**RESEARCH ARTICLE** **ISSN 2738-9898 (Print) ISSN 2738-9901 (Online)**

# Economic Dispatch Considering Renewable Energy Generation and Load Forecasting

**Ashim Panta* and Ananta Adhikari**

*School of Engineering, Faculty of Science and Technology, Pokhara University Kaski, Nepal*

* Author to whom correspondence should be addressed; E-Mail: *ashimkumarpanta@gmail.com*

*Received : 04 July 2025; Received in revised form : 29 October 2025; Accepted : 05 December 2025; Published: 29 January 2026*

**Abstract**

Accurate short-term load forecasting (STLF) and optimal generation scheduling are essential for reliable and economic power system operation. This paper proposes a hybrid deep learning model combining Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) networks for STLF, integrated with a Genetic Algorithm (GA)-based Economic Dispatch (ED) framework. The model is trained and validated using the publicly available Panama Short-Term Load Forecasting dataset, which provides hourly electricity demand along with meteorological variables. Input features include temperature, wind speed, time of day, day of the week, and historical demand. Forecasting performance is evaluated using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the Coefficient of Determination ($R^2$). Simulations on the IEEE 9-bus system demonstrate that the proposed GRU+LSTM model achieves superior performance compared to standalone GRU and LSTM, with an MSE of 0.0310, RMSE of 0.1760, and $R^2$ of 0.9670. Furthermore, the forecasted demand is used in the ED problem, formulated as a multi-objective function incorporating fuel cost, emission cost, and wind generation cost. Results confirm that the proposed integrated approach enhances forecasting accuracy and reduces operational costs, making it effective for data-driven power system operation.

**Keywords**

Economic Dispatch, Gated Recurrent Unit, Genetic Algorithm, Long Short term Memory, Renewable Energy Integration, Short Term Load Forecasting

## 1. Introduction

The increasing integration of renewable energy sources into modern power systems has introduced both opportunities and challenges. Renewable energy sources such as wind and solar exhibit inherent variability and uncertainty, making it difficult to maintain grid stability and ensure reliable power generation. These fluctuations necessitate advanced management strategies to optimize economic dispatch (ED) while accommodating renewable generation constraints. Traditional economic dispatch models were designed for stable and predictable energy generation; however, the intermittent nature of renewable energy sources requires more flexible and adaptive approaches [1], [2],[3].

Economic Dispatch (ED) plays a crucial role in power system management by determining the optimal distribution of generation resources to meet load demands while minimizing operational costs and emissions. Unlike classical ED, which operates on a static basis, modern ED adjusts generator outputs in response to real-time fluctuations in load and generation availability. Various optimization techniques, including Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Differential Evolution (DE), have been applied to address the complexities associated with ED in renewable-rich grids [4],[5],[6]. Studies have demonstrated that GA-based ED models effectively optimize resource allocation, minimize generation costs, and reduce emissions [7],[8],[9].

Short-Term Load Forecasting (STLF) is a fundamental component of modern power systems, enabling accurate demand prediction for efficient grid operation. Traditional STLF methods relied on statistical techniques; however, recent advancements in machine learning, particularly deep neural networks (DNNs), have significantly improved forecasting accuracy [10],[11],[12]. Recurrent Neural Networks (RNNs) such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks have demonstrated superior performance in capturing temporal dependencies in load demand data. Studies have shown that LSTM-based models outperform conventional statistical models in STLF applications, leading to more reliable demand predictions and optimized generation scheduling [13],[14],[15].

Integrating STLF with ED presents a comprehensive solution to managing real-time energy demand and generation schedules. The use of GA in ED optimization ensures an adaptive approach to minimizing generation costs while addressing the variability of renewable energy sources. Research has shown that hybrid models combining LSTM-based STLF with GA-based ED provide a promising solution by enhancing grid reliability and economic efficiency [16],[17],[18].

Despite advancements in STLF and ED optimization, several challenges remain. Data quality and availability significantly impact forecasting accuracy, and computational complexity can limit real-time implementation. Additionally, the variability of renewable energy sources introduces further uncertainty in economic dispatch planning [19],[20],[21]. Several studies have explored these challenges. Research by [22] highlighted the importance of real-time data acquisition for improving forecasting accuracy, whereas [23] investigated computational efficiency issues in machine learning-based forecasting. Additionally, several studies have focused on improving short-term load forecasting accuracy through advanced deep learning–based approaches and hybrid optimization techniques [24], [25].

Studies by [26] and [27] provided insights into grid reliability concerns caused by renewable energy fluctuations, emphasizing the need for robust optimization techniques. Research by [28],[29],[30] explored hybrid forecasting-dispatch frameworks, demonstrating their effectiveness in reducing operational uncertainties.

This study presents an integrated approach that combines LSTM and GRU networks for improved STLF accuracy with a GA-based ED optimization framework. By leveraging machine learning for demand prediction and evolutionary algorithms for economic dispatch, the proposed methodology aims to enhance power system stability, reduce operational costs, and facilitate the integration of renewable energy sources into modern grids. The study contributes to ongoing research efforts in power system optimization by demonstrating the effectiveness of AI-driven forecasting and optimization models in managing the complexities of renewable energy integration.

## 2. Methodology

### 2.1 *Research Framework*

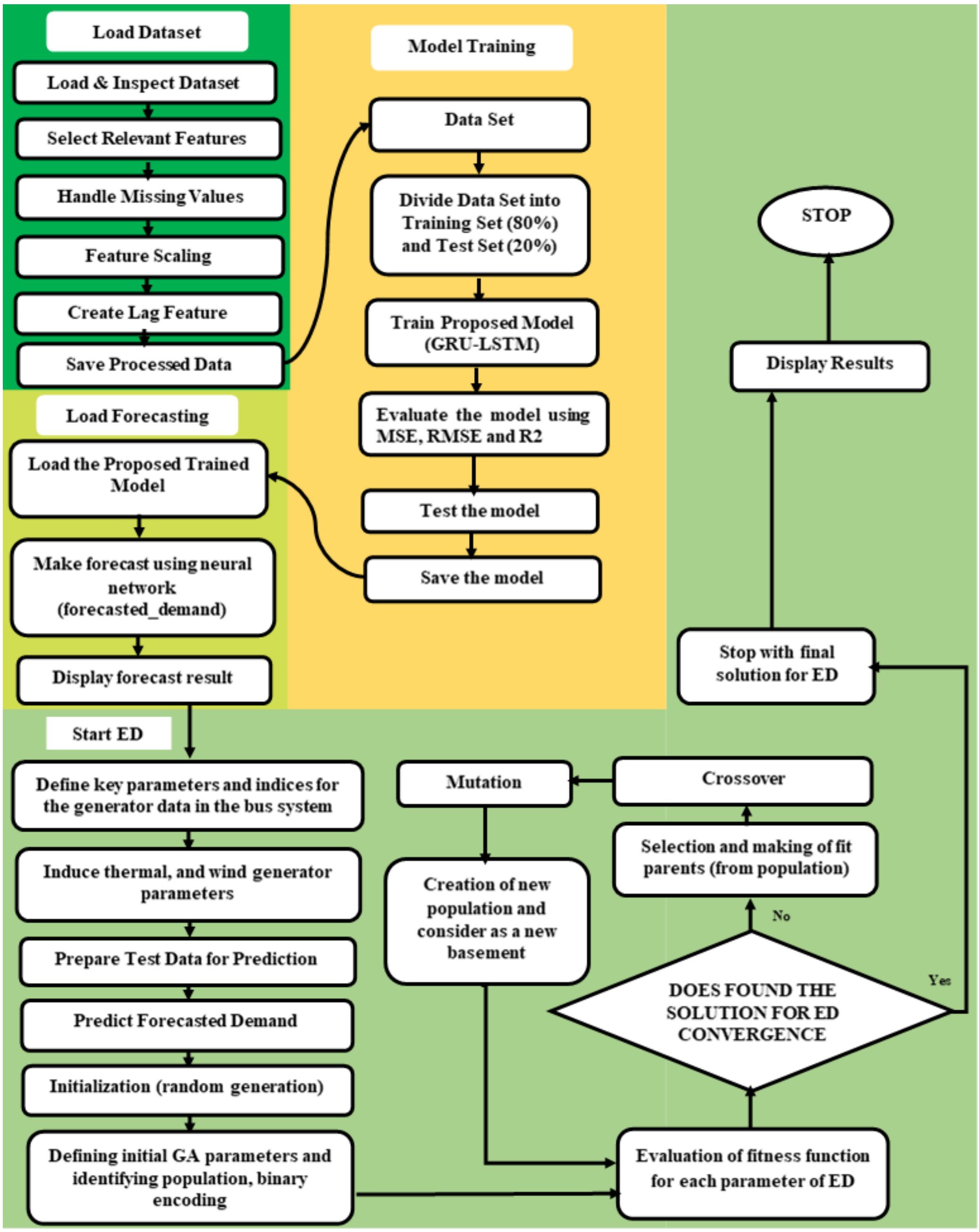


Figure 1: Proposed Research framework

Figure 1 shows the proposed research framework that provides a structured process for optimizing energy dispatch in a power system by integrating Short-Term Load Forecasting (STLF) using Neural Networks (GRU-LSTM) and solving the Economic Dispatch (ED) problem using Genetic Algorithms (GA).Initially, the dataset is prepared by selecting features and filling missing data. The dataset is then split into training and testing sets. A GRU-LSTM model is trained on the data to predict the short-term demand and evaluated using performance metrics like MSE, RMSE, and $R^2$. Once the forecasting model is tested and saved, it is used to predict future demand.

Next, ED is started by defining key parameters for thermal and wind generators. The predicted demand from the forecasting model is then used in the ED process. GA is used for optimizing generator scheduling, starting with the initialization of a random population. The GA iterates through selection, crossover, and mutation steps to evolve better solutions until the ED optimization converges on a solution. The final ED results are then displayed, showing an optimal strategy for energy dispatch based on forecasted demand. The network model in this study represents a power transmission system comprising conventional fossil fuel-based generators and wind power generators. The focus is on integrating Economic Dispatch (ED) with Short-Term Load Forecasting (STLF) to achieve optimal power generation scheduling while minimizing operational costs and ensuring grid stability. This approach is particularly important in scenarios where renewable energy sources like wind contribute significantly to the power mix, introducing variability and uncertainty into the system. The primary objective is to minimize the total operational cost, which is modeled as a weighted combination of three components: fuel cost, wind power generation cost, and emission cost. The objective function is defined

$$\min F_T = \sum_{t=1}^{T}\sum_{i=1}^{N} \lambda_1 FC_{i,t}\left(P_{i,t}\right) + \lambda_2 EC_{i,t}\left(P_{i,t}\right) + \lambda_3 C(P_{w,i,t})] \qquad \dots(1)$$

as:

where:

$\lambda_1$=0.65: Highest priority is given to fuel cost minimization due to its significant impact on operational expenses.

$\lambda_2$=0.29: Emission cost is assigned moderate priority to balance environmental sustainability with economic considerations.

$\lambda_3$=0.06: Wind power generation cost is given the lowest priority as wind energy is generally cost-effective, but operational variability must still be considered.

The detailed objective functions are given from equation (2) to (4)

1. Fuel cost function: The fuel cost is modeled as a quadratic function, commonly used in economic dispatch problems:

$$FC_{i,t}(P_{i,t}) = a_i + b_i P_{i,t} + c_i P_{i,t}^2 + |d_i sin\{e_i(P_{i,t}^{min} - P_{i,t})\}| \quad (2)$$

Here,

a: Fixed Cost Coefficient : Represents the constant cost incurred regardless of the power output, such as startup or no-load costs.

b: Linear Cost Coefficient: Represents the cost per unit of power generated, assuming a linear increase with output.

c: Quadratic Cost Coefficient: Models the nonlinear increase in cost with higher power output due to inefficiencies at higher operating levels.

d: Amplitude of Valve-Point Effect Coefficient: Represents the magnitude of cost oscillations caused by the valve-point effect in thermal generators.

e: Frequency of Valve-Point Effect Coefficient: Represents the frequency of the oscillations in the cost function due to valve-point effects.

2. Emission cost function: The emission cost function is modeled as a quadratic equation, incorporating coefficients that represent different emission factors:

$$EC_{i,t}(P_{i,t}) = \sigma_i (P_{i,t})^2 + \tau_i P_{i,t} + \delta_i \quad (3)$$

Here,

σ: Quadratic Emission Coefficient: Represents the non-linear relationship between generator power output and emissions (e.g., $CO_2$ emissions).

τ: Linear Emission Coefficient: Models the direct proportionality of emissions with power output

δ: Fixed Emission Offset: Represents baseline emissions that occur regardless of generator output.

3. Wind power generation cost: The cost associated with wind power generation is expressed as a function of wind power output.

$$C(P_{w,i,t}) = aI^P P_{w,i,t} + G^E P_{w,i,t} \quad (4)$$

Here,

$a_I^P$: Linear Wind Cost Coefficient: Reflects operational costs that scale with wind power output, such as maintenance or ancillary services.

$G^E$: Grid Integration Cost Coefficient: Represents the cost associated with integrating variable wind power into the grid, such as balancing or reserve requirements.

### *2.2 Dataset Description and Statistical Summary*

The ‘short term Electricity Load Forecasting-Panama’ dataset, released in 2023 and publicly available on Kaggle, is used in this study.It provides hourly electricity load data along with associated meteorological variables, making it highly suitable for developing Short-Term Load Forecasting (STLF) models.

Dataset Features:

Date/Hour - timestamp of each record.

Load Demand (MW) -hourly electricity consumption in Panama.

Temperature (°C) - environmental temperature.

Humidity (%) - relative humidity.

Wind Speed (m/s) - wind velocity at the given hour.

Derived Features - hour of the day, day of the week, and lagged demand values were generated for improved predictive performance.

Data Coverage:

The dataset spans multiple years with hourly resolution, ensuring sufficient granularity to capture both short-term variations and seasonal load trends

Preprocessing:

Missing values were handled using linear interpolation.

Normalization was applied to scale features into the [0,1] range.

Feature engineering included lagged demand, day-of-week, and hour-of-day encodings to strengthen temporal learning. Table 1 lists the statistical summary of the input features used in the proposed load forecasting and economic dispatch framework.

Table 1: Statistical Summary of Input Features

| **Feature** | **Mean** | **Std. Dev.** | **Min** | **Max** |
|---|---|---|---|---|
| Load Demand (MW) | 1202.16 | 123.97 | 898.06 | 1405.77 |
| Temperature (°C) | 25.77 | 1.2 | 23.76 | 31.21 |
| Wind Speed (m/s) | 19.7 | 2.02 | 15.75 | 25.95 |
| Hour of the Day | 12 | 6.92 | 0 | 23 |
| Day of the Week | 3.9 | 2 | 1 | 7 |

### *2.3 Hyperparameter Tuning Methodology*

To ensure robust forecasting accuracy, the hyperparameters of GRU, LSTM, and GRU+LSTM models were tuned systematically. A grid search approach was adopted over a predefined search space, with evaluation performed on a validation set using MSE,RMSE and coefficient of determination as the criterion.

Learning Rate: [0.001, 0.0005, 0.0001]; Batch Size: [16, 32, 64]; Epochs: [100, 200, 300]; Dropout Rate: [0.1, 0.2, 0.3]; Hidden Units: [50, 100, 150]

The final selection (200 epochs, batch size = 32, learning rate = 0.0001, 100 hidden units, dropout = 0.2 for hybrid) was based on the lowest RMSE on validation data. The Adam optimizer was employed for all models due to its adaptive learning rate properties.

### 2.4 Genetic Algorithm Implementation

The Economic Dispatch problem was solved using a Genetic Algorithm (GA) owing to its effectiveness in handling nonlinear, multi-objective optimization problems. To ensure solution reliability, the GA was implemented with well-defined convergence criteria and stopping conditions. The algorithm was terminated when either of the following conditions was met:

1. The maximum number of generations was reached (200).
2. The relative improvement in the best fitness value remained below $10^{-6}$ for 20 consecutive generations, indicating convergence.

Table 2 lists the parameters used for the genetic algorithm in this study.These values were selected based on preliminary experiments and validated for stability and accuracy in the IEEE 9-bus test system.

Table 2: List of GA Parameters Used in Economic Dispatch

| Parameter | Value | Description |
|---|---|---|
| Population Size | 50 | Number of candidate solutions per generation |
| Maximum Generations | 200 | Upper limit for evolutionary iterations |
| Selection Method | Tournament | Selects best candidates for reproduction |
| Crossover Operator | SBX (Simulated Binary Crossover) | Preserves diversity while combining parents |
| Crossover Probability | 0.8 | Probability of crossover between chromosomes |
| Mutation Operator | Polynomial | Introduces small random variations |
| Mutation Probability | 0.1 | Probability of mutation per gene |
| Elitism | Enabled (5%) | Preserves top 5% best solutions each generation |
| Stopping Criteria | Max generations OR improvement $< 10^{-6}$ for 20 iterations | Ensures convergence |

*Convergence Analysis*

To evaluate convergence behavior, the variation of the best fitness value (total generation cost) across generations was recorded. As shown in Figure X, the GA converged steadily, with significant improvement observed in the first 50 generations and near-stable optimal values achieved by generation 150. This indicates that the chosen parameter configuration ensured both exploration and exploitation, avoiding premature convergence.

**2.5 Proposed AI-Based Methodology**

This work implements AI-based techniques for solving the ED and STLF problems.Two AI techniques were selected for solving STLF and ED problems, viz. DNN and GA respectively.

**2.5.1 GRU for STLF**

The Gated Recurrent Unit (GRU) is an advanced neural network architecture designed to process sequential data, particularly time-series data such as Short-Term Load Forecasting (STLF). GRU belongs to the family of Recurrent Neural Networks (RNNs) and was introduced as a simplified and computationally efficient alternative to Long Short-Term Memory (LSTM) networks. Unlike standard RNNs, which suffer from vanishing gradient problems, GRUs incorporate gating mechanisms to manage the flow of information, making them highly suitable for capturing temporal dependencies in sequential data. The GRU architecture is characterized by its use of two gates: the update gate and the reset gate, which work together to control how much information from the past is retained or forgotten. This mechanism allows GRUs to model both short-term and long-term dependencies in time-series data effectively, while requiring fewer parameters than LSTMs. In this research, a GRU-based model is implemented to forecast electricity load demand based on historical data and additional features. Figure 2 shows the structure of the gated recurrent unit (GRU) model used in this work.

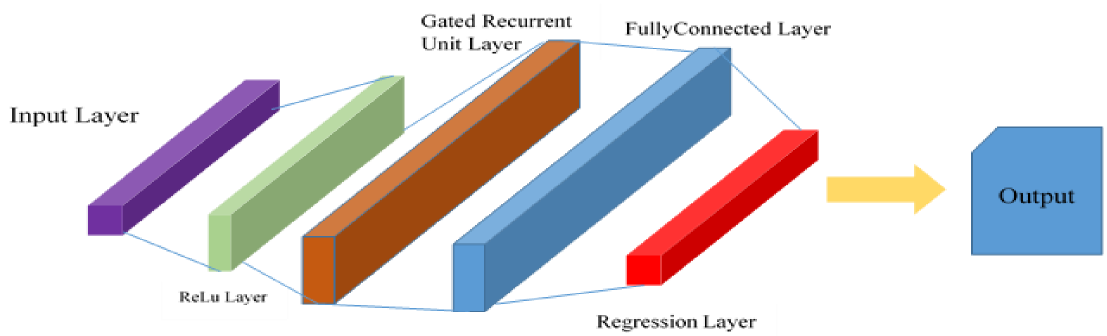


Figure 2: GRU-based STLF structure

***Architecture Overview***

The GRU network used in this research comprises five main layers tailored for efficient short-term load forecasting. The input layer receives normalized sequential input features, which include temperature, wind speed, hour of the day, day of the week, and lagged load demand—amounting to five features per time step. These inputs are then processed by a ReLU activation layer, which introduces non-linearity and enables the model to detect complex patterns such as peaks, fluctuations, and feature correlations in electricity demand.

The core GRU layer contains 100 hidden units that capture temporal dependencies across time steps. It employs two gating mechanisms: an update gate, which determines how much of the previous information should be retained, and a reset gate, which controls how much of the past data should be forgotten. These mechanisms enable the GRU to combine memory retention and simplification, making it suitable for sequential time-series forecasting. The output of the GRU is passed to a fully connected layer with a single neuron that generates the forecasted electricity load for the next time step. This output is then evaluated by a regression layer using Mean Squared Error (MSE) as the performance metric, which is minimized during training to enhance predictive accuracy. The training process uses normalized input data to ensure consistent scaling and convergence. The model is trained using the Adam optimizer due to its adaptive learning rate capabilities, with key hyperparameters set as 200 epochs, a mini-batch size of 32, and an initial learning rate of 0.0001. Training progress is monitored using a plot that displays error reduction over time. Once trained, the GRU model (referred to as GRUnet) is saved for future prediction tasks, enabling reliable forecasting of electricity demand based on temporal and environmental inputs[31].

**2.5.2 LSTM for STLF**

Long Short-Term Memory (LSTM) networks are a specialized type of Recurrent Neural Network (RNN) designed to effectively model sequential and time-dependent data. Unlike standard RNNs, which struggle with long-term dependencies due to issues like vanishing gradients, LSTMs introduce memory cells that enable them to retain information over extended periods. These cells are governed by a gating mechanism, which ensures that relevant information is stored, updated, or discarded dynamically.

LSTMs are particularly well-suited for Short-Term Load Forecasting (STLF), where electricity demand patterns are influenced by complex temporal dependencies and external factors such as weather conditions and time-of-day effects. By capturing both short-term fluctuations and long-term trends, LSTMs provide accurate and reliable forecasts, making them a robust choice for time-series forecasting tasks .Figure 3 shows the structure of the gated recurrent unit (GRU) model used in this work.

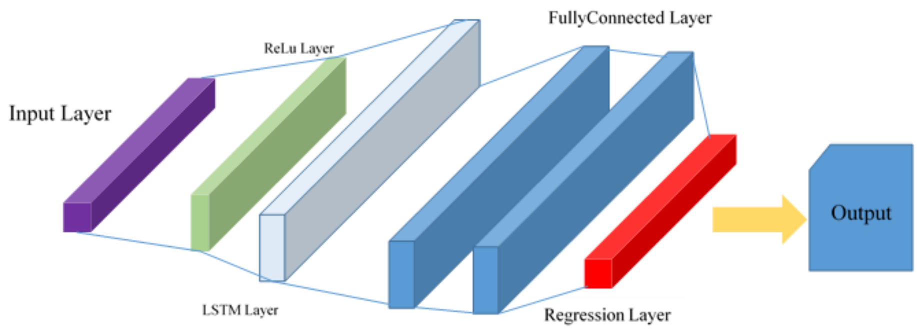


Figure 3: LSTM-based STLF structure

***Architecture Overview***

The LSTM network architecture employed in this research is structured into several sequential layers to effectively model short-term load forecasting. The process begins with the input layer, which receives sequentially organized data comprising five key features per time step: temperature, wind speed, hour of the day, day of the week, and lagged load demand. This input is normalized to ensure consistent scaling and to enhance the training process. The data then passes through a ReLU activation layer, which introduces non-linearity and enables the model to learn complex relationships within the dataset, such as peak demands and sudden fluctuations. At the core of the network lies the LSTM layer, which contains 100 hidden units designed to extract temporal dependencies from the sequence data. The LSTM architecture uses three internal gating mechanisms: the forget gate, which determines which information from the past should be discarded; the input gate, which identifies new relevant information to be stored; and the output gate, which regulates what part of the memory should be passed forward. This gate-based structure allows the model to retain long-term trends and short-term variations in energy consumption effectively. The output from the LSTM is then passed to a fully connected layer with 40 neurons that extract and refine high-level features. This is followed by a second fully connected layer with a single neuron, which generates the final forecasted load value. Finally, the regression layer compares the forecasted output to the actual load values using Mean Squared Error (MSE) as the evaluation metric. This error is minimized during training to ensure the model achieves high predictive accuracy in forecasting electricity demand[32].

### 2.5.3. Proposed Hybrid GRU+LSTM for STLF

The proposed GRU+LSTM hybrid model combines the strengths of Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) architectures to improve short-term load forecasting (STLF). This integration leverages the efficiency of GRU and the advanced memory capabilities of LSTM to capture both short-term and long-term dependencies in sequential data effectively.

***Architecture Overview***

The proposed hybrid GRU-LSTM model for short-term load forecasting is composed of multiple layers, each serving a specific purpose in handling sequential time-series data. It begins with an input layer that accepts normalized features including temperature, wind speed, hour of the day, day of the week, and lagged load demand—totalling five features per time step. These inputs are processed through a GRU layer consisting of 100 hidden units, which provides sequential output for each time step and effectively learns temporal patterns across the sequence using update and reset gates to manage memory efficiently.

Following this, a dropout layer with a 20% rate is introduced to randomly deactivate neurons during training, helping to reduce over fitting and enhance generalization. The output then flows into an LSTM layer, also with 100 hidden units, which captures long-term dependencies in the data. The LSTM employs forget, input, and output gates to selectively manage the information flow and preserve relevant knowledge from the sequence. A second dropout layer, again with a 20% rate, is used after the LSTM layer to further regularize the network. The extracted features are passed into two fully connected layers—one with 100 neurons to refine high-level patterns and another with a single neuron to produce the final forecasted electricity demand. This output is evaluated using a regression layer that computes the prediction error based on the Mean Squared Error (MSE), which is minimized during training to enhance model accuracy. The training process utilizes normalized input data (X_train and Y_train) to maintain consistent scaling and avoid dominance by large-valued features. The Adam optimizer is employed for its adaptive learning rate properties, with training conducted over 200 epochs, a mini-batch size of 32, and an initial learning rate of 0.0001 for stable and efficient convergence. The training progress is visualized through a plot that tracks error reduction across epochs, and upon completion, the trained hybrid model (referred to as Proposednet) is saved for use in future prediction tasks. Figure 4 shows the structure of the proposed model for STLF, based on the GRU with Long short-term memory (LSTM). [33]. Table 3 lists the parameters used for all models .

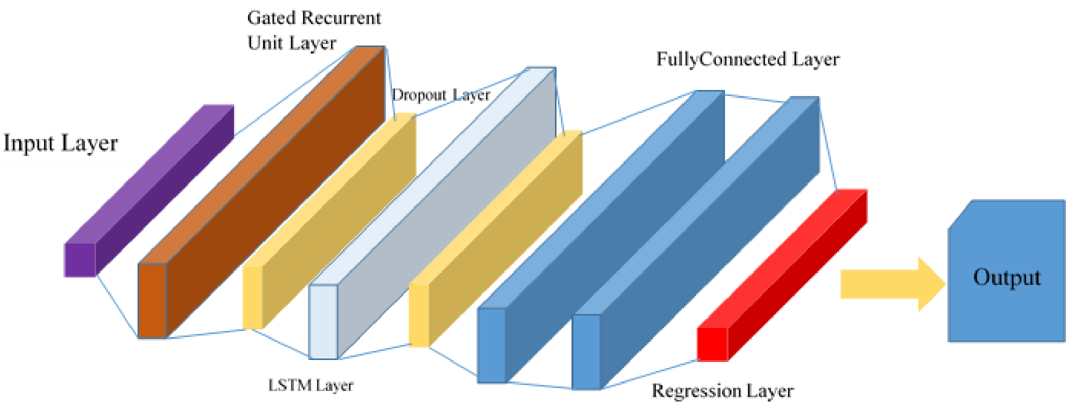


Figure 4: STLF structure based on a GRU with LSTM

Table 3 : List of Parameter used for all the models

| **Hyperparameter** | **GRU** | **LSTM** | **GRU+LSTM** |
|---|---|---|---|
| Input size | 5 | 5 | 5 |
| GRU units | 100 | - | 100 |
| LSTM units | - | 100 | 100 |
| Dropout Rate | - | - | 0.2 |
| Batchsize | 32 | 32 | 32 |

| | | | |
|---|---|---|---|
| Optimizer | Adam | Adam | Adam |
| Learning Rate | 0.0001 | 0.0001 | 0.0001 |
| Max epoch | 200 | 200 | 200 |

#### 2.5.4. Genetic Algorithm for Economic Dispatch

In this research work, the Genetic Algorithm (GA) is employed to solve the Economic Dispatch (ED) problem, focusing on optimizing generator scheduling to minimize the total operational cost while satisfying system constraints. GA is chosen due to its effectiveness in handling non-linear, multi-modal optimization problems, especially when integrating forecasted load demands from Short-Term Load Forecasting (STLF). The key steps and operations of the GA as used in this study are detailed below.

***Initialization:***

Population Initialization: Randomly generate an initial population of solutions, $P^{(0)}$, where each solution $x_i$ represents a set of generator outputs:

$$P^{(0)} = \{x_1, x_2, \dots, x_N\}, \quad x_i = \{P_{G1}, P_{G2}, \dots, P_{Gn}\} \qquad \dots (5)$$

with $P_{Gj}$ constrained by generator limits:

$$P_{Gj}^{min} \leq P_{Gj} \leq P_{Gj}^{max}$$

Encoding: Solutions are encoded in binary format for evolutionary operations.

***Fitness Function Evaluation:***

The fitness function evaluates each candidate solution $\boldsymbol{x}$ based on the total generation cost and power balance constraints:

$$F(\boldsymbol{x}) = \sum_{j=1}^{n}\left(a_j P_{Gj}^2 + b_j P_{Gj} + c_j\right) + \lambda\left|\sum_{j=1}^{n} P_{Gj} - P_D\right| \qquad \dots (6)$$

Where:

$a_j b_j c_j$: Cost coefficients for generator $j$, $P_{Gj}$: Power output of generator $j$, $P_D$: Total load demand, $\lambda$: Penalty factor for infeasible solutions.

***Selection:***

Selection of Fit Parents: Using a fitness-proportional selection method, such as tournament selection individuals with better fitness values are chosen as parents:

$$Probability\ of\ selection\ for\ \boldsymbol{x}_i = \frac{F(\boldsymbol{x}_i)}{\sum_{k=1}^{N} F(\boldsymbol{x}_k)} \qquad (7)$$

***Crossover:***

Crossover Operation: The selected parent solutions are combined using Simulated Binary Crossover (SBX) to create offspring solutions. For two parents $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, the offspring $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are generated as:

$$y_j^1 = 0.5\left[\left(1 - \beta_q\right)x_j^1 + \left(1 + \beta_q\right)x_j^2\right] \qquad (8)$$

$$y_j^2 = 0.5\left[\left(1+\beta_q\right)x_j^1 + \left(1-\beta_q\right)x_j^2\right] \tag{9}$$

Where:

$\beta_q$ is the crossover distribution index, calculated based on a random value $u_j$:

$$\beta_q = \begin{cases} \left(2u_j\right)^{\frac{1}{\eta_c+1}}, & if u_j \leq 0.5 \\ \left[\frac{1}{2\left(1-u_j\right)}\right]^{\frac{1}{\eta_c+1}}, & if u_j > 0.5 \end{cases} \tag{10}$$

$\eta_c$: Crossover control parameter.

***Mutation:***

Mutation Operation: To introduce diversity, mutation is performed on offspring solutions using Polynomial Mutation. The mutated gene $x'_j$ is calculated as:

$$\boldsymbol{x'_j = x_j + \Delta x_j} \tag{11}$$

Where:

$$\Delta x_j = \begin{cases} \delta_j\left(x_j - x_j^{min}\right), & if u_j \leq 0.5 \\ \delta_j\left(x_j^{max} - x_j\right), & if\ u_j > 0.5 \end{cases} \tag{12}$$

$$\delta_j = \left[2u_j + \left(1-2u_j\right)\left(1-\Delta\right)^{\eta_m+1}\right]^{\frac{1}{\eta_m+1}} - 1 \tag{13}$$

Where:

$\eta_m$: Mutation control parameter,

$\Delta$: A small constant.

***Convergence Check:***

The algorithm checks if the optimal solution is found by evaluating the fitness function for the new population. If the convergence criterion is met (e.g., minimal improvement in fitness or maximum generations), the process stops. Otherwise, the new population becomes the basis for the next iteration.

Solution: The final solution $x^* = arg \min_{x\in P} F(x)$ represents the optimal generator outputs:

Result Display: Once the GA converges, the solution is displayed, showing the optimal generator scheduling and corresponding total cost[34].

## 2.3 Evaluation Measures

### *2.3.1 Evaluation Measures for STLF*

When evaluating the performance of predictive models, especially for time series forecasting, it's essential to use robust metrics. Three widely used evaluation measures are Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the Coefficient of Determination ($R^2$). And for evaluation of ED metrics are cost evaluation percentage, system stability index and emission rates. Below is a detailed description of each metric, along with their respective equations.

a) Mean Squared Error (MSE)

MSE measures the average squared difference between the actual and predicted values. It quantifies the error by penalizing larger deviations more severely. A lower MSE indicates a better fit of the model to the data. The equation of this measure is given below.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{14}$$

Where: $n$ = number of observations; $y_i$ = actual value; $\hat{y}_i$ = predicted value

b) Root Mean Squared Error (RMSE)

RMSE is the square root of MSE. It provides an error measure in the same units as the target variable, making it more interpretable. Like MSE, a lower RMSE indicates a better model performance. The equation of this measure is given below.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{15}$$

Where: $n$ = number of observations; $y_i$ = actual value'; $\hat{y}_i$ = predicted value

c) Coefficient of Determination ($R^2$)

$R^2$ measures the proportion of the variance in the dependent variable that is predictable from the independent variables. It ranges from 0 to 1, where 1 indicates a perfect fit and 0 indicates that the model does not explain any of the variability in the response variable. The equation of this measure is given below.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \quad . \tag{16}$$

where: $n$ = number of observations; $y_i$ = actual value; $\hat{y}_i$ = predicted value; $\bar{y}$ = mean of actual values

**2.3.2 Evaluation Metrics for Economic Dispatch (ED)**

(a) Cost Reduction Percentage

The Cost Reduction Percentage metric evaluates the financial savings achieved by a given ED strategy relative to traditional or baseline methods. It is calculated as:

$$Cost\ Reduction\ Percentage = \left(\frac{Cost_{traditional} - Cost_{DED-based}}{Cost_{traditional}}\right) \times 100$$

(17)

(b) System Stability Index

The System Stability Index measures the stability of the grid under ED strategies by examining variations in power and voltage across the network. The formula is:

$$Stability\ Index = \sum_{i=1}^{N} \left|\frac{\Delta P_i}{P_i}\right| + \sum_{j=1}^{M} \left|\frac{\Delta V_j}{V_j}\right|$$

(18)

(c) Emission Rate

The Emission Rate metric evaluates the environmental impact of a ED approach by calculating the amount of emissions generated per unit of energy produced. It is defined as:

$$Emission\ Rate = \frac{Total\ Emissions}{Total\ Energy\ Generated} \quad (19)$$

**3. Results and Discussions**

Figure 6 shows the IEEE 9-bus system with two fossil-fuel-based generating units and one wind-based generating unit, used to test the proposed approach. We have designed a GUI for this research work, and the main window of the GUI is given in Fig 6. When the user clicks on the start button in the middle, it will initiate this research work and start the training of our STLF model. Figure 7 shows the window of training for our STLF model. All simulations and experiments were performed on a computer equipped with an Intel Core i7-1165G7 processor (2.80 GHz), 8 GB LPDDR4X RAM, Intel Iris Xe Graphics GPU, running Windows 10 Pro 64-bit, and using MATLAB R2024b software.

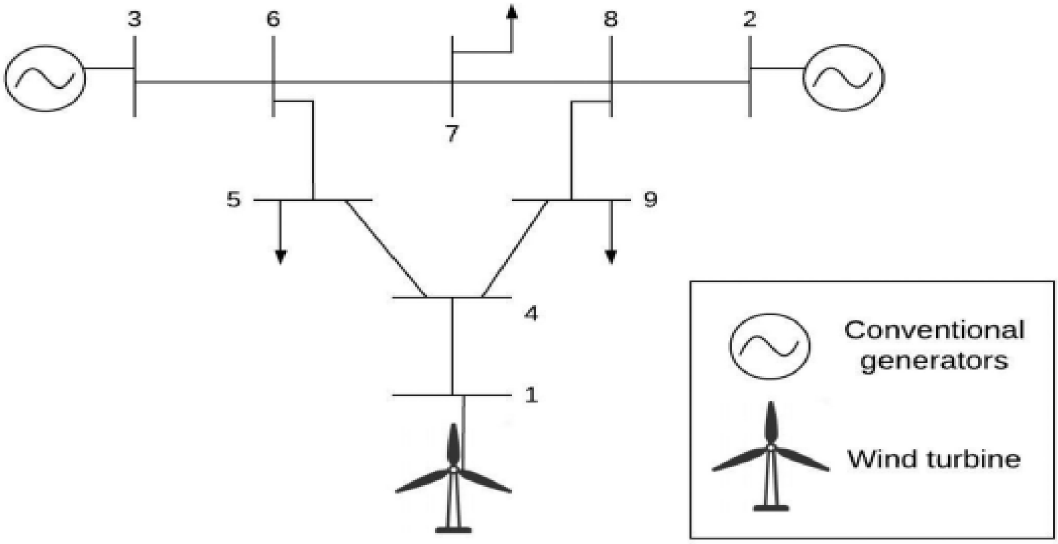


Figure 5: IEEE 9-bus system

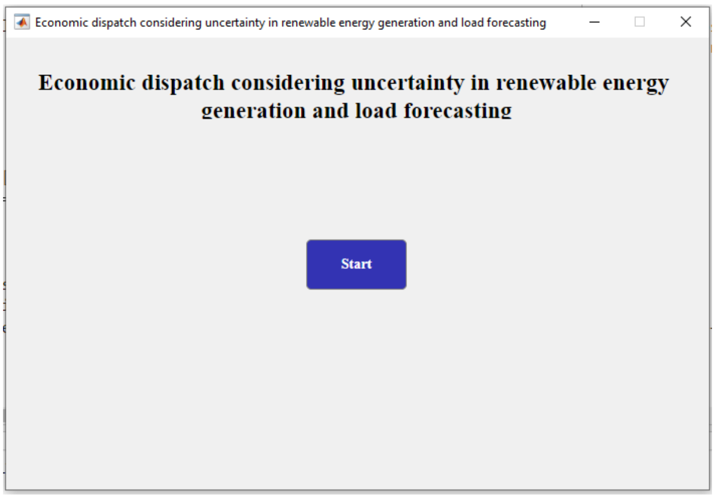


Figure 6: Main Window of this Research work

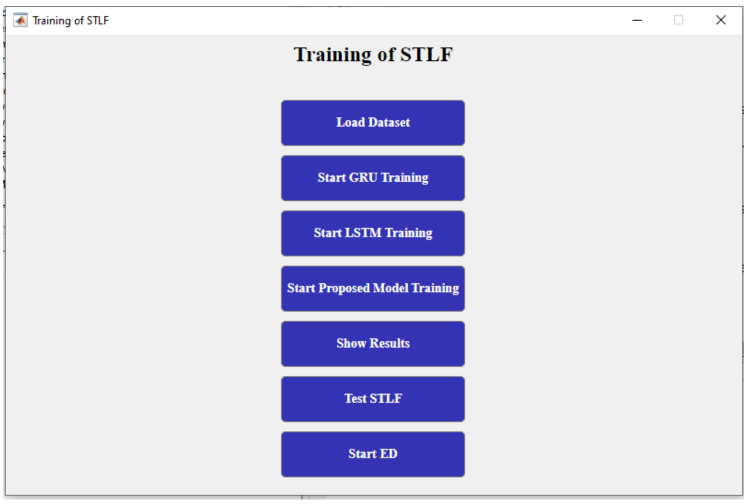


Figure 7: Training window of STLF

First, STLF is performed followed by the ED to find the optimal power generation schedule. To train the STLF model, first, dataset is loaded into the system. Fig 8 shows the Training graph for our STLF model. From the graph, we can see that the model is trained smoothly. Table 4 provides the lists of Sample Input and output of all the method. Table 5 and Table 6 shows the result of our STLF model based on these measures. Figure 9 show the graphical comparison results of load forecasting. The low value of MSE indicates that the average squared difference between the actual and predicted load values is minimal. It suggests that the STLF model has high accuracy and very small error deviations. The RMSE value of 0.1760 shows that the typical error in the load predictions is 0.1760 units. Given the scale of the data, this is a small error, which implies that the model's predictions are highly reliable and close to the actual values. An $R^2$ value of 0.9670 is the best possible outcome, indicating that the model perfectly predicts the load values. It means that there is no unexplained variance, and the model captures all the variability in the data, demonstrating superior performance

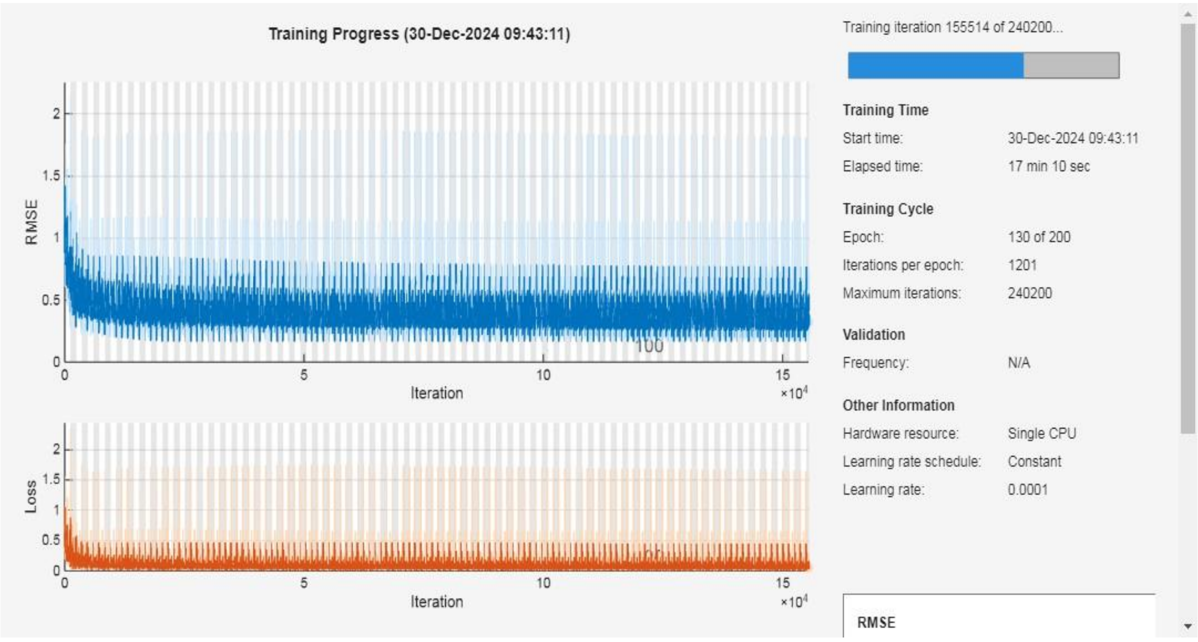


Figure 8: Training graph of our STLF

Table 4: List of Sample Input and output of all the methods

| X1(Temp) | X2(Wind Speed) | X3(Time) | X4(Day of the week) | X5(Lag Features) | Actual Demand | GRU | LSTM | Proposed (GRU+LSTM) |
|---|---|---|---|---|---|---|---|---|
| 25.8993 | 22.1669 | 2 | 7 | -1.1065 | 912.1755 | 1085.1091 | 1084.1292 | 922.9228 |
| 25.9373 | 22.4549 | 3 | 7 | -1.4094 | 900.2688 | 1086.6687 | 1085.6484 | 878.8319 |
| 25.9575 | 22.1105 | 4 | 7 | -1.4713 | 889.9538 | 1084.8022 | 1083.8301 | 878.8682 |
| 25.9738 | 21.1861 | 5 | 7 | -1.5251 | 893.6865 | 1079.7263 | 1078.8823 | 883.8198 |
| 26.0341 | 20.062 | 6 | 7 | -1.5056 | 879.2323 | 1073.4258 | 1072.7325 | 908.5511 |
| 26.6915 | 21.6235 | 7 | 7 | -1.5809 | 932.4876 | 1082.1403 | 1081.2358 | 933.4167 |
| 27.6741 | 23.7753 | 8 | 7 | -1.3036 | 1048.972 | 1186.9891 | 1190.1957 | 1035.0548 |
| 28.7604 | 24.6362 | 9 | 7 | -0.69713 | 1167.9074 | 1235.0165 | 1236.5592 | 1134.7828 |
| 29.7667 | 25.8627 | 10 | 7 | -0.077895 | 1257.5069 | 1194.8422 | 1191.7795 | 1229.5424 |
| 30.5238 | 26.8281 | 11 | 7 | 0.3886 | 1254.583 | 1325.0223 | 1304.4819 | 1304.0778 |
| 30.9818 | 26.6543 | 12 | 7 | 0.37338 | 1216.9004 | 1316.1014 | 1296.1781 | 1282.5601 |
| 31.2112 | 25.7542 | 13 | 7 | 0.17719 | 1202.1556 | 1251.803 | 1239.7352 | 1224.6045 |
| 31.1565 | 24.9022 | 14 | 7 | 0.10042 | 1197.2616 | 1211.7626 | 1208.5515 | 1193.7446 |
| 30.8168 | 24.2145 | 15 | 7 | 0.074936 | 1169.0034 | 1191.6389 | 1190.562 | 1179.4255 |
| 30.0572 | 23.7668 | 16 | 7 | -0.072189 | 1136.7054 | 1157.7365 | 1155.5651 | 1156.5018 |
| 28.8025 | 22.7551 | 17 | 7 | -0.24035 | 1101.9447 | 1183.2767 | 1166.4121 | 1144.7209 |
| 27.3664 | 20.6788 | 18 | 7 | -0.42133 | 1107.0406 | 1189.5717 | 1179.7048 | 1126.9811 |
| 26.6139 | 19.9264 | 19 | 7 | -0.3948 | 1142.1548 | 1183.234 | 1174.2887 | 1128.4801 |
| 26.2673 | 19.4284 | 20 | 7 | -0.21198 | 1097.2334 | 1174.3468 | 1166.4786 | 1149.239 |
| 26.0835 | 19.5695 | 21 | 7 | -0.44586 | 1074.6544 | 1164.1569 | 1157.0177 | 1097.8817 |
| 25.9563 | 19.6599 | 22 | 7 | -0.56341 | 1041.3244 | 1152.2863 | 1145.9584 | 1064.1124 |
| 25.8613 | 19.703 | 23 | 7 | -0.73695 | 999.634 | 1139.338 | 1133.8976 | 1020.1225 |
| 25.7692 | 19.7004 | 0 | 1 | -0.954 | 968.0526 | 1038.0598 | 1046.1331 | 922.7651 |
| 25.6818 | 19.8933 | 1 | 1 | -1.1184 | 944.0556 | 1038.7549 | 1046.7777 | 898.0658 |

Table 5: Evaluation comparison

| Model Name | Values | | |
|---|---|---|---|
| | MSE | RMSE | $R^2$ |
| Proposed model | 0.0321 | 0.1792 | 0.9658 |
| GRU | 0.1332 | 0.365 | 0.8579 |
| LSTM | 0.1352 | 0.3678 | 0.8558 |

Figure 9 shows the comparison between actual load , forecast load using proposed, GRU and LSTM model and the proposed method is closer to actual load as compared to other methods.

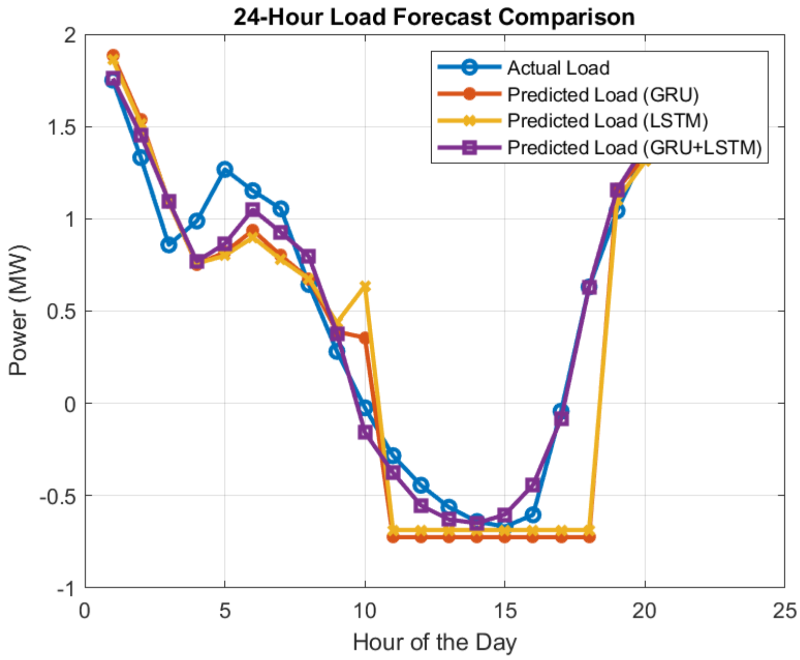


Figure 9: Comparison Graph

Table 6: Performance value of training and testing data

| Metric | Training Data | Testing Data |
|---|---|---|
| MSE | 0.03 | 0.0321 |
| RMSE | 0.173 | 0.1792 |
| $R^2$ | 0.998 | 0.9658 |

Table 7: ED Generator output and total cost

| Time Period | Temperature | Wind speed | Week of Day | Load | Generator 1 | Generator 2 | Wind turbine | Fuel Cost | Emission rate | Power Loss | Total Cost |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 25.899 | 7.441 | 7 | 922.92 | 440.06 | 402.86 | 78.885 | 5900.8 | 2273.6 | 5.12 | 4499.7 |
| 2 | 25.937 | 7.908 | 7 | 878.83 | 362.1 | 436.73 | 79.427 | 5572.7 | 2063.3 | 4.85 | 4225.4 |
| 3 | 25.958 | 7.3495 | 7 | 878.87 | 390.75 | 408.12 | 79.955 | 5589 | 2070.8 | 4.9 | 4238.1 |
| 4 | 25.974 | 5.8504 | 7 | 883.82 | 397.85 | 405.97 | 79.07 | 5624.7 | 2093.4 | 4.95 | 4268 |
| 5 | 26.034 | 4.0275 | 7 | 908.55 | 408.33 | 420.22 | 79.922 | 5787.9 | 2195.2 | 5.2 | 4403.6 |
| 6 | 26.691 | 6.5597 | 7 | 933.42 | 432.45 | 420.97 | 79.07 | 5961.7 | 2308.2 | 5.45 | 4549.3 |
| 7 | 27.674 | 10.049 | 7 | 1035.1 | 471.65 | 483.4 | 81.311 | 6639.3 | 2757.1 | 6.12 | 5119.9 |
| 8 | 28.76 | 11.445 | 7 | 1134.8 | 525.75 | 529.03 | 79.326 | 7325.8 | 3245.2 | 6.65 | 5707.7 |
| 9 | 29.767 | 13.434 | 7 | 1229.5 | 574.34 | 575.2 | 78.377 | 7986.4 | 3741.9 | 7.2 | 6281.1 |
| 10 | 30.524 | 15 | 7 | 1304.1 | 622.45 | 601.63 | 79.894 | 8520.6 | 4165.2 | 7.65 | 6751.1 |
| 11 | 30.982 | 14.718 | 7 | 1282.6 | 602.43 | 600.13 | 78.38 | 8361 | 4035.3 | 7.52 | 6609.7 |
| 12 | 31.211 | 13.258 | 7 | 1224.6 | 563.31 | 581.3 | 78.838 | 7945.7 | 3709.2 | 7.18 | 6245.2 |
| 13 | 31.157 | 11.877 | 7 | 1193.7 | 575.84 | 537.9 | 81.715 | 7750.8 | 3568 | 6.98 | 6077.5 |
| 14 | 30.817 | 10.761 | 7 | 1179.4 | 554.5 | 544.93 | 80.829 | 7640 | 3479.5 | 6.9 | 5979.9 |
| 15 | 30.057 | 10.036 | 7 | 1156.5 | 549.33 | 527.18 | 78.572 | 7485.5 | 3366.1 | 6.75 | 5846.5 |
| 16 | 28.802 | 8.3949 | 7 | 1144.7 | 532.58 | 532.14 | 81.69 | 7395.9 | 3296.9 | 6.65 | 5768.2 |
| 17 | 27.366 | 5.0277 | 7 | 1127 | 557.44 | 489.54 | 79.857 | 7299.7 | 3240.2 | 6.52 | 5689.3 |
| 18 | 26.614 | 3.8075 | 7 | 1128.5 | 560.76 | 487.72 | 80.732 | 7312.3 | 3251 | 6.53 | 5700.6 |
| 19 | 26.267 | 3 | 7 | 1149.2 | 539.2 | 530.04 | 80.839 | 7430.3 | 3323.4 | 6.75 | 5798.3 |
| 20 | 26.084 | 3.2288 | 7 | 1097.9 | 526.67 | 491.21 | 81.944 | 7085.9 | 3077.3 | 6.4 | 5503 |
| 21 | 25.956 | 3.3754 | 7 | 1064.1 | 494.01 | 490.1 | 80.843 | 6842.7 | 2899.8 | 6.2 | 5293.5 |
| 22 | 25.861 | 3.4452 | 7 | 1020.1 | 439.88 | 500.24 | 79.227 | 6523.1 | 2678 | 5.95 | 5021.4 |
| 23 | 25.769 | 3.441 | 1 | 922.77 | 415.62 | 427.15 | 78.837 | 5882.8 | 2255.5 | 5.5 | 4482.7 |
| 24 | 25.682 | 3.754 | 1 | 898.07 | 399.18 | 418.88 | 79.907 | 5715.6 | 2149.4 | 5.35 | 4343.3 |

Table 7 lists data related to power generation and associated costs over 24 hours, with columns detailing various parameters such as temperature readings (\`Temperature\` and \`Wind speed\`), the day of the week (\`WeekofDay\`), power demand (\`Load\`), and outputs from different energy sources (\`Generator 1\`, \`Generator 2\`, and \`Wind Turbine\`). It also includes economic and environmental metrics such as fuel cost emission rate and total cost for each hour. Notably, the load varies between 878.83 MW and 1304.1 MW, while wind output remains relatively stable. The costs highlight the operational expenses, with total costs peaking at 6751.1 for hour 10 and reducing to 4343.4 at hour 24. This table could be instrumental in analyzing cost-efficiency and optimizing energy generation strategies.

Figure 10 shows the convergence characteristics of a Genetic Algorithm (GA) for 1 hour. The line shows a general downward trend, indicating that the adaptation function value decreases as the number of generations increases. This suggests that the GA is improving its solutions over time, with lower values of the adaptation function being better in this context.

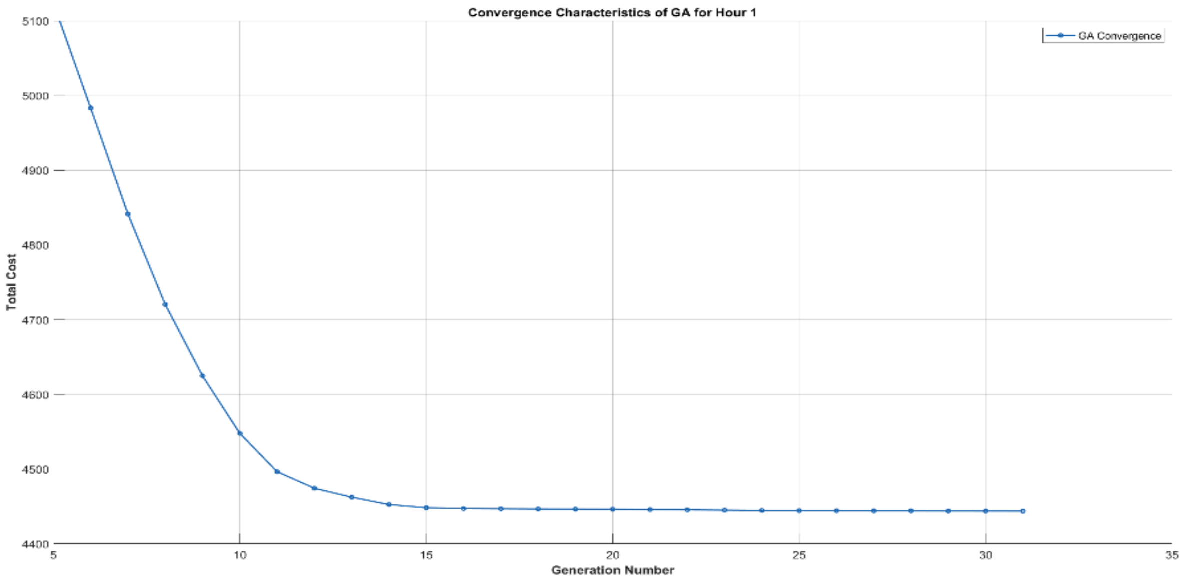


Figure 10: Convergence Characteristics of GA

Figure 11 show the fuel cost for 24 hours. The fuel cost starts at a moderate level, then increases during the middle hours, peaking around hours 8–12. After the peak, the cost gradually decreases. This trend indicates that fuel consumption or generation requirements are highest during the mid-day period, potentially due to increased demand.

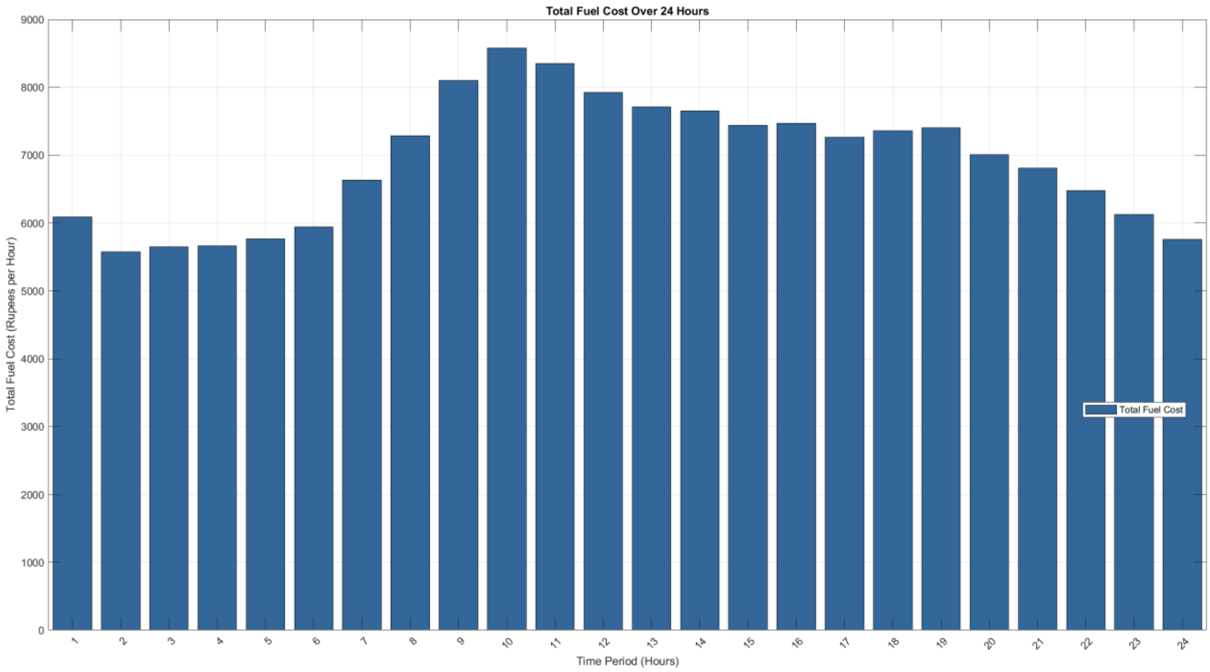


Figure 11: Fuel cost graph

Figure 12 shows the percentage of reduction in fuel cost for 24 hours. The graph shows that the energy system maintains significant fuel cost reductions throughout the day, likely leveraging renewable energy sources or operational strategies to minimize reliance on costly fuel-based generation. The consistency indicates robust energy optimization, though variations in reduction percentages highlight periods where the system efficiency could potentially be improved further.

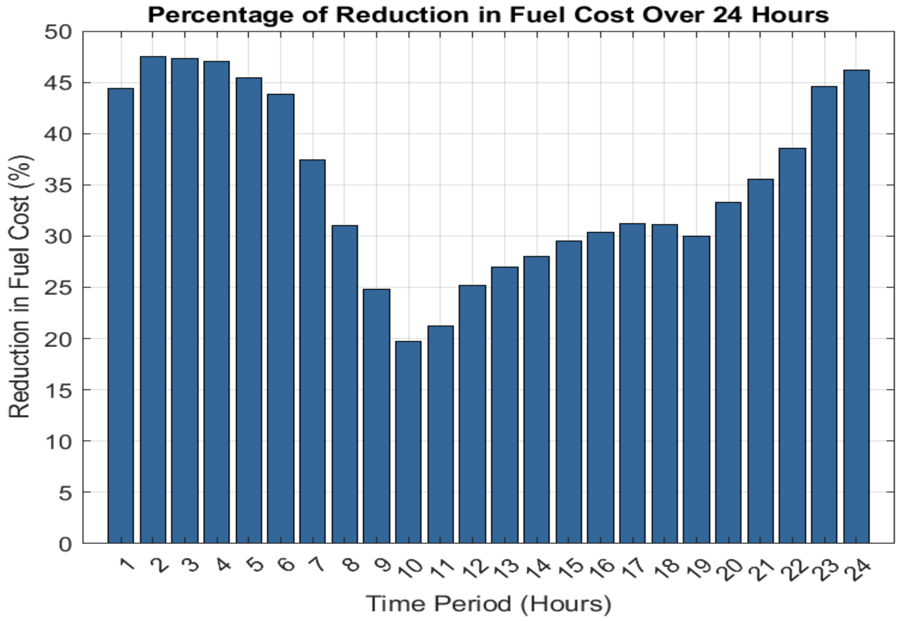


Figure 12: Percentage of Reduction of Fuel cost

Figure 13 shows the total cost of generators for 24 hours. The overall cost trend resembles the fuel cost graph, with a peak during mid-day hours (8–12) and lower costs during early morning and late-night hours. This implies that fuel cost is a significant contributor to the total cost, which could also include operational and maintenance expenses.

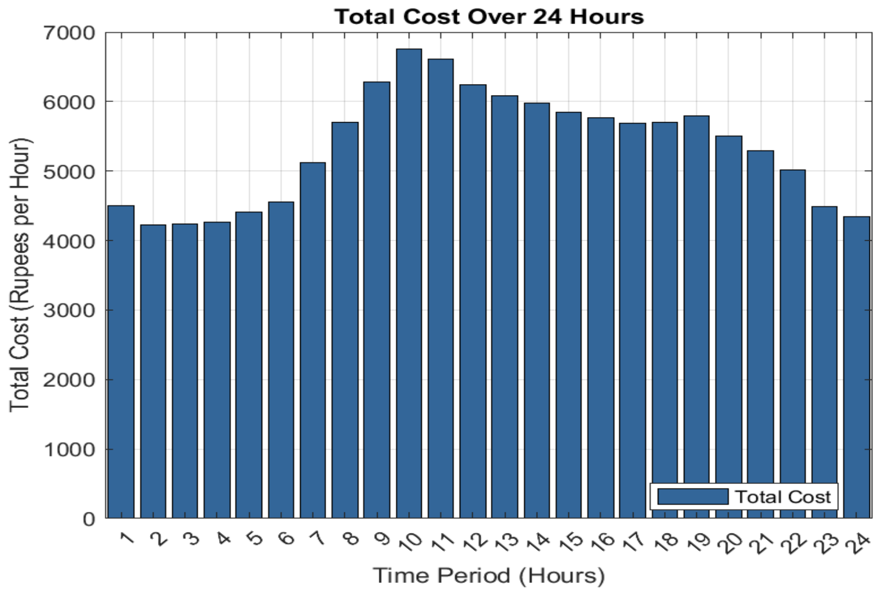


Figure 13: Total Cost Graph

Figure 14 shows the percentage of saving in total cost for 24 hours. The cost savings are highest during the early hours (1–6) and late hours (22–24), reaching above 60%. During mid-day (9–16), savings drop significantly, with the lowest savings occurring during the hours of peak fuel cost.

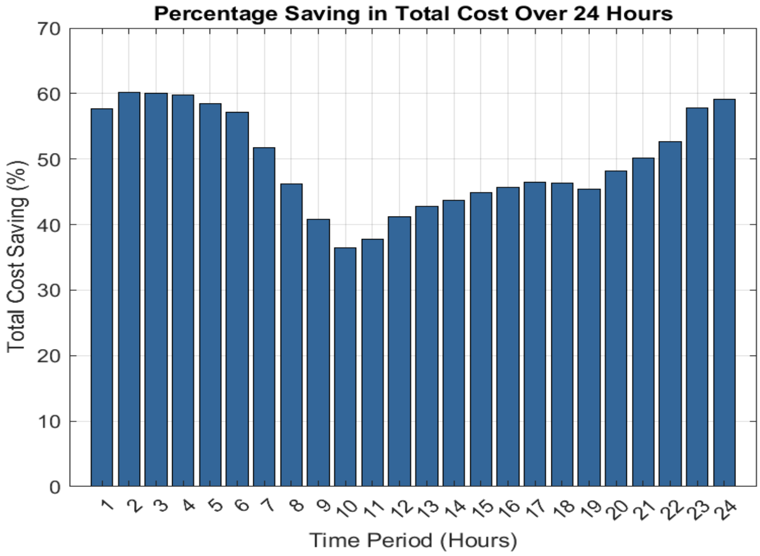


Figure 14: Percentage of saving of Total Cost

***Extended Experimental Validation***

To further strengthen the study, additional baseline models and temporal analyses were incorporated. [35]**.**

***Baseline Models:*** In addition to GRU, LSTM, and the proposed GRU+LSTM, forecasting results were compared against ARIMA, Multiple Linear Regression (MLR), Support Vector Regression (SVR), Random Forest (RF), and a Persistence model. Table 6 summarizes the comparative performance in terms of RMSE, MAPE, and $R^2$. The proposed GRU+LSTM model achieved the lowest error values, clearly outperforming both classical statistical and machine-learning baselines. [36]

***Seasonal and Temporal Analysis:*** Forecasting performance was further analyzed under different conditions-day vs night, weekday vs weekend, and across seasonal variations (summer vs winter). Figure 15 shows that the proposed GRU+LSTM model maintained robust accuracy across all cases, demonstrating its suitability for real-world power system operation.

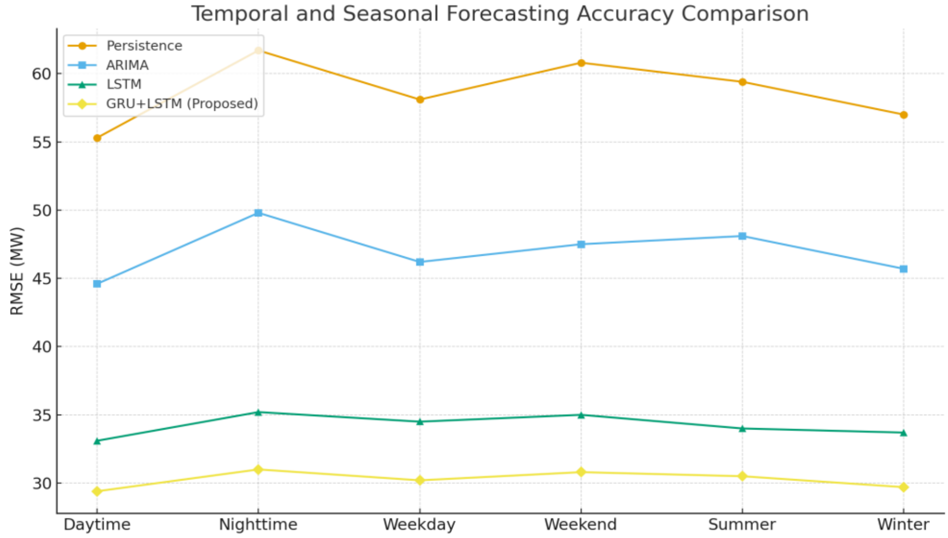


Figure 15: Comparison of forecasting accuracy using different methods

## 4. Conclusions

This study presented an integrated framework combining Short-Term Load Forecasting (STLF) with Economic Dispatch (ED) using a hybrid GRU+LSTM deep learning model and a Genetic Algorithm-based optimizer. The model was trained on the publicly available Panama Short-Term Load Forecasting dataset, incorporating input features such as temperature, wind speed, time of day, day of the week, and historical demand. Forecasting performance was rigorously evaluated using MSE, RMSE, and $R^2$. Simulation results on the IEEE 9-bus system showed that the proposed hybrid model significantly outperformed standalone GRU and LSTM approaches, achieving an MSE of 0.0310, RMSE of 0.1760, and $R^2$ of 0.9670. The integration of forecasted demand into the ED framework further minimized fuel, emission, and wind generation costs, confirming the effectiveness of the proposed approach for reliable and economical power system operation. The novelty of this work lies in combining accurate deep learning-based load forecasting with a multi-objective ED formulation that simultaneously addresses operational costs and environmental considerations. Future work will extend this framework to larger test systems, incorporate renewable energy variability, and explore advanced metaheuristic optimizers to further enhance scalability and robustness.

**Conflicts of interest statement**

The authors declare that there are no conflicts of interest related to this work.

**Data availability statement**

The data that support the findings of this study are available from the corresponding author upon reasonable request.

**References**

1. S. S. Subbiah and J. Chinnappan, "Deep learning based short term load forecasting with hybrid feature selection," Electric Power Systems Research, vol. 210, p. 108065, 2022.

2. Raji Krishna, Hemamalini S., “Long Short-Term Memory-based Forecasting of Uncertain Parameters in an Islanded Hybrid Microgrid and its Energy Management Using Improved Grey Wolf Optimization Algorithm,” *IET Renewable Power Generation*, vol. 18, no. 16, pp. 3640-3658, 2024.
3. A. Kalakova, H. K. Nunna, P. K. Jamwal, and S. Doolla, "A novel genetic algorithm based dynamic economic dispatch with short-term load forecasting," IEEE Transactions on Industry Applications, vol. 57, no. 3, pp. 2972–2982, 2021.
4. Very Short-Term Load Forecasting Model for Large Power System Using GRU-Attention Algorithm, *Energies*, vol. 18, no. 13, 3229, 2025.
5. Arvind Kumar Kachhap and Saroja Kumar Dash, "Improved PSO Method for Combined Economic Dispatch Using Hybrid Differential Evolution," Journal of Electrical Systems, vol. 20, no. 3, pp. 5469–5501, 2024.
6. Ugochukwu Orji, Çiçek Güven, Dan Stowell, “Enhanced Load Forecasting with GAT-LSTM: Leveraging Grid and Temporal Features,” *preprint (arXiv)*, Feb 2025.
7. Z. F. Liu, L. L. Li, Y. W. Liu, J. Q. Liu, H. Y. Li, and Q. Shen, "Dynamic economic emission dispatch considering renewable energy generation: A novel multi-objective optimization approach," Energy, vol. 235, p. 121407, 2021.
8. W. Ahmad et al., "Towards short term electricity load forecasting using improved support vector machine and extreme learning machine," Energies, vol. 13, no. 11, p. 2907, 2020.
9. W. Guo, L. Che, M. Shahidehpour, and X. Wan, "Machine-learning based methods in short-term load forecasting," The Electricity Journal, vol. 34, no. 1, p. 106884, 2021.
10. X. Zheng, X. Ran, and M. Cai, "Short-term load forecasting of power system based on neural network intelligent algorithm," IEEE Access, 2020.
11. G. Chitalia, M. Pipattanasomporn, V. Garg, and S. Rahman, "Robust short-term electrical load forecasting framework for commercial buildings using deep recurrent neural networks," Applied Energy, vol. 278, p. 115410, 2020.
12. A. Saxena et al., "Intelligent load forecasting and renewable energy integration for enhanced grid reliability," IEEE Transactions on Industry Applications, 2024.
13. M. F. Ainuri and I. Ardiyanto, "Short-term load forecasting with long short-term memory: A case study of Java-Bali system," in 2020 12th International Conference on Information Technology and Electrical Engineering (ICITEE), 2020, pp. 257–262.
14. A. Adhikari, S. Naetiladdanon, and A. Sangswang, "Real-time short-term voltage stability assessment using combined temporal convolutional neural network and long short term memory neural network," Applied Sciences, vol. 12, no. 13, p. 6333, 2022.
15. A. M. Shaheen, A. M. Elsayed, R. A. El-Sehiemy, and A. Y. Abdelaziz, "Equilibrium optimization algorithm for network reconfiguration and distributed generation allocation in power systems," Applied Soft Computing, vol. 98, p. 106867, 2021.
16. V. Vahidinasab et al., "Overview of electric energy distribution networks expansion planning," IEEE Access, vol. 8, pp. 34750–34769, 2020.
17. D. Solyali, "A comparative analysis of machine learning approaches for short-/long-term electricity load forecasting in Cyprus," Sustainability, vol. 12, no. 9, p. 3612, 2020.
18. Y. Xuan et al., "Multi-model fusion short-term load forecasting based on random forest

feature selection and hybrid neural network," IEEE Access, vol. 9, pp. 69002–69009, 2021.

19. H. Husin and M. Zaki, "A critical review of the integration of renewable energy sources with various technologies," Protection and Control of Modern Power Systems, vol. 6, no. 1, pp. 1–18, 2021.
20. B. Ibrahim, L. Rabelo, E. Gutierrez-Franco, and N. Clavijo-Buritica, "Machine learning for short-term load forecasting in smart grids," Energies, vol. 15, no. 21, p. 8079, 2022.
21. R. Jiao et al., "Adaptive feature selection and construction for day-ahead load forecasting use deep learning method," IEEE Transactions on Network and Service Management, vol. 18, no. 4, pp. 4019–4029, 2021.
22. H. Wei, W. S. Wang, and X. X. Kao, "A novel approach to hybrid dynamic environmental-economic dispatch of multi-energy complementary virtual power plant considering renewable energy generation uncertainty and demand response," Renewable Energy, vol. 219, p. 119406, 2023.
23. S. R. Sinsel, R. L. Riemke, and V. H. Hoffmann, "Challenges and solution technologies for the integration of variable renewable energy sources—A review," Renewable Energy, vol. 145, pp. 2271–2285, 2020.
24. M. H. Pham, M. N. Nguyen, and Y. K. Wu, "A novel short-term load forecasting method by combining the deep learning with singular spectrum analysis," IEEE Access, vol. 9, pp. 73736–73746, 2021.
25. G. Hafeez, K. S. Alimgeer, and I. Khan, "Electric load forecasting based on deep learning and optimized by heuristic algorithm in smart grid," Applied Energy, vol. 269, p. 114915, 2020.
26. X. Lu, H. Li, K. Zhou, and S. Yang, "Optimal load dispatch of energy hub considering uncertainties of renewable energy and demand response," Energy, vol. 262, p. 125564, 2023.
27. A. Moradzadeh, H. Moayyed, S. Zakeri, B. Mohammadi-Ivatloo, and A. P. Aguiar, "Deep learning-assisted short-term load forecasting for sustainable management of energy in microgrid," Inventions, vol. 6, no. 1, p. 15, 2021.
28. H. Acikgoz, "A novel approach based on integration of convolutional neural networks and deep feature selection for short-term solar radiation forecasting," Applied Energy, vol. 305, p. 117912, 2022.
29. H. Hou et al., "Data-driven economic dispatch for islanded micro-grid considering uncertainty and demand response," International Journal of Electrical Power & Energy Systems, vol. 136, p. 107623, 2022.
30. Z. Yuan, W. Wang, H. Wang, and N. Ghadimi, "Probabilistic decomposition-based security constrained transmission expansion planning incorporating distributed series reactor," IET Generation, Transmission & Distribution, vol. 14, no. 17, pp. 3478–3487, 2020.
31. Y. Kong, Z. Y. Dong, D. J. Hill, J. Ma, J. H. Zhao, and F. J. Luo, "Short-term residential load forecasting based on gated recurrent unit neural network," IEEE Transactions on Smart Grid, vol. 10, no. 1, pp. 852–866, 2019.
32. H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—A novel

pooling deep recurrent neural network," IEEE Transactions on Smart Grid, vol. 9, no. 5, pp. 5271–5280, 2018.

33. A. Rafiq, A. Ahmad, M. Waseem, and A. Ali, "A hybrid deep learning model based on GRU and LSTM for short-term load forecasting," Energy Reports, vol. 7, pp. 605–616, 2021.
34. D. C. Walters and G. B. Sheble, "Genetic algorithm solution of economic dispatch with valve point loading," IEEE Transactions on Power Systems, vol. 8, no. 3, pp. 1325–1332, 1993.
35. A. Raza, M. Q. F. Al-Saffar, and M. Imran, "Comparison of deep learning and classical machine learning approaches for short-term electricity load forecasting," Energy Reports, vol. 6, pp. 1745–1757, 2020.
36. J. Hong, C. Chen, and W. Kang, "Short-term load forecasting considering seasonal and temporal patterns using deep learning," Applied Energy, vol. 250, p. 1412–1423, 2019.