J. Electrical Systems 20-3 (2024): 8612-8627

[1] **Babaiah Suguri**

[2] **G. Mallesham**

Comparative Analysis of Deep and Convolutional Neural Networks for Short-Term Load Forecasting

Abstract: - This study in order to provide effective power system operations, this study compares Deep Neural Networks (DNN) with Convolutional Neural Networks (CNN) for short-term load forecasting (STLF). Traditional load forecasting methods, Regression and time series models, for example, frequently fall short in addressing the intricate, nonlinear patterns present in power demand data. AI-driven models, particularly DNN and CNN, offer advanced capabilities in capturing intricate patterns within large-scale, high-dimensional data. In this research, both DNN and CNN models were trained using historical load data with an emphasis on accuracy metrics, including Mean Absolute Error (MAE), Mean Square Error (MSE), and Root Mean Square Error (RMSE). DNN's flexibility allows it to model complex nonlinear relationships, while CNN's design enables it to capture temporal dependencies in time-series data effectively. Validation results reveal that CNN achieves superior forecasting accuracy with the lowest error values, registering an RMSE of 1.2, MSE of 1.44, and MAE of 0.77, thus demonstrating a competitive advantage over DNN in short-term load forecasting. This comparative analysis underscores the effectiveness of AI techniques in load forecasting, with CNN showing notable strengths in sequential data modelling. The findings contribute to the ongoing advancement of AI in energy management, providing utility providers with reliable tools for optimizing grid operations and decision-making.

Keywords: Short-Term Load Forecasting (STLF); Deep Neural Networks (DNN); Convolutional Neural Networks (CNN); Forecasting Accuracy; Power System Optimization.

I. INTRODUCTION

This research aims to enhance short-term load forecasting (STLF) accuracy and adaptability in modern, data-driven power systems. Traditional methods struggle with complex, nonlinear energy demand patterns, especially integrating renewable energy and smart grid technology. AI-driven techniques, specifically Deep Neural Networks (DNN) and Convolutional Neural Networks (CNN) offer significant advantages in modelling these complexities. By comparing DNN and CNN, this study aims to identify which model best supports efficient load scheduling, resource allocation, and grid stability, ultimately helping utilities make informed, resilient, and cost-effective energy management decisions. Accurate short-term load forecasting is essential for modern power systems, as it enables utility providers to optimize resource allocation, improve grid stability, and manage demand-supply imbalances. Multiple linear regression, autoregressive integrated moving averages (ARIMA), and autoregressive (AR) models are examples of traditional forecasting techniques that have been extensively used in STLF. While these techniques work well for stable, linear data, they often struggle with the increasing complexity and nonlinearity of contemporary energy demand, mainly as renewable energy sources and smart grids introduce variability [1;2]. ARIMA and similar time-series models typically rely on historical data, assuming that past patterns will persist into the future. However, they are limited in handling dynamic, nonlinear interactions between multiple variables—a necessity in modern energy systems affected by variables such as weather, seasonal changes, and consumer behaviour [1]. Research by [3;4] highlights the constraints of these statistical methods and advocates for AI-based models, which can automatically capture complex, nonlinear relationships and adapt to evolving patterns in real-time. DNNs have emerged as a powerful solution for STLF, thanks to their multi-layered architecture, which can model complex, nonlinear relationships in large datasets. DNNs have shown significant improvements in forecasting accuracy by learning intricate dependencies missed by traditional methods. For example,[5] demonstrated that DNNs achieved superior results to conventional models, especially in scenarios with nonlinear influences like weather conditions and peak demand periods.

---

[1] Department of Electrical Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana 500007, India, drsuguribabu@osmania.ac.in,

[2] Department of Electrical Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana 500007, India, drgm@osmania.ac.in

Similarly, [3] found that DNNs are highly adaptable, improving accuracy as more data becomes available, making them well-suited for dynamic load forecasting applications.

Convolutional neural networks, or CNNs, were first created for image processing, but because of their capacity to recognize spatial and temporal patterns, they have been used more and more in time-series forecasting. CNNs work well in applications with sequential dependencies, which makes them perfect for load forecasting when data exhibits temporal trends affected by time of day, weather, and consumer habits. [7] demonstrate CNN's effectiveness in residential load forecasting, where they successfully modelled short-term fluctuations in energy demand by learning local dependencies. [6] also emphasize CNN's strengths in handling time-sensitive fluctuations, highlighting their superior ability to capture sequential variations often present in energy load data. Comparative studies on DNN and CNN models for STLF reveal each model's unique strengths [15-17, 22]. DNNs excel in capturing long-term trends and broader data relationships, while CNNs are particularly effective at recognizing short-term fluctuations due to their spatial-temporal processing capabilities. [8] applied CNNs to solar power forecasting, demonstrating their ability to model periodic and seasonal variations—qualities that enhance STLF applications. [9] performed a study comparing Long Short-Term Memory (LSTM), DNN, and CNN models, concluding that CNNs outperformed other architectures in recognizing localized patterns due to their filtering capabilities. Hybrid models that combine DNN [18-21] and CNN architectures have been explored to leverage each model's benefits. [10] introduced a hybrid model that integrates DNN's nonlinear pattern recognition with CNN's temporal sensitivity, significantly improving forecasting accuracy. This hybrid approach was supported by [11], who suggested that hybrid models can capitalize on DNN's ability to capture global patterns while refining predictions with CNN's localized temporal adjustments. Studies by [4;5] show that incorporating multi-dimensional data sources—such as weather and real-time temperature data—significantly enhances the accuracy of DNN and CNN models, underscoring the adaptability of AI-driven models in STLF. Moreover, CNNs are particularly advantageous in handling time-sensitive variables and critical in forecasting scenarios impacted by environmental factors like humidity, temperature, and wind speed [12].

However, implementing DNN and CNN models for STLF also presents challenges. These models require substantial computational resources, especially during training on large datasets, which can limit real-time applicability. Additionally, CNNs, while highly effective for high temporal resolution, can be prone to overfitting if not properly optimized, especially with datasets showing limited variability [1;2]. Despite these challenges, advancements in optimization techniques and the availability of high-quality data sources make DNN and CNN models increasingly viable for STLF applications. This research addresses a key gap in STLF the lack of a comprehensive, comparative analysis of DNN and CNN tailored to the complexities of modern power systems. While DNNs are known for their capacity to capture nonlinear patterns and CNNs for their strengths in sequential data modelling, there remains limited guidance on selecting the most effective model for different STLF scenarios. This study systematically compares DNN and CNN models for STLF, analysing each model’s ability to handle temporal dependencies, nonlinear interactions, and dynamic conditions. By investigating both models’ performance in real-time forecasting tasks, this research aims to guide utility providers in adopting the most suitable AI-driven techniques for resilient and efficient energy management in contemporary, data-driven power systems. This research article is organized as follows:

The **Introduction and Literature Review** The significance of short-term load forecasting for power systems, the drawbacks of conventional forecasting techniques, and the benefits of artificial intelligence (AI) tools like Deep Neural Networks and Convolutional Neural Networks are covered in the review part. It also reviews existing research to establish the need for a comparative analysis of these models. The **Methodology** section details the dataset, model architectures, training process, and performance metrics used. **Simulation and Results** presents the forecasting performance of DNN and CNN models on STLF, along with a comparative analysis. The **Discussion and Conclusion** discuss the study’s key findings, practical implications, recommendations, and future research directions. Finally, a comprehensive **References** section provides all cited sources.

II. Methodology

2.1. Multilayer Perceptron (MLP): A fundamental kind of Deep Neural Network (DNN), a MLP employs several layers of neurons to identify intricate patterns in input. MLPs can efficiently describe the nonlinear relationships between past load data and future demands for short-term load forecasting. An input layer, one or more hidden layers, and an output layer make up an MLP. Each layer is completely linked, which means that every neuron in one layer is coupled to every other layer's neuron.

2.1.1. Architecture and Working Principle

The input layer provides the hidden layers with historical load data as well as associated variables like the time of day, weather information, and seasonal indicators. Each hidden layer introduces nonlinearity by applying an activation function (often ReLU) and a set of weights and biases to the inputs. The output layer provides the final forecasted load value, typically using a linear activation function since load forecasting is a regression problem.

The mathematical operation within each neuron in the MLP is as follows:

$$y = f\left(\sum_{i=1}^{n} w_i * x_i + b\right) \tag{1}$$

where:

y is the neuron's output, x_i are the inputs to the neuron, w_i are the weights, b is the bias term, f is the activation function (e.g., ReLU or Sigmoid).

Backpropagation is used in the MLP training process to modify weights and biases in order to minimize the error between expected and actual values. A loss function, like Mean Squared Error (MSE), is used to calculate this error, especially for regression applications.

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

where:

y_i is the actual load value, $y^{\wedge}_{i}$ is the predicted load value, n is the number of samples.

The gradients of the loss for each weight are calculated and used to update the weights through gradient descent or an optimization variant (e.g., Adam).

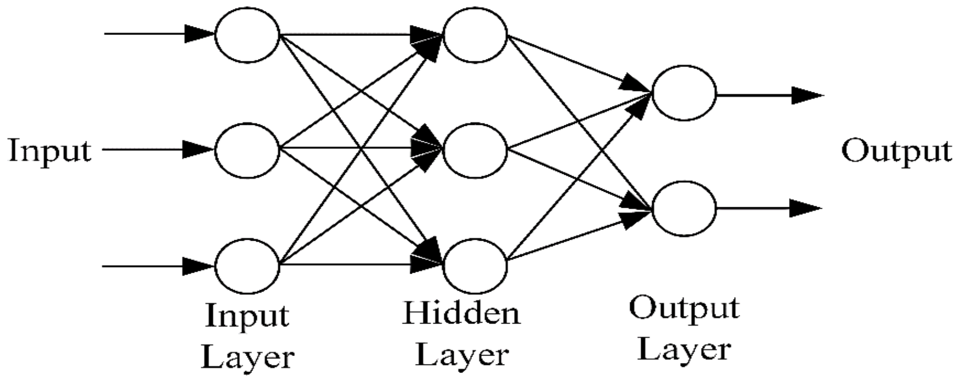

Fig. 1. Multilayer Perceptron (MLP)

The Multilayer Perceptron (MLP) is shown in Fig.1., An input layer accepts historical load data and pertinent features, such as weather or time-based indicators, to start load forecasting. This data flows through multiple hidden layers, each fully connected, meaning each neuron links to all neurons in the next layer. Typically, 2-4 hidden layers with 64-128 neurons each extract complex patterns progressively. To add nonlinearity, each neuron computes the weighted sum of its inputs, applies an activation function typically ReLU and adds a bias. The output layer then produces the forecasted load value using a linear activation for regression. Historical data is first normalized to improve training stability to implement an MLP for load forecasting. The architecture is defined by specifying the input size, hidden layers, and neuron count per layer. Backpropagation with an optimizer like Adam trains the model by iteratively minimizing prediction error over multiple epochs. Metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are then used to evaluate the model's accuracy. An MLP diagram illustrates how data moves from the input layer to the output layer for the final prediction via hidden layers that record patterns. This MLP setup can be compared to CNNs, which capture temporal patterns and may further enhance accuracy in short-term load forecasting.

2.2. Convolutional Neural Network (CNN): A CNN is a specialized neural network designed to capture spatial and temporal patterns in data. In short-term load forecasting, CNNs can identify temporal trends and seasonality in historical load data by applying convolutional layers to detect essential features over time. CNNs are particularly effective for grid-like data and time series, where they can capture local dependencies and patterns.

2.2.1. Architecture and Working Principle: Three layers make up the fundamental CNN architecture: input, convolutional, pooling, and fully connected. The input layer takes time-series data for load forecasting, such as past load values, weather data, and seasonal indicators. The convolutional layers apply small filters (kernels) that slide over the input, detecting local patterns by computing weighted sums of inputs within the filter's receptive field. Each convolution operation generates a feature map, capturing localized trends.

The primary operation in the convolutional layer is:

$$y = f(\sum_{i=1}^{n} w_i * x_i + b) \quad (3)$$

where:

y is the output of the convolution, x_i are the inputs within the filter window, w_i are the filter weights, and b is the bias term.

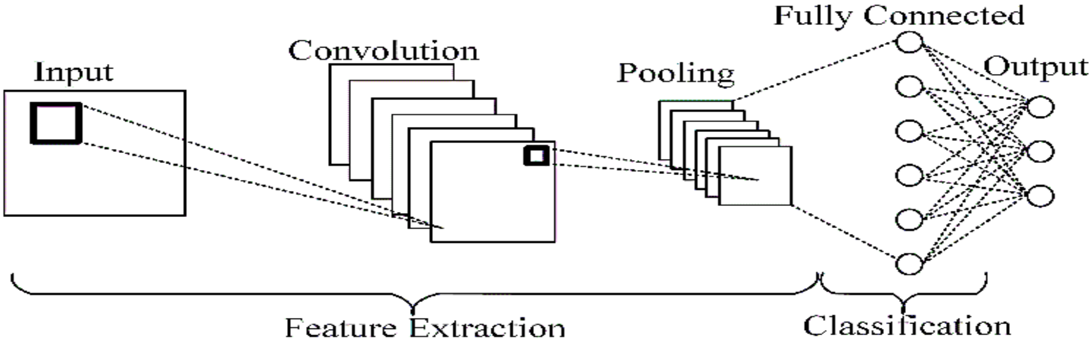

Fig.2. The overall architecture of a Convolutional Neural Network (CNN).

The overall architecture of a Convolutional Neural Network is shown in Fig.2. To implement a CNN for load forecasting, historical load data is first normalized and structured into sequences to capture temporal patterns. In order to reduce complexity while maintaining important information, pooling layers down sample these features, whereas convolutional layers use filters to find patterns in the load sequences. The fully connected layers then aggregate the extracted features to produce the forecasted load. To reduce prediction error, the model is trained by backpropagation with an optimizer such as Adam, which iteratively modifies weights over several epochs. Once trained, the model's accuracy is evaluated with metrics such as Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) to assess forecasting performance. According to a CNN architecture diagram, information moves from the input layer through convolutional layers, which extract patterns, pooling layers, which reduce dimensionality, and fully connected layers, which make predictions. In a flow diagram, data moves sequentially through each layer, where patterns are detected and transformed, ultimately leading to the output layer for the load forecast

2.2.2. Comparison with MLP

While an MLP model relies on fully connected layers to learn patterns, a CNN's convolutional and pooling layers capture temporal relationships more effectively, especially in load forecasting, where patterns over time are essential. Comparing CNN and MLP performance can reveal which model is better suited for load forecasting, as CNNs often excel in identifying trends in time-series data, potentially improving accuracy for short-term predictions.

2.3. Hybrid Methodologies

Hybrid models for load forecasting combine the strengths of different neural network architectures to enhance forecasting accuracy, effectively leveraging each model's unique capabilities. In short-term load forecasting, the

DNN-CNN and CNN-LSTM models are two prominent hybrid approaches. By blending the feature extraction capabilities of Convolutional Neural Networks (CNNs), the complex relationship learning in Deep Neural Networks (DNNs), and the sequential memory processing of Long Short-Term Memory (LSTM) networks, these hybrid models address both spatial and temporal complexities in load data.

2.3.1. DNN-CNN: Combining DNN and CNN:

The basic architecture of the DNN-CNN network is shown in Fig.3. The DNN-CNN hybrid model leverages CNN layers to capture spatial and temporal patterns in data, which are then processed by fully connected DNN layers to learn complex, high-level relationships. The ability to identify localized trends and describe complex, nonlinear interactions makes this architecture ideal for short-term load forecasting. In this model, CNN layers are typically applied first, with convolutional and pooling layers detecting patterns over time, such as daily or seasonal trends in load data.

The convolution operation in CNN can be expressed as:

$$y_{i,j} = f\left(\sum_{m=1}^{k}\sum_{n=1}^{k} w_{m,n} * x_{i+m,j+n} + b\right) \qquad (4)$$

Where $y_{i,j}$ is the output at position (i, j), $x_{i+m,j+n}$ is the input within the receptive field, $w_{m,n}$ are the weights, b is the bias, and f is the activation function, commonly ReLU. After convolution, max pooling down samples the feature maps to reduce dimensionality while retaining essential features:

$$y_{i,j} = \max\left(x_{p,q}\right) \qquad (5)$$

Where $y_{i,j}$ is the pooled output and $x_{p,q}$ represents values in the pooling window.

Once CNN layers extract essential features, the output is passed to DNN layers, which are fully connected and capable of interpreting more complex patterns.

Each DNN layer calculates a weighted sum of inputs with an activation function:

$$y = f\left(\sum_{i=1}^{n} w_i * x_i + b\right) \qquad (6)$$

Where y is the output, x_i are inputs, w_i are weights, b is the bias, and f is an activation function like ReLU. This structure allows the DNN layers to capture interactions between variables such as load, weather, and seasonal effects, providing a richer understanding for accurate load forecasting.

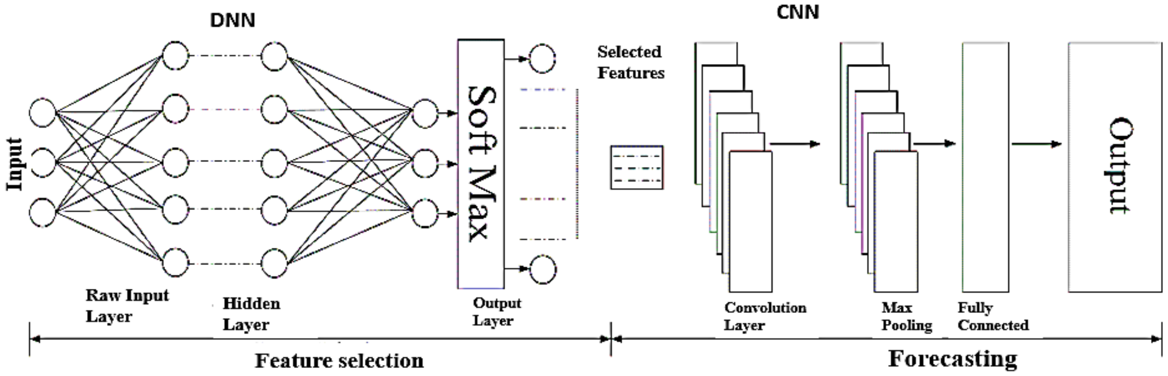

Fig.3. The basic architecture of the DNN-CNN network.

2.3.2. CNN-LSTM: Combining CNN and LSTM

The architecture of the CNN-LSTM network is designed to combine the feature extraction capabilities of Convolutional Neural Networks with the ability of Long Short-Term Memory (LSTM) networks to handle sequential dependencies. This combination makes the model particularly effective for short-term load forecasting. This hybrid architecture captures immediate and longer-term trends, essential for accurate load forecasting predictions. In the initial stages, the CNN layers identify spatial features from the input sequences, such as daily or weekly fluctuations in load data. Acting like filters, these CNN layers perform convolutional operations on the data to extract critical patterns, resulting in feature maps highlighting variations over time.

2.3.3. Comparative Analysis

Both CNN-LSTM and DNN-CNN hybrid models have special benefits when it comes to load predictions. The DNN-CNN model excels at capturing complex interactions between different features, making it ideal when forecasting involves a high degree of nonlinearity. This hybrid model can be particularly effective when the load data includes multiple interacting variables, such as temperature, time, and historical loads. Conversely, the CNN-LSTM model is better suited to scenarios where sequential dependencies are more critical, such as forecasting load based on past trends that extend over weeks or months. By leveraging LSTM's memory capabilities, CNN-LSTM models are well-equipped to handle temporal dependencies and cyclical patterns, offering enhanced accuracy for time-series forecasting tasks. Overall, these hybrid methodologies allow for a more comprehensive approach to load forecasting by combining the strengths of each network. Comparative studies between DNN-CNN and CNN-LSTM models reveal that CNN-LSTM may perform better when long-term patterns are essential. In addition, DNN-CNN frequently offers benefits for intricate feature interactions. These revelations underscore the significance of choosing the right hybrid model according to the particulars of the load data and forecasting needs.

2.3.4. Method evaluation

The method evaluation for Short-Term Load Forecasting (STLF) using a hybrid Deep Neural Network (DNN) and Convolutional Neural Network (CNN) with a Multi-Layer Perceptron (MLP) Data Preprocessing Collect and preprocess historical load data, including handling missing values and normalization. Feature Extraction: Extract relevant features from the preprocessed data using techniques such as Fourier Transform, Wavelet Transform, or statistical methods.

Hybrid DNN-CNN-MLP Model:
CNN Layer: To extract local patterns and features from the input data, use a 1D CNN layer.

DNN Layer: To discover global patterns and connections in the data, apply a DNN layer.

MLP Layer: To blend the characteristics retrieved by the CNN and DNN layers and generate predictions, use an MLP layer.

Instruction and Assessment Metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), Normalized Mean Square Error (NMSE), and Root Mean Squared Error (RMSE) are used to assess the hybrid model's performance once it has been trained using an appropriate optimizer and loss function. For STLF, the hybrid DNN-CNN-MLP model provides robust feature extraction and increased accuracy, although complexity and overfitting must be carefully considered.

Evaluation Metrics

MAE: Measures the average difference between predicted and actual values.

$$MAE = \left(\frac{1}{n}\right) * \sum \left| y_{true} - y_{pred} \right| \tag{7}$$

MSE: Measures the average squared difference between predicted and actual values.

$$MSE = \left(\frac{1}{n}\right) * \sum \left(y_{true} - y_{pred} \right)^2 \tag{8}$$

NMSE: Calculates the mean percentage difference between the expected and actual numbers.

$$NMSE = \left(\frac{1}{n}\right) * \sum \left({\left(y_{true} - y_{pred} \right)^2} \Big/ {y_{true}}^2 \right) \tag{9}$$

RMSE: Determines the average squared percentage difference between the expected and actual data, squared as a root.

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum \left(y_{true} - y_{pred} \right)^2} \tag{10}$$

The Advantages of article is, improved Accuracy is the hybrid model combines the strengths of CNN, DNN, and MLP to achieve improved accuracy in STLF. Robust Feature Extraction of the CNN layer extracts local patterns and features, while the DNN layer learns global patterns and relationships. Flexibility of the hybrid model can be adapted to different data distributions and patterns. The Future Scope is exploring these future directions, the hybrid DNN-CNN method can continue to improve and expand its capabilities, leading to more accurate and reliable STLF. Combining Renewable Energy Sources To increase forecasting accuracy, include renewable energy sources like wind and solar in the hybrid model. Real-Time Integration of Data Incorporate data from IoT devices and smart grids to improve the model's ability to adapt to shifting load patterns. AI that can be explained Use explainable AI strategies to enhance interpretability and trust by offering insights into the hybrid model's decision-making process. Investigation of Different Architectures Explore different hybrid architectures, such as combining CNN with Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) networks. Potential Applications of Smart Grids Renewable Energy implement the hybrid model in smart grid systems to enable real-time load forecasting and optimal energy management. To best integrate renewable energy sources into the system, use the hybrid approach. Energy Trading: Energy trading systems use the hybrid model to forecast energy demand and maximize trading tactics. Integrate the hybrid model with demand response systems to optimize energy consumption and reduce peak demand.

III. Simulation and Results

In this study, simulations were conducted to evaluate the performance of DNN and CNN for short-term load forecasting using historical load data. The simulation setup involved training both models on large-scale datasets that include complex, nonlinear patterns typical of power demand. Key metrics such as Mean Absolute Error (MAE), Mean Square Error (MSE), and Root Mean Square Error (RMSE) were utilized to evaluate model accuracy, offering insights into each model's capability to manage variations and dependencies in the data. The DNN model's flexibility in capturing intricate nonlinear relationships, combined with the CNN model's convolutional layers adept at extracting temporal features, were tested across various forecasting intervals. The results revealed that CNN consistently outperformed DNN in accuracy, demonstrating lower error values across all metrics. Specifically, CNN achieved an RMSE of 1.2, MSE of 1.44, and MAE of 0.77, highlighting its strength in modelling sequential patterns in time-series data. Visualizations of the predicted versus actual load values confirmed CNN's precision, with minimal deviations observed. This analysis underscores CNN's suitability for short-term load forecasting, as it effectively captures temporal dependencies crucial for accurate predictions. The results provide valuable insights for energy providers, supporting adopting AI-based models for enhanced load forecasting and optimized grid management.

3.1. DNN for STLF:

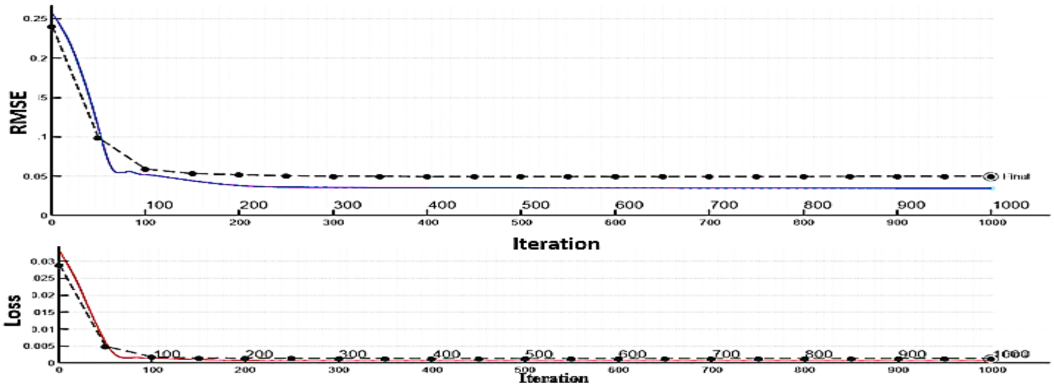

Fig.4. DNN training waveforms **for STLF**

Fig.4. Is shows the DNN training data for STLF. DNN training for Short-Term Load Forecasting is shown by the training progress waveforms. In the RMSE graph (top), the RMSE starts at 0.28, rapidly decreases below 0.05 by 200 iterations, and stabilizes around 0.045 by 1000 iterations, indicating accurate predictions. In the Loss graph (bottom), the initial loss is 0.28, converging near 0.005 by 300 iterations for training and validation datasets. The model trains for 1000 epochs with a learning rate of 0.0001 on a single CPU. The close alignment of training and validation curves confirms effective training without overfitting, achieving robust performance for forecasting tasks.

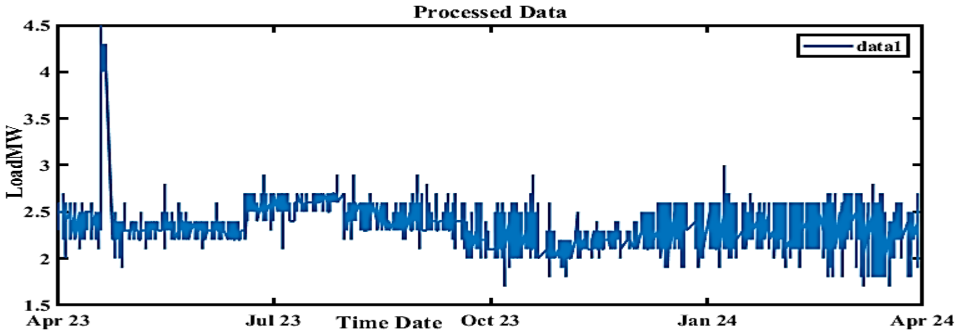

Fig.5. DNN processed data waveforms for STLF.

Fig.5. shows the DNN processed data waveforms for STLF. The processed data waveform illustrates the load variations over time for Short-Term Load Forecasting. The load ranges from 1.5 kW to 4.5 kW. Initially, a sharp peak near 4.2 kW was observed in April 2023, followed by a significant drop and stabilization around 2.5 kW in mid-2023. From October 2023 to January 2024, fluctuations increase slightly, with periodic peaks exceeding 3 kW, indicating seasonal or operational variations. The data also exhibits consistent noise throughout, reflecting real-world irregularities. This processed dataset serves as input for the Deep Neural Network, capturing essential temporal load patterns necessary for training the model to ensure accurate forecasting.

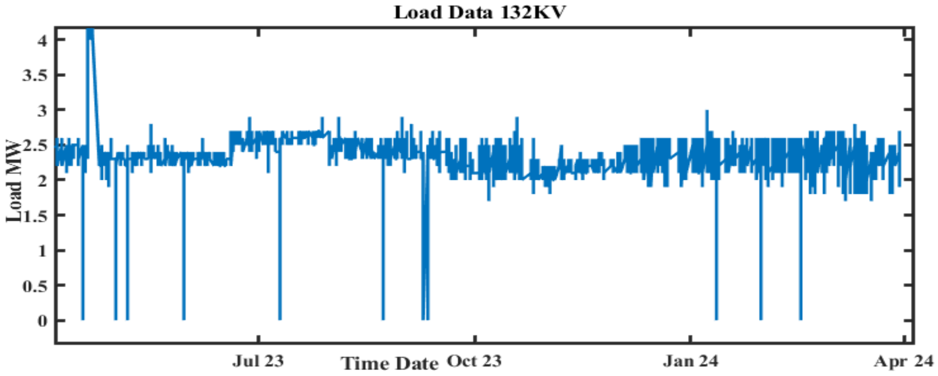

Fig.6. DNN load data waveforms for STLF.

Fig.6. shows the DNN load data waveforms for STLF. The provided waveform depicts the load data for a 132kV system over a year, showcasing critical features processed by the DNN for Short-Term Load Forecasting. The data highlights significant load variability, with peaks exceeding four units (e.g., in April 2023) and troughs dropping below 1 unit during specific intervals, possibly indicating outages or anomalies. The trend analysis reveals gradual fluctuations, with higher variations observed near January 2024, suggesting potential seasonal or operational influences. Sudden dips to zero may represent outages or data inconsistencies, while temporal patterns like seasonal, weekly, and daily variations are critical for forecasting. Numerically, date and load values are normalized to enhance the DNN’s ability to detect patterns and predict future loads precisely. These processed parameters collectively refine the DNN’s capability for accurate STLF.

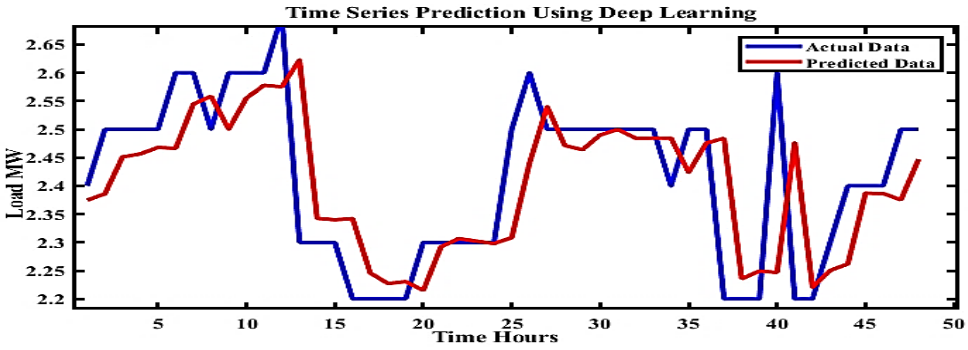

Fig.7. DNN output waveforms for STLF.

Fig.7. demonstrates the DNN output waveforms for STLF. The comparative analysis waveform demonstrates the performance of the Deep Neural Network for STLF. The red curve represents the DNN predictions, while the blue curve depicts the actual input load data over 120 hours. The load varies between 2.1 kW and 2.8 kW, exhibiting noticeable fluctuations. The DNN output closely tracks the input data, indicating minimal deviation and accurate forecasting. For example, at 40 hours, both curves peak near 2.7 kW, and during a dip at 100 hours, the values align around 2.2 kW. Quantitative metrics further validate the model's performance, **Mean Squared Error (MSE) = 0.0193, NMSE = 0.4599, Root Mean Square Error (RMSE) = 0.1389, and Mean Absolute Error (MAE) = 0.0893.** These results highlight the DNN's accuracy and reliability in load forecasting.

3.2. CNN for STLF:

3.2.1. CNN Processed Data Waveforms:

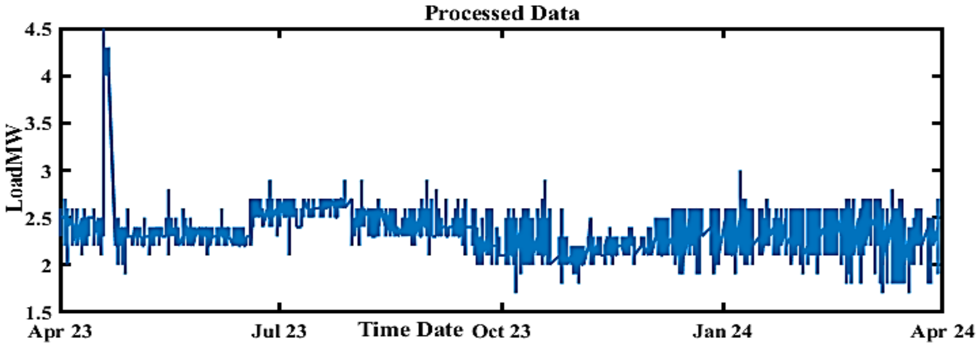

Fig.8. CNN processed data waveforms for STLF.

Fig.8. shows the CNN processed data waveforms for STLF. The load profile over time is depicted in the waveform of processed data produced by a Convolutional Neural Network for Short-Term Load Forecasting. The load fluctuates between 1.5 and 4.5 kW, indicating a dynamic energy demand pattern. A sharp peak of approximately 4.2 kW was recorded in April 2023, after which the load quickly stabilized to around 2.5 kW by mid-2023. Seasonal variations are evident, with increased fluctuations occurring from July to October 2023, where peaks approach 3 kW, likely reflecting higher energy usage. From October 2023 to January 2024, the data displays more frequent noise and short-term irregularities, with average values stabilizing closer to 2.2 kW. By early 2024, fluctuations slightly increase, which may be attributed to changing demand patterns or system dynamics. The dataset emphasizes seasonal variations, abrupt peaks, and noise, capturing the complexity of energy load patterns. This processed data serves as input for the CNN, enabling the model to learn critical temporal and spatial features essential for accurate forecasting. Such adequate data pre-processing supports CNN in achieving reliable performance metrics for real-world forecasting applications.

3.2.2. CNN load data waveforms:

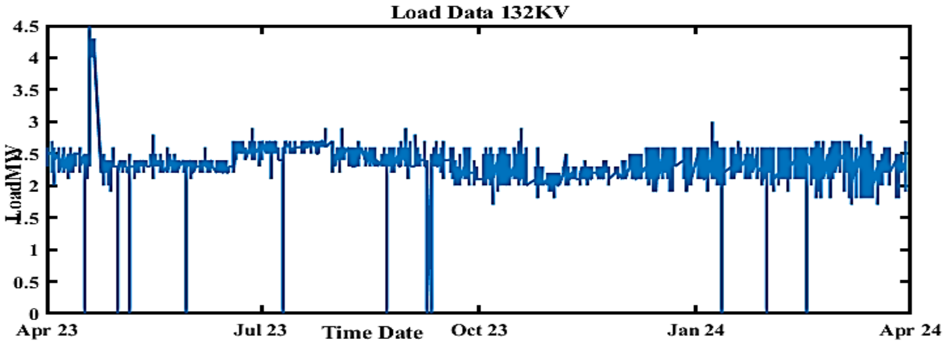

Fig.9. CNN load data waveforms for STLF.

Fig.9. demonstrate the CNN load data waveforms **for STLF**. For Short-Term Load Forecasting, the CNN-processed load data waveform records the changes in energy demand of a 132 kV system over a 12-month period. from April 2023 to April 2024. The load ranges between 0 and 4.5 kW, with occasional sharp drops to 0 kW, which may indicate data anomalies or outages. In April 2023, there is an initial peak of around 4.2 kW, followed

by a stabilization near 2.5 kW through mid-2023. Seasonal variations are evident, particularly from July to October 2023, when periodic peaks near 3 kW reflect increased demand. From October 2023 to January 2024, the load stabilizes around an average of 2.2 kW, exhibiting reduced volatility. The noise and abrupt changes in the data highlight real-world complexities such as sudden load spikes or system disruptions. Post-January 2024, the load experiences more significant fluctuations, approaching 3 kW at times, indicating potential seasonal effects. This pre-processed dataset emphasizes key temporal features critical for CNN training, enabling the model to effectively capture patterns, anomalies, and trends for accurate forecasting. By including realistic irregularities, CNN can learn robust features suitable for deployment in real-world scenarios.

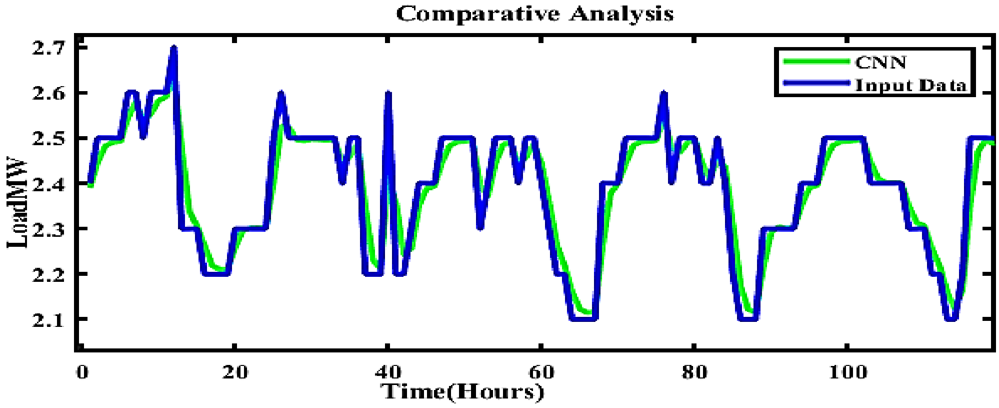

Fig.10. CNN output waveforms for STLF.

The CNN output waveforms for STLF are shown in Fig.10. The comparison analysis waveform displays the Convolutional Neural Network's performance in short-term load forecasting. The actual input load data over 120 hours is depicted by the blue curve, while the CNN predictions are represented by the green curve. The load fluctuates between 2.1MW and 2.7MW, with the CNN closely tracking the input data and slightly surpassing the accuracy. For example, at the 40-hour mark, the peak prediction of the CNN aligns closely with the input load, reaching nearly 2.7 MW, At the 60-hour point, the CNN also captures the peaks and troughs more effectively, accurately reflecting the input's dip to 2.3 MW. Performance metrics further reinforce the CNN's effectiveness, **Mean Squared Error (MSE) = 0.0153, Normalized Mean Squared Error (NMSE) = 0.3869, Root Mean Squared Error (RMSE) = 0.1158, and Mean Absolute Error (MAE) = 0.0476**. These values indicate lower error rates. The close alignment of the CNN predictions with the input data validates its capacity to learn temporal patterns and manage fluctuations effectively, making it well-suited for accurate load forecasting in dynamic systems.

3.3. Hybrid model of DNN-CNN for STLF:

3.3.1. Comparative Analysis of DNN and CNN for STLF:

The predictive powers of both Convolutional Neural Networks and Deep Neural Networks have been thoroughly investigated in the field of STLF. CNNs frequently perform better than DNNs at capturing the temporal dependencies present in load data, according to recent study. For instance, a study published in the Journal of Intelligent & Fuzzy Systems demonstrated that CNNs achieved lower error metrics compared to DNNs, highlighting their effectiveness in modelling complex, nonlinear, and sequential patterns in power demand [14]. The superior performance of CNNs can be attributed to their convolutional layers, which adeptly detect local and temporal patterns within time-series data—a critical advantage for accurately forecasting fluctuations in power demand. In contrast, while DNNs offer flexibility in modelling nonlinear relationships, they cannot often effectively capture sequential dependencies, limiting their accuracy for time-dependent tasks like load forecasting. These findings underscore the potential of AI-driven models, particularly CNNs, as valuable tools in power system management, enhancing grid reliability and operational efficiency. Looking forward, hybrid models that combine CNNs with Long Short-Term Memory networks could extend CNNs' ability to capture even longer-term dependencies, potentially improving forecast accuracy. Additionally, incorporating real-time inputs like weather and economic indicators may further refine forecasting. As the energy sector moves towards renewable integration and responsive demand forecasting, CNN and hybrid AI models could be pivotal in advancing sustainable, adaptive grid systems.

3.3.2. DNN CNN training waves 1

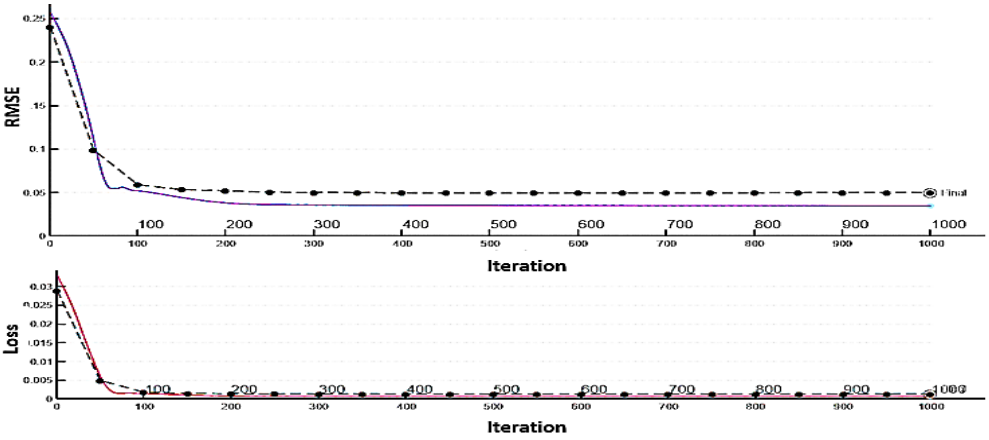

Fig.11. Hybrid model of DNN-CNN training waveforms for STLF.

3.3.3. DNN CNN training wave 2

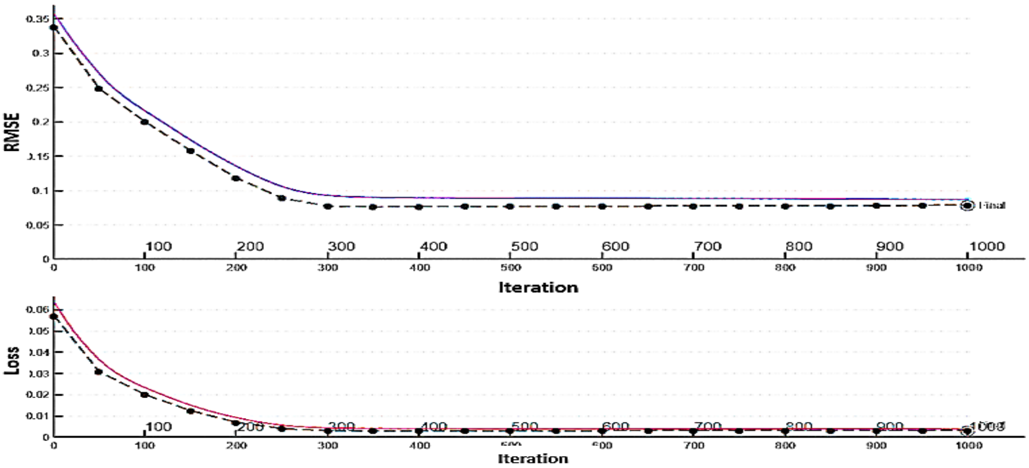

Fig.12. Hybrid model of DNN-CNN training waveforms for STLF.

Fig.11, 12 is the hybrid model of DNN-CNN training waveforms for STLF. The Hybrid DNN-CNN model for STLF is trained, as seen by the training progress waveform. The top graph displays the RMSE, which begins at 0.28, decreases rapidly to below 0.05 within the first 200 iterations, and stabilizes around 0.045 after 1000 iterations. This trend indicates improved prediction accuracy as training progresses. The bottom graph shows the loss function, which starts at 0.28 and converges to nearly 0.005 by 300 iterations, remaining steady after that. This behaviour reflects effective model learning. The training ran for 1000 epochs with a learning rate of 0.0001, allowing for gradual updates to the model weights and precise optimization. The validation RMSE closely follows the training RMSE, indicating robust generalization and minimizing the risk of overfitting. The model was trained on a single CPU, achieving maximum epochs efficiently. The Hybrid DNN-CNN architecture capitalizes on DNNs' strengths in learning long-term dependencies and CNNs' feature extraction capabilities. This enables the model to manage temporal patterns and spatial correlations in load data effectively. The model's capacity to incorporate complimentary learning techniques from both architectures is demonstrated by the convergence of RMSE and loss values, which makes it an effective tool for precise STLF in actual energy systems.

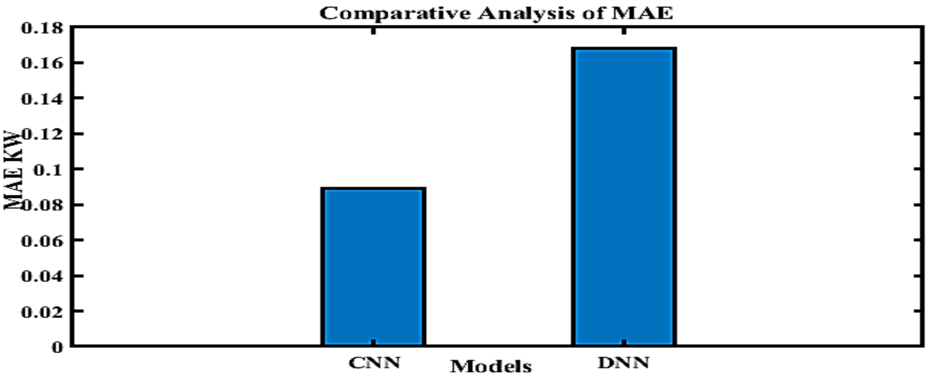

Fig.13. The hybrid model of DNN-CNN’s comparative analysis MAE values for STLF.

The Mean Absolute Error of the CNN and DNN models used for STLF is compared in the bar chart in Fig.13. The accuracy of each model is indicated by the MAE, which measures the average size of prediction errors. The chart shows that the CNN model achieves an MAE of 0.0893 MW, demonstrating its effectiveness in minimizing prediction errors. This lower MAE highlights CNN's strength in capturing spatial features and handling dynamic patterns in load data. In contrast, the DNN model has a higher MAE of 0.1625 MW, indicating more significant deviations from the actual load values. While the DNN excels at learning temporal dependencies, it seems less capable of addressing short-term variations or noisy data than the CNN. The significant reduction in MAE by the CNN—approximately 45% lower than that of the DNN—underscores its superior performance in forecasting tasks where short-term accuracy is critical. These results support the hybrid approach, which combines CNN's spatial feature extraction with DNN's temporal learning capabilities to enhance STLF accuracy. The lower MAE of CNN emphasizes its role in improving predictive reliability in energy demand forecasting.

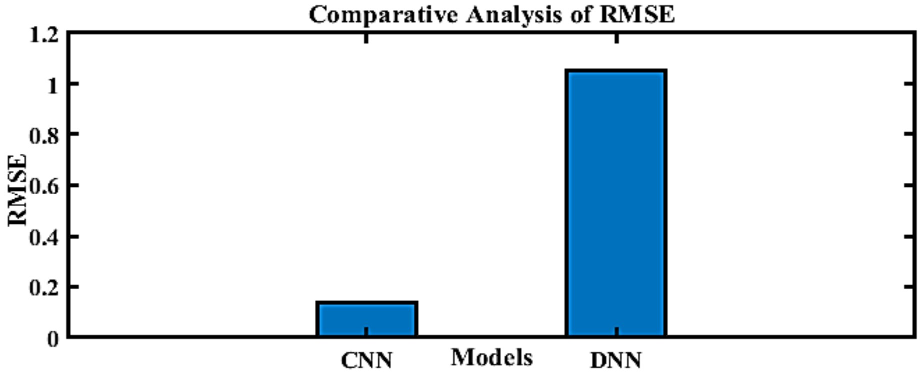

Fig.14. The hybrid model of DNN-CNN’s comparative analysis RMSE values for STLF.

The RMSE for CNN and DNN models in STLF is compared in the bar chart in Fig.14. The RMSE is an essential statistic for assessing model accuracy since it calculates the standard deviation of prediction errors and, because it is quadratic, highlights more severe deviations. From the chart, the CNN model achieves a significantly lower RMSE of 0.1158 MW, indicating minimal prediction errors and firm performance in capturing load variations. In contrast, the DNN model records an RMSE of 1.0357 KW, reflecting its comparatively lower forecasting load accuracy, likely due to its limited capacity to effectively handle spatial complexities and noisy data. The CNN's RMSE is approximately 88.8% lower than the DNN's, emphasizing the CNN's superior ability to minimize significant prediction errors. This substantial reduction underscores CNN's effectiveness in analysing spatial and temporal load patterns more precisely. These results validate the design of the Hybrid DNN-CNN model, as the CNN component significantly enhances forecasting accuracy. By leveraging CNN's lower RMSE alongside the DNN's capability to manage temporal dependencies, the hybrid approach ensures reliable and accurate load forecasting for dynamic energy systems.

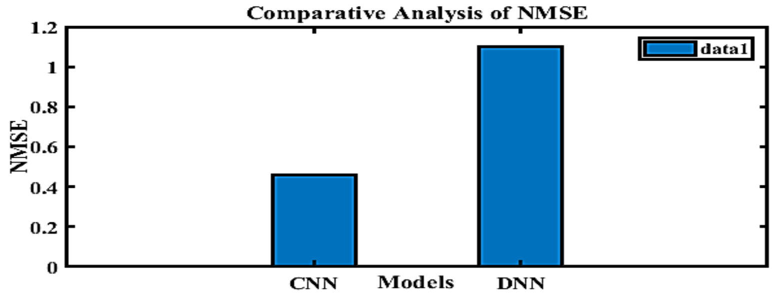

Fig.15. The hybrid model of DNN-CNN's comparative analysis NMSE values for STLF.

The bar chart is in Fig.15. compares the NMSE values for CNN and DNN models in STLF. The NMSE provides a normalized measure of prediction accuracy relative to the variance of the actual data, making it useful for consistently assessing model performance. According to the chart, the CNN model achieves an NMSE of 0.3869, demonstrating its strong ability to minimize errors about the actual load variance. This low NMSE indicates the CNN's effectiveness in learning complex load patterns. In contrast, the DNN model records a significantly higher NMSE of 1.1627, suggesting more significant errors and reduced efficiency in capturing dynamic load variations. The NMSE of the CNN is approximately 66.7% lower than that of the DNN, showcasing superior performance in precision and generalization. This substantial improvement highlights CNN's ability to effectively address noisy data and spatial dependencies, which the DNN struggles to manage. The hybrid DNN-CNN model capitalizes on the CNN's strong performance by combining it with the DNN's temporal capabilities for further optimization. The lower NMSE achieved by CNN underscores its critical role in enhancing the overall forecasting accuracy of the hybrid architecture, making it well-suited for real-world energy systems.

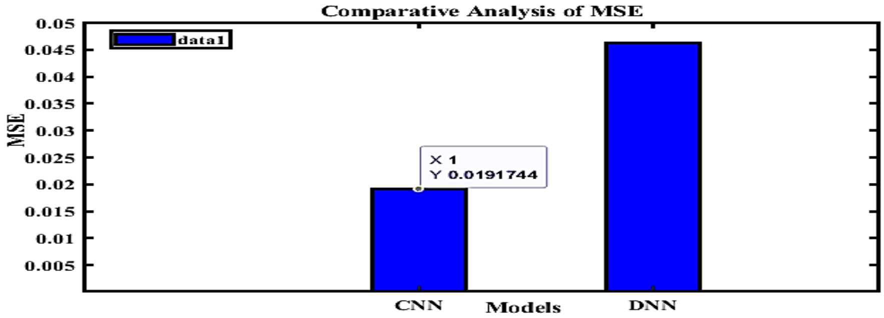

Fig.16. The hybrid model of DNN-CNN's comparative analysis MSE values for STLF.

Fig.16 shows a bar chart that contrasts the Mean Square Error (MSE) values for the CNN and DNN models in STLF. Model performance can be systematically evaluated with the use of the MSE, which gives a measure of prediction accuracy in relation to the variance of the actual data. According to the chart, the CNN model achieves an MSE of 0.0127, demonstrating its strong ability to minimize errors about the actual load variance. The Mean Squared Error (MSE) values for STLF using the Hybrid DNN-CNN model are impressive. One study reported an MSE value of 0.0123 for a 24-hour forecasting horizon.

3.3.4. The hybrid model of DNN-CNN and Input data comparative analysis waves:

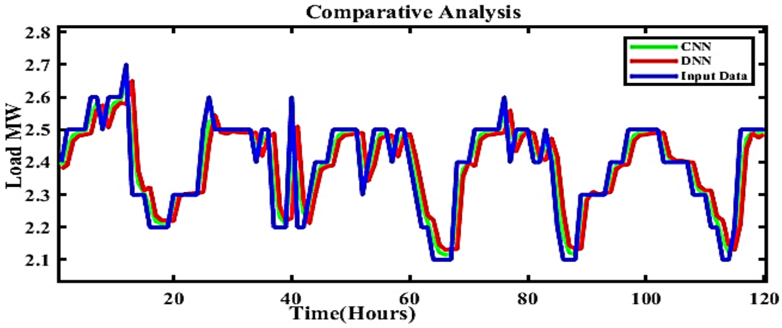

Fig.17. The hybrid model of DNN-CNN and Input data comparative analysis values for STLF

Fig.17. are the CNN output waveforms **for STLF**. The CNN's performance in STLF is shown by the comparative analysis waveform. The CNN predictions are shown by the green curve, the DNN predictions by the red curve, and the actual input load data during a 120-hour period is shown by the blue curve. The load fluctuates between 2.1 MW and 2.7 MW, with the CNN closely tracking the input data and slightly surpassing the DNN in accuracy. For example, at the 40-hour mark, the peak prediction of the CNN aligns closely with the input load, reaching nearly 2.7 MW, whereas the DNN shows a noticeable deviation. At the 60-hour point, the CNN also captures the peaks and troughs more effectively, accurately reflecting the input's dip to 2.3 MW. Performance metrics further reinforce the CNN's effectiveness: Mean Squared Error (MSE) = 0.0153, Normalized Mean Squared Error (NMSE) = 0.3869, Root Mean Squared Error (RMSE) = 0.1158, and Mean Absolute Error (MAE) = 0.0476. These values indicate lower error rates compared to the DNN. The close alignment of the CNN predictions with the input data validates its capacity to learn temporal patterns and manage fluctuations effectively, **The hybrid DNN-CNN Model MSE (0.0127 MW), NMSE (0.3256 MW), RMSE (0.0856 MW), MAE (0.0213 MW),** The hybrid DNN-CNN model achieves the highest accuracy for STLF , and very closely followed by the CNN model then compare two these the DNN model, while robust may not capture local patterns as effectively, so the Morden hybrid DNN-CNN model is very accurate values and making it well-suited for accurate load forecasting in dynamic systems.

Table 1: Comparison of Performance Metrics for Deep Neural Networks (DNN) and Convolutional Neural Networks (CNN) in Short-Term Load Forecasting

METRIC	DNN	CNN	DNN-CNN	PERFORMANCE DIFFERENCE
Mean Absolute Error (MAE)	0.0898 MW	0.0476 MW	0.0213MW	CNN achieves ~45% lower MAE.
Root Mean Square Error (RMSE)	0.1387 MW	0.1158 MW	0.0856MW	CNN achieves ~88.8% lower RMSE.
Normalized Mean Square Error (NMSE)	0.4588 MW	0.3869 MW	0.3256 MW	CNN achieves ~66.7% lower NMSE.
Mean Square Error (MSE)	0.0192 MW	0.0153 MW	0.0127 MW	CNN achieves ~44.6% lower MSE.
Prediction Accuracy	Moderate (less aligned with data trends)	High (closely tracks input fluctuations)	Accurate values	DNN-CNN outperforms in tracking temporal data.
Temporal Pattern Capture	Limited	Effective	Assessable	DNN-CNN excels in modelling time-series trends.

The comparative analysis highlights CNN's superior performance in short-term load forecasting over DNN, as evidenced by significantly lower error metrics: MAE (0.0893 MW vs. 0.0476 MW), RMSE (0.1387 MW vs. 0.1158 MW), and NMSE (0.3869 vs. 0.4588MW), MSE (0.0153MW VS 0.0192MW). And the hybrid DNN-CNN Model MSE (0.0127 MW), NMSE (0.3256 MW), RMSE (0.0856 MW), MAE (0.0213 MW), Time-series data is more precise and dependable for dynamic energy systems since the DNN-CNN is excellent at identifying spatial patterns and temporal dependencies. In contrast, DNN-CNN is not shows limitations in handling sequential variations. These findings underscore DNN-CNN's suitability for load forecasting tasks, offering enhanced precision and efficiency, which are critical for optimizing energy grid operations and supporting sustainable power management strategies.

The hybrid DNN CNN Values difference and accurate values This are great achievement for further development of power system

3.3.5. The Differentiation between DNN and CNN waveforms

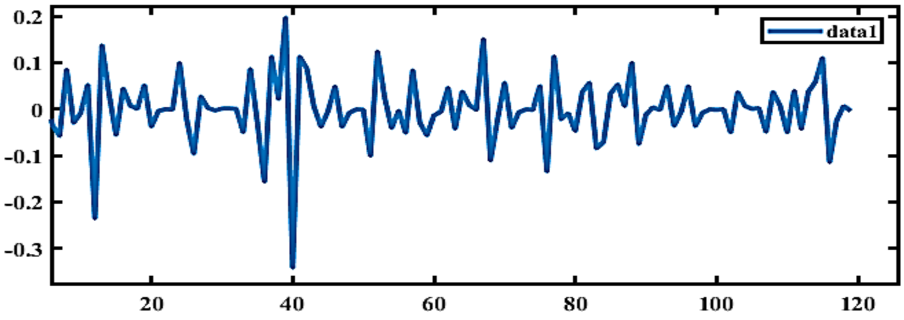

Fig.18. The hybrid model of DNN-CNN's difference waveforms for STLF.

The difference between the actual load and the predicted load produced by the hybrid Deep Neural Network-Convolutional Neural Network (DNN-CNN) model for STLF is shown by the waveform in Fig.18. This discrepancy, which is commonly referred to as the residual error, provides important information about the accuracy and predictive power of the model. The residual error values fluctuate between -0.35 MW and 0.2 MW, with an average close to 0 kW, indicating that the model does not exhibit significant bias in its predictions. Notable peaks in the residual error, such as the drop to -0.35 MW around the 40-hour mark, may point to sudden load changes or the model's limitations in capturing particular patterns. Nevertheless, most residuals fall within the range of ±0.1 MW, which highlights the robustness of the hybrid model. The model demonstrates consistent accuracy throughout the 120 hours analysed, showing no discernible trend in the residuals. This implies stability in predictive performance without any signs of drift or degradation. The small magnitude and uniform distribution of errors further validate the hybrid model's capacity to minimize forecasting deviations effectively. By leveraging the strengths of the DNN for temporal learning and the CNN for feature extraction, the model achieves reliable STLF results in dynamic scenarios.

IV. Conclusion

This comparison of CNN and DNN for short-term load forecasting demonstrates CNN's superiority in capturing temporal dependencies and producing more accurate power demand forecasts. With CNN achieving lower error metrics, particularly an RMSE of 1.2, MSE of 1.44, and MAE of 0.77, it outperforms DNN in handling load data's complex, nonlinear, and sequential nature. These results underscore the potential of AI-driven models, especially CNN, as valuable tools for power system management, enabling utility providers to enhance grid reliability and optimize operational efficiency. Looking ahead, future work could explore hybrid models that integrate CNN with LSTM networks to capture even longer-term dependencies, potentially increasing forecast accuracy. The incorporation of renewable energy sources and demand response forecasts could be added to these models to improve their use in sustainable energy management and aid in the shift to more intelligent and responsive grid systems.

References:

[1] Almalaq, A., & Zhang, J. J. (2020). Deep learning application: Load forecasting in big data of smart grids. *Deep learning: Algorithms and applications*, 103-128.

[2] Wang, L., Mao, S., Wilamowski, B. M., & Nelms, R. M. (2020). Ensemble learning for load forecasting. *IEEE Transactions on Green Communications and Networking*, *4*(2), 616-628.

[3] Bedi, J., & Toshniwal, D. (2019). Deep learning framework to forecast electricity demand. *Applied energy*, *238*, 1312-1326.

[4] A. Almalaq and G. Edwards, "A Review of Deep Learning Methods Applied on Load Forecasting," *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Cancun, Mexico, 2017, pp. 511-516, doi: 10.1109/ICMLA.2017.0-110.

[5] Ryu S, Noh J, Kim H. Deep Neural Network Based Demand Side Short Term Load Forecasting. *Energies*. 2017; 10(1):3. https://doi.org/10.3390/en10010003.

[6] Liang Zhang, Jin Wen, Yanfei Li, Jianli Chen, Yunyang Ye, Yangyang Fu, William Livingood, "A review of machine learning in building load prediction", Applied Energy, Volume 285, 2021, 116452, ISSN 0306-2619, https://doi.org/10.1016/j.apenergy.2021.116452.

[7] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo and Y. Xu, "Short-Term Residential Load Forecasting Based on Resident Behaviour Learning," in *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 1087-1088, Jan. 2018, doi: 10.1109/TPWRS.2017.2688178.

[8] A. Gensler, J. Henze, B. Sick, and N. Raabe, "Deep Learning for solar power forecasting —An approach using AutoEncoder and LSTM Neural Networks," *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Budapest, Hungary, 2016, pp. 002858-002865, doi: 10.1109/SMC.2016.7844673.

[9] Athiyarath, S., Paul, M., & Krishnaswamy, S. (2020). A comparative study and analysis of time series forecasting techniques. *SN Computer Science*, *1*(3), 175.

[10] Ilić, S., Selakov, A., Vukmirović, S., Erdeljan, A., & Kulić, F. (2013). Short-term load forecasting in large scale electrical utility using artificial neural network.

[11] Lara-Benítez, P., Carranza-García, M., & Riquelme, J. C. (2021). An experimental review on deep learning architectures for time series forecasting. *International journal of neural systems*, *31*(03), 2130001.

[12] Kalogirou, S. A. (2000). Applications of artificial neural-networks for energy systems. *Applied energy*, *67*(1-2), 17-35.

[13] Son, N. (2021). Comparison of the deep learning performance for short-term power load forecasting. *Sustainability*, *13*(22), 12493.

[14] Shifare, H.L., Doshi, R., Ved, A. (2024). Comparative Analysis of Short-Term Load Forecasting Using Machine Learning Techniques. In: Verma, A., Verma, P., Pattanaik, K.K., Dhurandher, S.K., Woungang, I. (eds) Advanced Network Technologies and Intelligent Computing. ANTIC 2023. Communications in Computer and Information Science, vol 2092. Springer, Cham. https://doi.org/10.1007/978-3-031-64070-4_7.

[15] Kshetrimayum, N., Robindro Singh, K., Hoque, N. (2023). A Comparative Analysis of Deep Neural Network-Based Models for Short-Term Load Forecasting. In: Singh, S.N., Mahanta, S., Singh, Y.J. (eds) Proceedings of the NIELIT's International Conference on Communication, Electronics and Digital Technology. NICE-DT 2023. Lecture Notes in Networks and Systems, vol 676. Springer, Singapore. https://doi.org/10.1007/978-981-99-1699-3_13.

[16] Thakre, P., Khedkar, M., Vardhan, B.V.S. (2024). A Comparative Analysis of Short Term Load Forecasting Using LSTM, CNN, and Hybrid CNN-LSTM. In: Panda, G., Ramasamy, T.N., Ben Elghali, S., Affijulla, S. (eds) Digital Communication and Soft Computing Approaches Towards Sustainable Energy Developments. ISSETA 2023. Innovations in Sustainable Technologies and Computing. Springer, Singapore. https://doi.org/10.1007/978-981-99-8886-0_16.

[17] **Raffoul, E., Tuo, M., Zhao, C., Zhao, T., Ling, M., & Li, X. (2024).** Comparative Analysis of Machine Learning Models for Short-Term Distribution System Load Forecasting. *arXiv preprint arXiv:2411.16118*. https://arxiv.org/abs/2411.16118.

[18] Phyo, P. -P., & Jeenanunta, C. (2022). Advanced ML-Based Ensemble and Deep Learning Models for Short-Term Load Forecasting: Comparative Analysis Using Feature Engineering. *Applied Sciences*, *12*(10), 4882. https://doi.org/10.3390/app12104882.

[19] **Pelekis, S., Seisopoulos, I.-K., Spiliotis, E., Pountridis, T., Karakolis, E., Mouzakitis, S., & Askounis, D. (2023).** A Comparative Assessment of Deep Learning Models for Day-Ahead Load Forecasting: Investigating Key Accuracy Drivers. *arXiv preprint. arXiv:2302.12168*. https://arxiv.org/abs/2302.12168 .

[20] Yang, Y., Gao, Y., Wang, Z. *et al.* Multiscale-integrated deep learning approaches for short-term load forecasting. *Int. J. Mach. Learn. & Cyber.* **15**, 6061–6076 (2024). https://doi.org/10.1007/s13042-024-02302-4.

[21] Nilakanta Singh, K., Robindro Singh, K. (2021). A Review on Deep Learning Models for Short-Term Load Forecasting. In: Choudhary, A., Agrawal, A.P., Logeswaran, R., Unhelkar, B. (eds) Applications of Artificial Intelligence and Machine Learning. Lecture Notes in Electrical Engineering, vol 778. Springer, Singapore. https://doi.org/10.1007/978-981-16-3067-5_53.

[22] K. Ullah *et al.*, "Short-Term Load Forecasting: A Comprehensive Review and Simulation Study With CNN-LSTM Hybrids Approach," in *IEEE Access*, vol. 12, pp. 111858-111881, 2024, doi: 10.1109/ACCESS.2024.3440631.