers. However, the performance of the test hinges on the assumption that the underlying data are (or approximately are) normally distributed. If the data is excessively skewed the transformation may not work well [25].

D. Model Architecture

The data set was split into two parts, 80% of the data was used to train the LSTM model and the other 20% was used to test and evaluate the model's anticipated performance. This train-test split enables the model to learn the basic patterns from a large percentage of the data and to be checked on unseen samples to evaluate its generalization capabilities [26]. Such a partitioning strategy is a popular practice in time series forecasting research to prevent overfitting and to provide a credible evaluation of model performance. The input features were normalized using four different normalization processes before the sequence data was fed into the LSTM model. Normalization is a pre-processing step necessary for time-series and sequence modelling as it scales data into a consistent numerical range, minimizes inequalities among the size of features and increases the convergence speed and stability of the training process [27]. In deep learning-based forecasting models, suitably scaled inputs stabilize the gradient updates and increase the overall training efficiency. Several normalization methods can be used to undertake a comparison analysis to determine which scaling strategy works best for the LSTM model. Different normalization procedures could have an impact on the ability of the model to learn temporal dependencies and nonlinear interactions in the data. Therefore, it is necessary to analyze several preprocessing techniques to select the most efficient normalization method to increase the forecasting accuracy and robustness for LSTM load forecast model [28].

III. LONG SHORT-TERM MEMORY (LSTM)

Long Short Term Memory (LSTM) is a particular form of Recurrent Neural Network (RNN) model, which is meant to model sequential and time series data. Unlike standard neural networks, LSTM networks can learn temporal connections by remembering important information from prior time-steps. This makes LSTM particularly suited for energy load forecasting because future demand is highly dependent on past load patterns.

A major disadvantage of typical RNNs is the vanishing gradient problem, which limits the network's capacity to learn long-term dependencies. This problem is addressed by LSTM due to its memory cell structure and gating mechanisms that control information flow [29].

An LSTM cell contains three gates:

Forget gate
Input gate .
Output gate

The forget gate determines which information from the previous cell state should be discarded.

$$ft = \sigma(Wf[ht-1, xt] + bf) \quad \dots (6)$$

The input gate determines which new information should be stored in the cell state.

$$it = \sigma(Wi[ht-1, xt] + bi) \quad \dots (7)$$

The candidate cell state is calculated as:

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad \ldots (8)$$

The updated cell state is then computed by combining the forget gate and input gate outputs.

$$Ct = ft * Ct - 1 + it * Ct\sim \quad \ldots (9)$$

The output gate determines the final hidden state of the LSTM cell.

$$ot = \sigma(Wo[ht - 1, xt] + bo) \quad \ldots (10)$$

Finally, the hidden state is calculated as:

$$ht = ot * tanh\ (Ct) \quad \ldots (11)$$

where:

x_t is the input at time step t,
h_{t-1} is the previous hidden state,
C_{t-1} is the previous cell state,
W, b represents weight matrices and bias vectors,
σ: denotes the sigmoid activation function,
tanh: denotes the hyperbolic tangent activation function.

LSTM networks are commonly used for load forecasting, as they can efficiently capture the short- and long-term relationships in the electricity consumption patterns. They are ideally suited for simulating daily, weekly and seasonal fluctuations. They also fit modelling the influence of external variables such as temperature [30].

IV. EVALUATION METRICS

To evaluate, four statistical measurement methods were employed to evaluate the performance of the LSTM model with different normalization methods: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and coefficient of determination (R^2) [31].

A. Root Mean Square Error (RMSE)

One of the evaluation metrics used for regression and forecasting models is the Root Mean Square Error (RMSE). This is the square root of the average squared disparities between the anticipated values and the actual values . This indicates the average magnitude of the errors in a set of predictions . Because the errors are squared before they are averaged, RMSE lends relatively more weight to larger errors. This makes it particularly beneficial when huge predicting errors are undesirable.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad \ldots (12)$$

where:

n is the total number of observations,
y_i is the actual value,
$\hat{y}_i$ is the predicted value.

A lower value of RMSE implies a higher model performance as the predicted values are closer to the actual values. The RMSE is particularly ideal for the electrical load forecasting since it would penalise more large deviations than smaller ones. This is crucial for the power system operations as large forecasting errors might result in inefficient energy generation planning and higher operating expenses.

The root mean square error (RMSE) is a measure of the average magnitude of the error, where higher error has more weight due to the square operation. Lower RMSE indicates greater model performance. It plays an important role in the field of electrical load forecasting where big inaccuracies may have huge impact on system operations [32].

B. Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is a commonly used evaluation statistic for regression and forecasting models. It measures the average absolute difference between the actual and anticipated values, without considering the direction of errors. MAE unlike RMSE does not square the mistakes and hence all errors in prediction contribute equally to the final statistic.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad \ldots (13)$$

where:

n is the total number of observations,

y_i is the actual value,

$\hat{y}_i$ is the predicted value.

Since the average error of prediction is smaller with a smaller value of MAE, the model performs better for a smaller value of MAE. The MAE is especially valuable since it can be easily interpreted and directly measures the average forecasting error in the same unit as the target variable, such as megawatts (MW) in electricity load forecasting.

MAE is the average of the absolute differences between the values you actually saw and the values your model projected. All errors are treated equally and it offers an easily interpretable measure of prediction accuracy in the same unit as the target variable [33].

C. Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) is a commonly used metric for evaluating forecasting accuracy. It measures the average absolute percentage difference between the actual values and the predicted values. Because the error is expressed as a percentage, MAPE is easy to interpret and allows comparison of forecasting performance across different datasets or scales.

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i-\hat{y}_i}{y_i}\right| \qquad \ldots (14)$$

where:

n is the total number of observations,

y_i is the actual value,

$\hat{y}_i$ is the predicted value.

When analysing forecasting models, one always seeks a lower Mean Absolute Percentage Error (MAPE). It offers a clear, easy-to-understand measure of accuracy to it evaluates the mean percentage variation between the prediction and the actual outcome.

This makes MAPE extremely useful for power demand forecasting, since it allows a direct comparison of performance between completely distinct data sets. However, it does have one important Achilles' heel: if the real baseline values decrease too near to zero, the metric can become exceedingly unstable and biased [34].

D. Coefficient of Determination (R^2)

R^2 is the coefficient of determination . It is a statistical measure of how well the model explains the variability in the observed data . This is a measure of the variance of actual values divided by the variance of predicted values from the model.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2} \quad \ldots (15)$$

where:

y_i is the actual value,

$\hat{y}_i$ is the predicted value,

$\bar{y}$ is the mean of the actual values,

n is the total number of observations.

R^2 can take values between 0 and 1, and can be negative if the model performs worse than predicting the mean. The closer the value is to 1 the better the prediction performance of the model as it explains a substantial amount of the variability in the data. An R^2 value of 1 is a perfect prediction, whereas a value of 0 indicates that the model explains none of the variation in the target variable.

R squared is the amount of variation explained by the model in the observed data. A value closer to 1 means greater prediction accuracy while a value nearer to 0 means weak explanatory power [35].

V. RESULTS AND DISCUSSION

In order to find the optimal preprocessing step for our power load predictions, we tried an LSTM model with four alternative normalisation techniques: Min-Max, Mean, Z-score and Gaussian function normalisation. In all experiments, we had a rigorous 80% training and 20% testing split to keep the playing field totally level.

The study compares the performance of several data normalisation strategies on Long Short-Term Memory (LSTM) neural networks for electrical load forecasting. For each normalisation approach we trained an LSTM model with identical architecture. The evaluation metrics included Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2), coupled with visual examination of the forecast plots.

A. *Quantitative Results*

The comparative results of the four normalization methods show that the choice of normalization technique has a significant impact on the forecasting performance of the LSTM model.

Table -1

Normalization Method	RMSE	MAE	MAPE (%)	R^2
Min-Max Normalization	4523.25	3219.09	7.8	0.8984
Mean Normalization	4509.01	3184.56	7.68	0.899
Z-Score Normalization	4508.86	3241.09	7.81	0.899
Gaussian Normalization	3810.52	2789.3	5.25	0.693

B. *Performance Analysis*

From the experiment of this model Min-Max, Mean and Z-Score Normalizations performed well and relatively similarly (high R^2 around 0.89-0.90, and similar RMSE, MAE, MAPE), visually portraying load patterns well. The tailored Gaussian Normalization was a special case. While it had the lowest RMSE, MAE and MAPE (e.g. RMSE of ~3810 versus ~4500 for others), its R^2 was much lower (0.6930) and visual forecasts had severe distortion.

The loss of information and approximate inverse transformation degrades the reconstruction of the original data. In this context Gaussian normalization approach is less suitable for the applications where exact reconstruction of original values (as in forecasting) is crucial.

The optimum normalization method was selected according to the assessment criteria. Min-Max, Mean and Z-Score demonstrated good performance with high R^2 values of approximately 0.89–0.90 and difference become negligible of RMSE, MAE and MAPE values. Gaussian showed smaller errors but substantially lower R^2 of 0.6930. It also led to information loss due to its many-to-one mapping, which created distorted results after the inverse translation.

C. *Visual Performance*

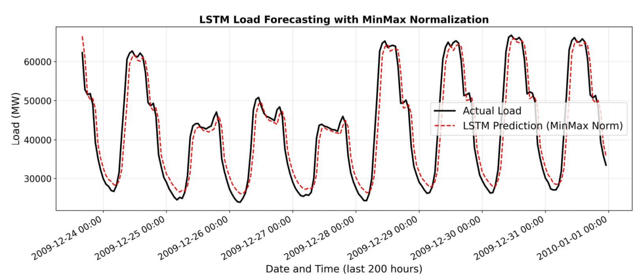

Fig. 2 LSTM load forecasting with Min-max Normalization

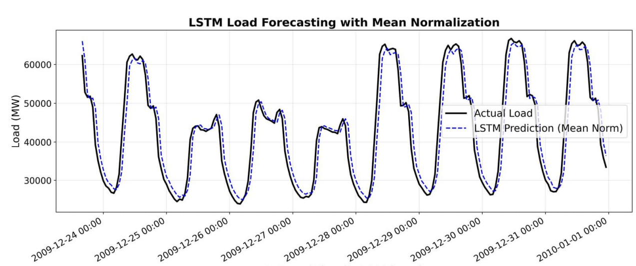

Fig. 3 LSTM load forecasting with Mean Normalization

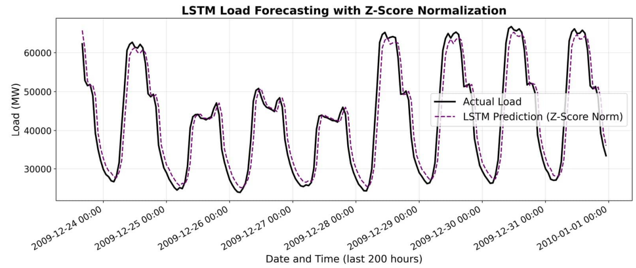

Fig. 4 LSTM load forecasting with Z Score Normalization

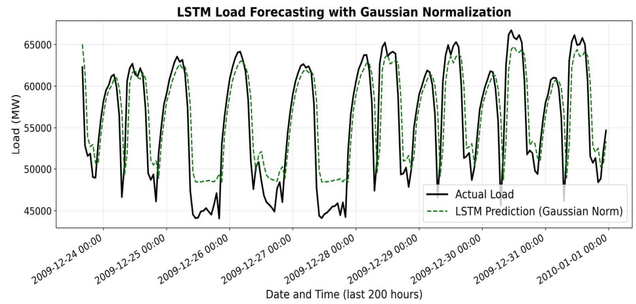

Fig. 5 LSTM load forecasting with Gaussian Function Normalization

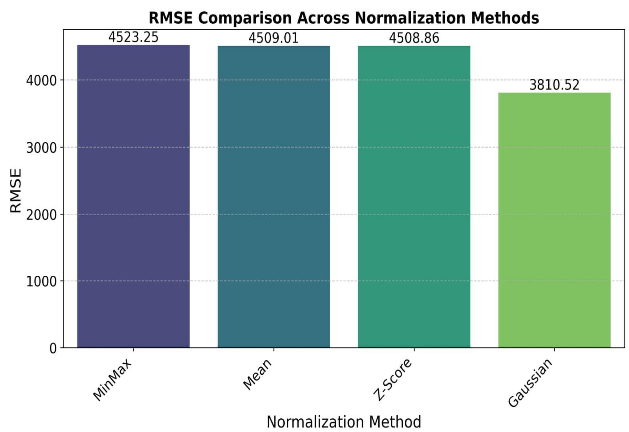

Fig. 6 RMSE Comparison Across Normalization Methods

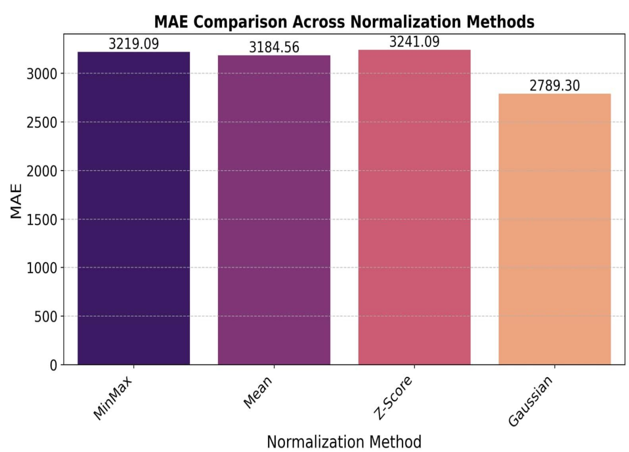

Fig. 7 MAE Comparison Across Normalization Methods

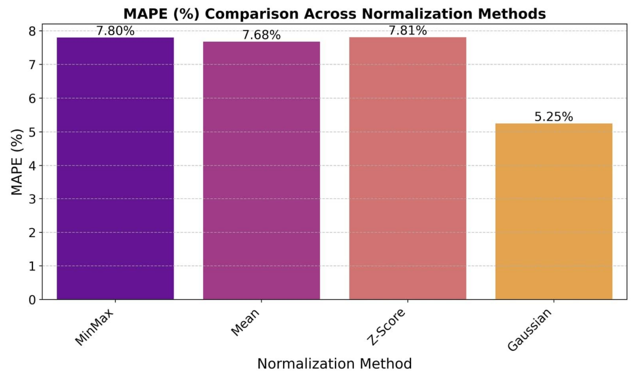

Fig. 8 MAPE (%) Comparison Across Normalization Methods

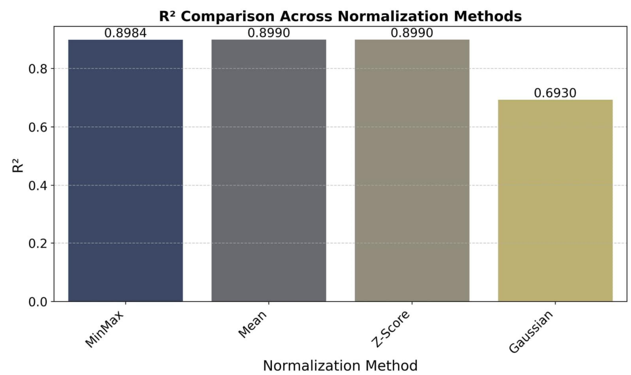

Fig. 9 R^2 Comparison Across Normalization Methods

Table - 2 Normalization methods and their pros & cons and uses.

Normalization Method	Primary Mathematical Strength	Target Power Sector Department	Core Operational Use Case
Gaussian	Minimized error magnitude (5.25% MAPE)	Commercial Billing & Energy Trading	Day-ahead market bidding; financial penalty mitigation
Mean	High variance fit (R^2: 0.8990); balanced error	Load Dispatch Centers & Grid Operators	Peak demand tracking; generation scheduling & ramp planning
Z-Score	Identical structural fit to Mean; scale neutrality	Distribution Network Operators	Substation asset management; transformer stress analysis
Min-Max	Bounded data scaling (0–1 range)	Corporate Planning & Policy	Long-term demand trend visualization; high-level dashboards

The table no. 2, compares four popular data normalisation approaches including Min-Max Scaling, Mean Normalisation, Z-Score Standardisation and Gaussian Transformation with their benefits, drawbacks and use cases. Data normalisation is a fundamental preprocessing step in machine learning to provide consistent feature scaling resulting in better model convergence and performance.

Methodologically, Min-Max Normalization is more susceptible to outliers because it relies on the minimum and maximum values, while Mean Normalization is influenced by the mean and data range. Z-Score Normalization is generally more robust as it uses the mean and standard deviation. According to the results, all three techniques achieved nearly similar performance, with Mean and Z-Score Normalization showing slightly better results in some evaluation metrics. In contrast, the custom Gaussian Normalization is not suitable because its transformation creates a many-to-one mapping, where different original values can produce the same normalized value. This causes information loss and makes accurate inverse transformation difficult, resulting in distorted forecasts and a lower R^2 value despite achieving lower error metrics.

VI. CONCLUSION

This paper explores the impact of various data normalization techniques on the short-term electrical load forecasting performance of a long short-term memory (LSTM) model. We conducted the evaluation of the four normalization methods (Min-Max Scaling, Mean Normalization, Z-Score Standardization, and Gaussian Normalization) using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R^2).

The experimental results showed that Gaussian Normalization achieved the lowest RMSE, MAE and MAPE, which indicates lower prediction errors but the lower value of R^2 indicated that the model was less capable of capturing the variations in the actual load pattern. However, Min-Max, Mean Normalization and Z-Score Standardization gave higher values of R^2 which means better predictive performance and consistency.

The results show that the choice of normalization method has a large impact on the performance of LSTM- based electrical load forecasting. Furthermore, more advanced deep learning architectures such as Gated Recurrent Unit (GRU), Transformer models, or hybrid LSTM models may enhance a model’s capacity to learn complex temporal dependencies and non-linear relationships in electricity load data.

REFERENCES

[1] J. He, K. Yuan, Z. Zhong, and Y. Sun, “Enhancing Short-Term Power Load Forecasting With a TimesNet-Crossformer-LSTM Approach,” *IEEE Access*, vol. 12, pp. 56774–56788, 2024, doi: 10.1109/ACCESS.2024.3383912.

[2] T. Hong and S. Fan, “Probabilistic electric load forecasting: A tutorial review,” *International Journal of Forecasting*, vol. 32, no. 3, pp. 914–938, Jul. 2016, doi: 10.1016/j.ijforecast.2015.11.011.

[3] P. Balachandra and V. Chandru, “Modelling electricity demand with representative load curves,” *Energy*, vol. 24, no. 3, pp. 219–230, Mar. 1999, doi: 10.1016/S0360-5442(98)00096-6.

[4] A. Azeem, I. Ismail, S. M. Jameel, and V. R. Harindran, “Electrical Load Forecasting Models for Different Generation Modalities: A Review,” *IEEE Access*, vol. 9, pp. 142239–142263, 2021, doi: 10.1109/ACCESS.2021.3120731.

[5] A. Berrada, K. Loudiyi, and I. Zorkani, “Profitability, risk, and financial modeling of energy storage in residential and large scale applications,” *Energy*, vol. 119, pp. 94–109, Jan. 2017, doi: 10.1016/j.energy.2016.12.066.

[6] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[7] K. D. Martinez-Zapata *et al.*, “Data-Driven Load Forecasting in Microgrids: Integrating External Factors for Efficient Control and Decision-Making,” *Energies*, vol. 19, no. 2, p. 555, Jan. 2026, doi: 10.3390/en19020555.

[8] K. Ijaz, Z. Hussain, J. Ahmad, S. F. Ali, M. Adnan, and I. Khosa, “A Novel Temporal Feature Selection Based LSTM Model for Electrical Short-Term Load Forecasting,” *IEEE Access*, vol. 10, pp. 82596–82613, Jan. 2022, doi: 10.1109/access.2022.3196476.

[9] N. Aqilah, S. A. Zaki, A. Hagishima, H. B. Rijal, and F. Yakub, “Analysis on electricity use and indoor thermal environment for typical air-conditioning residential buildings in Malaysia,” *Urban Climate*, vol. 37, p. 100830, May 2021, doi: 10.1016/j.uclim.2021.100830.

[10] H. Dong, Y. Gao, X. Meng, and Y. Fang, “A Multifactorial Short-Term Load Forecasting Model Combined With Periodic and Non-Periodic Features - A Case Study of Qingdao, China,” *IEEE Access*, vol. 8, pp. 67416–67425, Jan. 2020, doi: 10.1109/access.2020.2986031.

[11] L. Lin, L. Xue, Z. Hu, and N. Huang, “Modular Predictor for Day-Ahead Load Forecasting and Feature Selection for Different Hours,” *Energies*, vol. 11, no. 7, p. 1899, Jul. 2018, doi: 10.3390/en11071899.

[12] R. Srinivasan, V. Balasubramanian, and B. Selvaraj, “Short-Term Forecasting of Load and Renewable Energy Using Artifical Neural Network,” *International Journal of Engineering Trends and Technology - IJETT*, vol. 69, 2021, doi: 10.14445/22315381/IJETT-V69I6P226.

[13] T. A. Alghamdi and N. Javaid, “A Survey of Preprocessing Methods Used for Analysis of Big Data Originated From Smart Grids,” *IEEE Access*, vol. 10, pp. 29149–29171, Jan. 2022, doi: 10.1109/access.2022.3157941.

[14] M. Lepot, J.-B. Aubin, and F. H. L. R. Clemens, “Interpolation in Time Series: An Introductive Overview of Existing Methods, Their Performance Criteria and Uncertainty Assessment,” *Water*, vol. 9, no. 10, p. 796, Oct. 2017, doi: 10.3390/w9100796.

[15] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Comput. Surv.*, vol. 41, no. 3, p. 15:1-15:58, Jul. 2009, doi: 10.1145/1541880.1541882.

[16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. in Adaptive Computation and Machine Learning series. Cambridge, MA, USA: MIT Press, 2016. Accessed: Jul. 10, 2026. [Online]. Available: https://mitpress.mit.edu/9780262035613/deep-learning/

[17] J. M. H. Pinheiro *et al.*, "The Impact of Feature Scaling in Machine Learning: Effects on Regression and Classification Tasks," *IEEE Access*, vol. 13, pp. 199903–199931, Jan. 2025, doi: 10.1109/access.2025.3635541.

[18] S. Lu and T. Bao, "Short-Term Electricity Load Forecasting Based on NeuralProphet and CNN-LSTM," *IEEE Access*, vol. 12, pp. 76870–76879, 2024, doi: 10.1109/ACCESS.2024.3407094.

[19] P. Koukaras and C. Tjortjis, "Data Preprocessing and Feature Engineering for Data Mining: Techniques, Tools, and Best Practices," *AI*, vol. 6, no. 10, p. 257, Oct. 2025, doi: 10.3390/ai6100257.

[20] Niño-Adan, Iratxe & Portillo, Eva & Landa-Torres, Itziar & Manjarres, Diana. (2021). Normalization Influence on ANN-Based Models Performance: A New Proposal for Features' Contribution Analysis. IEEE Access. 9. 125462-125477. 10.1109/ACCESS.2021.3110647.

[21] J. Sola and J. Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems," *IEEE Transactions on Nuclear Science*, vol. 44, no. 3, pp. 1464–1468, Jun. 1997, doi: 10.1109/23.589532.

[22] Sujon, Khaled & Hassan, Rohayanti & Towshi, Zeba & Othman, Manal & Samad, Md Abdus & Choi, Kwonhue. (2024). When to Use Standardization and Normalization: Empirical Evidence From Machine Learning Models and XAI. IEEE Access. 12. 135300-135314. 10.1109/ACCESS.2024.3462434.

[23] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Deep Adaptive Input Normalization for Time Series Forecasting," *IEEE Trans Neural Netw Learn Syst*, vol. 31, no. 9, pp. 3760–3765, Sep. 2020, doi: 10.1109/TNNLS.2019.2944933.

[24] S. K.n., L. Pinto, S. Gopalan, and P. Balasubramaniam, "Equivalence class and modified Gaussian methods for normalization of time series data on AI models," *Expert Systems with Applications*, vol. 277, p. 127166, Jun. 2025, doi: 10.1016/j.eswa.2025.127166.

[25] Rousseeuw, Peter & Hubert, Mia. (2018). Anomaly Detection by Robust Statistics. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 8. 10.1002/widm.1236.

[26] Tunnicliffe Wilson, Granville. (2016). Time Series Analysis: Forecasting and Control, 5th Edition, by George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel and Greta M. Ljung, 2015. Published by John Wiley and Sons Inc., Hoboken, New Jersey, pp. 712. ISBN: 978-1-118-67502-1. Journal of Time Series Analysis. 37. 709-711. 10.1111/jtsa.12194.

[27] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Deep Adaptive Input Normalization for Time Series Forecasting," Sep. 22, 2019, *arXiv*: arXiv:1902.07892. doi: 10.48550/arXiv.1902.07892.

[28] "Investigating the Impact of Data Normalization Methods on Predicting Electricity Consumption in a Building Using different Artificial Neural Network Models | Request PDF," *ResearchGate*, doi: 10.1016/j.scs.2024.105570.

[29] Gers, Felix & Schmidhuber, Jürgen & Cummins, Fred. (2000). Learning to Forget: Continual Prediction with LSTM. Neural Computation. 12. 2451-2471. 10.1162/089976600300015015.

[30] Y. Wang, N. Zhang, and X. Chen, "A Short-Term Residential Load Forecasting Model Based on LSTM Recurrent Neural Network Considering Weather Features," *Energies*, vol. 14, no. 10, p. 2737, Jan. 2021, doi: 10.3390/en14102737.

[31] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, Oct. 2006, doi: 10.1016/j.ijforecast.2006.03.001.

[32] T. Hong, P. Pinson, and S. Fan, "Global Energy Forecasting Competition 2012," *International Journal of Forecasting*, vol. 30, no. 2, pp. 357–363, Apr. 2014, doi: 10.1016/j.ijforecast.2013.07.001.

[33] L. G. Rocha, S. Gomes Soares Alcala, and L. P. Garces Negrete, "Short-term electric load forecasting using neural networks: A comparative study," in *2020 IEEE PES Transmission & Distribution Conference and Exhibition - Latin America (T&D LA)*, Montevideo, Uruguay: IEEE, Sep. 2020, pp. 1–6. doi: 10.1109/TDLA47668.2020.9326196.

[34] A. D. Myttenaere, B. Golden, B. L. Grand, and F. Rossi, "Mean Absolute Percentage Error for regression models," *Neurocomputing*, vol. 192, pp. 38–48, Jun. 2016, doi: 10.1016/j.neucom.2015.12.114.

[35] Chicco, Davide & Warrens, Matthijs & Jurman, Giuseppe. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. PeerJ Computer Science. 7. e623. 10.7717/peerj-cs.623.