Power System Technology
ISSN:1000-3673

Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

Power Quality Monitoring and Forecasting Using LSTM and ANFIS Methods for Isolated Power System Networks

Vijayshree G[1*], Kumar P K [2], Sumathi S[3]

[1,3]*Department of Electrical and Electronics Engineering, RNS Institute of Technology, Bangalore.*

[2]*Department of Computer Science Engineering, Visvesvaraya Technological University, Mysuru.*

[*]*Correspondence author:* shreechetu@gmail.com

Abstract: - In the present environment, the concept of Power Quality (PQ) is of high interest for any power system researcher. The majority of loads at the consumer end are designed using power electronic components and also integration of renewable energy resources, which introduces PQ issues in terms of voltage and current. Hence, appropriate monitoring and forecasting methods of PQ parameters are essential for maintaining grid reliability and stability. In this paper, we introduce a data-driven method for forecasting PQ disturbances using LSTM networks and ANFIS models in a power system environment. A dataset generated from MATLAB/Simulink simulations of an isolated distribution system with nonlinear loads is used for model training and evaluation. The models are evaluated using key performance metrics, including Mean Absolute Error (MAE), Mean Squared Error (MSE), and Coefficient of Determination (R^2). Additionally, a comparative benchmarking analysis with conventional machine learning models, including Multilayer Perceptron (MLP), Random Forest (RF), and XGBoost (XGB), is presented. Experimental results demonstrate that LSTM achieves superior predictive accuracy ($R^2 > 0.998$), while ANFIS offers competitive performance and interpretability. These findings underscore the potential of AI-based techniques for proactive PQ monitoring and forecasting in power system environments.

Keywords*: Power Quality Monitoring, LSTM, ANFIS, Machine Learning, Deep Learning, Time-Series Forecasting, Performance Benchmarking.*

1. Introduction

PQ plays a crucial role in maintaining the operational efficiency, stability, and reliability of modern power systems. Due to the need for a reduction in carbon emissions, the present power system environment includes the integration of renewable energy resources for power generation. This design includes a large number of power electronics devices which contribute to current-related PQ issues such as harmonics, and based on the integration of renewable resources, the voltage variation occurs at the supply terminal, which contributes to voltage-related PQ issues such as voltage sag, swell, or harmonics. These PQ issues can result in

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

equipment failure, malfunction of switchgear equipment, an increase in power loss, and a reduction in system stability and reliability.

The traditional PQ monitoring approaches implement hardware compensations such as custom power devices to provide voltage and current-related compensations, which lack adaptability and predictive intelligence. In order to overcome this issue, intelligent approaches such as artificial intelligence (AI) and machine learning (ML) methods are designed using a programming platform for classifying, predicting, and generating appropriate control signals to provide suitable compensation.

The work explained in this paper mainly focuses on Long Short-Term Memory (LSTM) networks, a deep learning method for time-series data, alongside the Adaptive Neuro-Fuzzy Inference System (ANFIS), which merges fuzzy logic with neural networks for interpretability and precision. A comparative performance analysis is also conducted using conventional ML models Multilayer Perceptron (MLP), Random Forest (RF), and XGBoost (XGB) to evaluate their forecasting capabilities under various PQ disturbance conditions simulated in a MATLAB/Simulink-based power system network. The objective is to benchmark these intelligent methods to identify the most effective approach for predictive PQ monitoring in smart grid environments. Numerous studies have contributed to this domain, focusing on classification, prediction, and data visualization for power system stability.

One major strand of research involves deep learning methods for PQ disturbance detection. For instance, Machlev et al. [1] proposed an open-source PQ disturbance dataset along with a convolutional neural network (CNN) and bidirectional LSTM (BiLSTM) classifiers, offering a foundational reference for future benchmarking. Similarly, various works [2], [3], [4] have integrated LSTM with hybrid approaches such as particle swarm optimization and fuzzy logic for predicting disturbances and enhancing voltage stability under dynamic conditions.

The application of distributed computing frameworks has enabled scalable PQ analysis by handling large volumes of real-time energy data. Guo et al. [5] discussed how distributed computing tools like Apache Spark and PySpark can process high-volume sensor data for real-time system health assessment and renewable energy forecasting. These platforms have been successfully used for implementing classification and clustering tasks using models such as random forest, gradient boosting, and support vector machines.

The transition toward smart grid environments has been supported by the adoption of smart sensors and Internet of Things (IoT) frameworks, as noted in Zayas et al. [6]. These infrastructures facilitate real-time data acquisition and enable predictive maintenance through ML-driven anomaly detection. Additional emphasis on hardware-in-the-loop systems and synchro phasor-based PQ analytics [7] has further enhanced operational reliability in dynamic scenarios.

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

In terms of software tools, Python has become a preferred language for both academic and industrial applications. Gómez Tapias et al. [8] and Milano [9] introduced Python-based PQ

diagnostic tools like HCE-PQD and Pandapower, which support modular analysis and visualization of power disturbances, including harmonic distortion, voltage sag, swell, and flicker. Visualization strategies leveraging dashboards and analytics libraries such as Grafana [10], Tableau, Dash, and Power BI [11] have become common for interpreting large-scale energy datasets effectively.

Several studies [12], [13], [14] have investigated the value of interactive and data-driven visualizations for performance monitoring and fault localization. Meanwhile, efforts by Itagi [15] and Akhter et al. [16] emphasized the predictive modeling of renewable generation systems, using time-series models for solar photovoltaic (PV) power forecasting and dynamic load estimation.

The research carried out in [17], [18] is to develop ML models such as SVM, RNN, and Q-learning to obtain a solution for voltage profile improvement and to address the issues related to power loss minimization. Additionally, power system simulators and educational packages [9], [19] are increasingly used to train models and test forecasting frameworks under controlled and realistic scenarios.

Recent technological developments include the deployment of low-cost, real-time PQ analyzers using embedded systems and Python-based microcontrollers [20], as well as applications in lithium-ion battery diagnostics [21], [22], where ML is used for estimating state-of-charge and cycle life analysis [23], [24]. These studies highlight the growing convergence of power electronics, embedded systems, and intelligent data processing for sustainable energy management.

2. Methodology

This study adopts a simulation-driven approach to generate a comprehensive dataset representing various power quality (PQ) disturbances in a low-voltage distribution system. The methodology encompasses three primary stages: system modeling and data acquisition, data preprocessing, and prediction using intelligent models namely, Long Short-Term Memory (LSTM) networks and Adaptive Neuro-Fuzzy Inference System (ANFIS). Comparative evaluation is also performed using conventional machine learning (ML) algorithms.

(i) Power System Simulation and Data Acquisition

A three-phase isolated power system is modeled using MATLAB/Simulink to simulate PQ events under diverse operating conditions. The network includes a renewable energy-based generation system consisting of solar photovoltaic (PV) panels, a wind turbine, and a battery storage unit, supplying power to nonlinear loads through a transmission line. Nonlinear load

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

behavior introduces harmonic distortions in the current, while dynamic variations in the supply system create voltage sags, swells, and imbalances.

The simulation captures critical electrical parameters at both the supply and load ends. These include voltages, currents, real and reactive power, and power factor across three phases. The resulting dataset consists of 21,876 rows and 19 attributes, offering a rich basis for time-series forecasting. Data is exported in Excel format for further processing.

(ii) Data Preprocessing

Feature normalization was conducted via Min-Max scaling to align all variables within a 0–1 interval for model consistency. For time-series analysis, the dataset is restructured into sequences of 50 consecutive timesteps per sample. This sliding window method allows each model to predict the next time step based on historical trends.

The normalized and sequenced dataset is divided into training and testing subsets with an 80:20 ratio, ensuring sufficient data for model learning and generalization.

(iii) Predictive Modeling Framework

a) Long Short-Term Memory (LSTM): It is a powerful "recurrent neural network system designed to overcome the vanishing gradient problems that typically arise when learning long term dependencies" [25].

Two types of LSTM architectures are implemented:

- Multi-Target LSTM: Independent LSTM models are trained for each voltage phase (e.g., Vabc1, Vabc2, Vabc3), for predicting the next voltage value in terms of 50-step input windows.
- Multi-Output LSTM: A single LSTM model simultaneously predicts multiple output variables (Vs1, Vs2, Vs3), capturing inter-phase dependencies.

Both models use two stacked LSTM layers followed by a Dense output layer. Training is conducted over 20 epochs using the Adam optimizer with mean squared error (MSE) as the loss function. Predictions are inverse-transformed to the original scale before evaluation.

b) Adaptive Neuro-Fuzzy Inference System (ANFIS)

The ANFIS model is configured to map three-phase supply voltages (Vabc1, Vabc2, Vabc3) as inputs to predict the corresponding three-phase load voltages (Vs1, Vs2, Vs3). Each input variable is associated with three Gaussian membership functions, and the system uses a hybrid learning rule combining least-squares estimation and backpropagation for training. The model runs for 50 epochs and produces phase-wise predictions with interpretable fuzzy rules.

(iv) Benchmark Models

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

To contextualize the performance of LSTM and ANFIS, three standard ML models are implemented for comparative analysis:

- Multilayer Perceptron (MLP): A feed-forward neural network with ReLU activations and dense layers.
- Random Forest (RF): An ensemble of decision trees trained on bootstrap samples.
- Extreme Gradient Boosting (XGBoost): A sequential ensemble model using gradient boosting optimization.

All models are trained on the same normalized dataset and evaluated using identical metrics for fair benchmarking.

3. Evaluation Metrics

The performance of the designed model is measured using metrics like Mean absolute Error (MAE), Mean Squared Error (MSE) and coefficient of determination(R^2). These metrics are considered to be the standard evaluation metrics as per the literature review carried out. Let me explain each of the metrics along with it's corresponding equation used for calculation.

a) Mean Absolute Error (MAE): It is the average magnitude of prediction errors. The lower the value of MAE, the better the prediction model is considered. It is calculated as

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{i=n}(y_i - \hat{y_i}) \tag{1}$$

where y_i= actual value
$\hat{y_i}$ =Predicted value
n=Number of samples.

b) Mean Squared Error (MSE): It calculates larger errors via squared differences. The lower the value of MSE, the better the prediction model is considered. It is calculated as

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

where y_i= actual value
$\hat{y_i}$ =Predicted value
n = Number of samples.

c) Coefficient of Determination (R^2)**:** Indicates the proportion of variance explained by the model. If the value of R^2 =1, then the prediction is said to be a perfect prediction.

$$R^2 = 1 - \frac{\sum_{i=1}^{i=n} (y_i - \hat{y_i})^2}{\sum_{i=1}^{i=n} (y_i - \bar{y_i})^2} \tag{3}$$

where y_i= actual value
$\hat{y_i}$ =Predicted value
n = Number of samples.

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

Each metrics are calculated based on the above equation parameters and based on the obtained value, the output is inferred. For the model to be considered as perfect prediction model, the value of MAE, MSE should be lower and value of R^2 should be approximately equal to 1. These metrics are calculated for all the five different approaches. The MLP, RF and XGBoost models are considered as a benchmark reference models and the LSTM model and ANFIS model performance are compared with the benchmark models to analyze which model performs best.

4. Machine learning coding

This section presents the performance comparison of the developed LSTM and ANFIS models. In LSTM model, two different algorithms are derived (i) Multi-Target LSTM and (ii) Multi-Output LSTM. The MLP, RF, and XGBoost models are analyzed using key evaluation metrics such MAE, MSE, and Coefficient of Determination (R^2). In this study, the variable considered for prediction analysis is voltage parameter. Since no controllers are introduced in the system, the supply voltage is same as the load voltage. Hence prediction performed on supply voltage (Vabc) is same as prediction performed on the load voltage (Vs).

a) Multi-Target LSTM (MTLSTM) Model for Power Quality Prediction

In this type of algorithm, the prediction is done for multiple target variables separately. one LSTM model is trained to predict one target variable. In the algorithm defined below, the target variables considered are the three-phase supply voltages Vabc1, Vabc2, and Vabc3. The model is trained for one independent variable LSTM per target using a fixed window of 50 timesteps, which is further used to predict the next value of the target variable. Hence, the Dense layer with 1 neuron(output) is considered in the code. Each model computes Mean Absolute Error (MAE), Mean Squared Error (MSE), and the Coefficient of Determination (R^2) value.

Algorithm for MTLSTM:

Input:
    Dataset from Excel (sheet 'AllData')
    Target Variables T: Vabc1, Vabc2, Vabc3
    Hyperparameters: Time Steps = 50, Epochs = 20, Batch Size = 64
Output:
    Evaluation metrics: MAE, MSE, R^2 for each target
    Visualization: Actual vs Predicted plots and Metric Comparison Chart

  Start
  Load dataset from the Excel (sheet ‘AllData’)
  and remove the 'Time' column from Excel sheet
  Define target columns T

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

```
    Initialize lists: y_test_list, y_pred_list, metrics_list
    For each target t in T Do

Separate features X and target y, where y = column t
    Normalize X and y using Min-Max scaling
    Create sequences for LSTM:
          For i = 1 To (length(X) - Time Steps) Do
                X_seq[i] = X[i : i + Time Steps]

            Y_seq[i] = Y[i + Time Steps]
           End For
   Split X_seq and Y_seq into training (80%) and testing (20%) sets
   Initialize the LSTM model:
   Input Layer: shape = (Time Steps, Features)
   LSTM Layer with 64 units, return_sequences = TRUE
   LSTM Layer with 32 units
   Dense Layer with 1 neuron (output)
   Compile the model with Adam optimizer and MSE loss
  Train model for 20 epochs with batch size 64 and validation split = 0.2
  Predict Y_pred for X_test
  Inverse transform Y_pred and Y_test to the original scale
   Compute metrics:
              MAE = Mean Absolute Error
              MSE = Mean Squared Error
              R² = Coefficient of Determination
   Append predictions and metrics to the respective lists
    End For

Visualization:
         a) For each target t in T Do
               Plot Actual vs Predicted for the first 100 points
                  Display metrics on each subplot
                End For
             b) Generate a bar chart for MAE, MSE, and R² across all targets

      Print a summary table of metrics.
      End
```

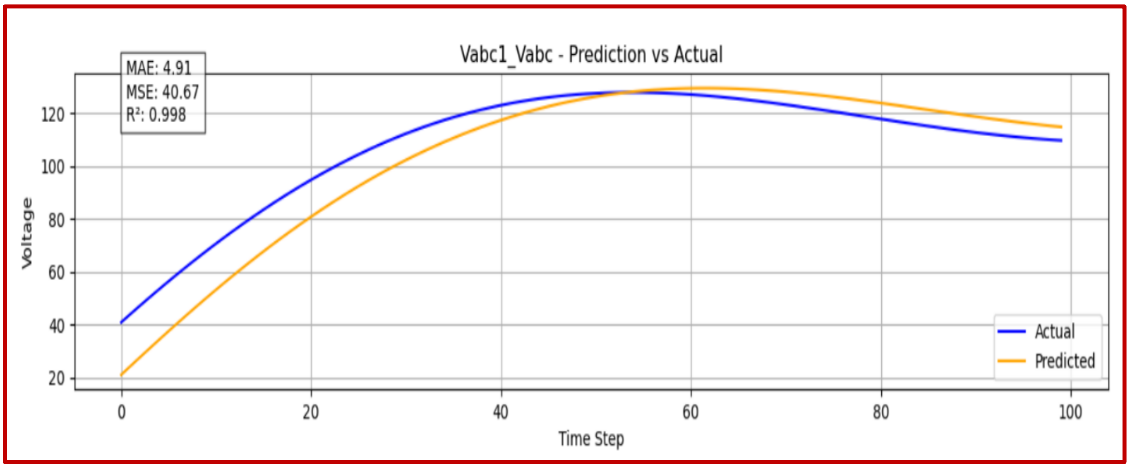
The pseudocode for the multi-target LSTM method derives the value of MAE, MSE, and R^2 value for the three-phase supply voltage as shown in the Table 1.

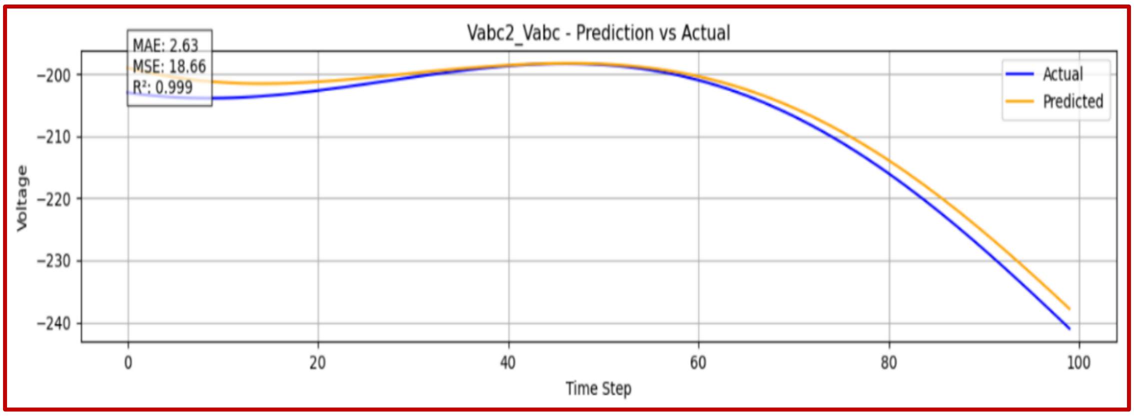
Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

Table 1: Evaluation metrics for MTLSTM model

Three Phase Voltage	MAE	MSE	R^2
Vabc1	4.906	40.672	0.998
Vabc2	2.627	18.664	0.999
Vabc3	3.427	29.179	0.999

The plot representing the actual value and the predicted value across timestamps is shown in the Figure 1. The plot indicates the output graph of actual voltage and predicted voltage graph independently plotted for each voltage. It is observed that the actual and predicted graph are almost in line to each other. In this method, since separate models are required to train each target variable, the computational expenses are high. The main disadvantage of this type of model is that it ignores the relationship among the target variables. In order to overcome this disadvantage, a multi-output LSTM is developed.





Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

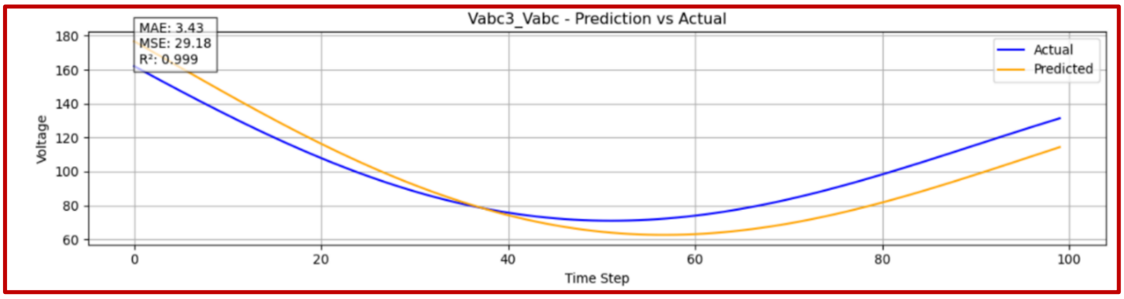


Figure 1: Actual and Predicted voltage plot for MTLSTM model for three phase voltage.

b) Multi-Output LSTM (MOLSTM) for Multi-Phase Voltage Prediction

In this type of algorithm, the prediction is performed on multiple target variables simultaneously. One LSTM model can predict multiple target variables. Hence, the computational expenses are reduced as compared to the multi-target LSTM model. The algorithm also builds interdependency between different target variables.

Algorithm for MOLSTM:

```
Input:
    Dataset D from Excel (sheet 'AllData')
    Target columns T: Vs1, Vs2, Vs3
    Features: All columns except T
    Hyperparameters: Time Steps = 50, Epochs = 20, Batch Size = 64
Output:
    Performance metrics: MAE, MSE, R² for each target
    Visualization of Actual vs Predicted voltages

Start
    Load dataset
    Remove the time column
    Define target columns T and feature columns X_cols = D \ T
    Split the dataset into:
        X = features, y = targets
        Normalize X and y using Min-Max scaling
    Create sequences:
        For i = 1 To (len(X) - Time Steps) Do
        Append X[i : i + Time Steps] to X_seq
        Append y[i + Time Steps] to y_seq
    End For

        Split X_seq, y_seq into:
```

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

Training set (80%) and Testing set (20%)

Define Multi-Output LSTM model:
    Input Layer: shape = (Time Steps, Features)
    LSTM Layer with 64 units (return_sequences = True)
    LSTM Layer with 32 units
    Dense Layer with N outputs (N = number of targets)
Compile the model with Adam optimizer and MSE loss
    Train the model on the training data:
    Epochs = 20, Batch Size = 64, Validation Split = 0.2
    Predict on testing data → y_pred_scaled
    Inverse transform y_pred_scaled and y_test to the original scale
    FOR each target t in T Do
        a) Compute metrics:
            MAE = Mean Absolute Error
            MSE = Mean Squared Error
            R^2 = Coefficient of Determination

        b) Plot Actual vs Predicted for the first 100 samples
            Display a summary table of metrics for all targets
            End

The pseudocode for the multi-output LSTM method derives the value of MAE, MSE, and R^2 value for the three-phase system voltage as shown in the Table 2.

Table 2: Evaluation metrics for MTLSTM model

Three Phase Voltage	MAE	MSE	R^2
Vs1	3.887393	51.830163	0.998056
Vs2	3.040819	25.772454	0.999043
Vs3	2.637553	28.164954	0.998954

The plot representing the actual value and the predicted value across timestamps is as shown in the Figure 2. It represents the plot of actual and predicted values for three three-phase voltages independently. It is observed that compared to MTLSTM, the value of MSE is higher in MOLSTM. Hence ANFIS model is designed to reduce the MSE value and derive a efficient predictive model.

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

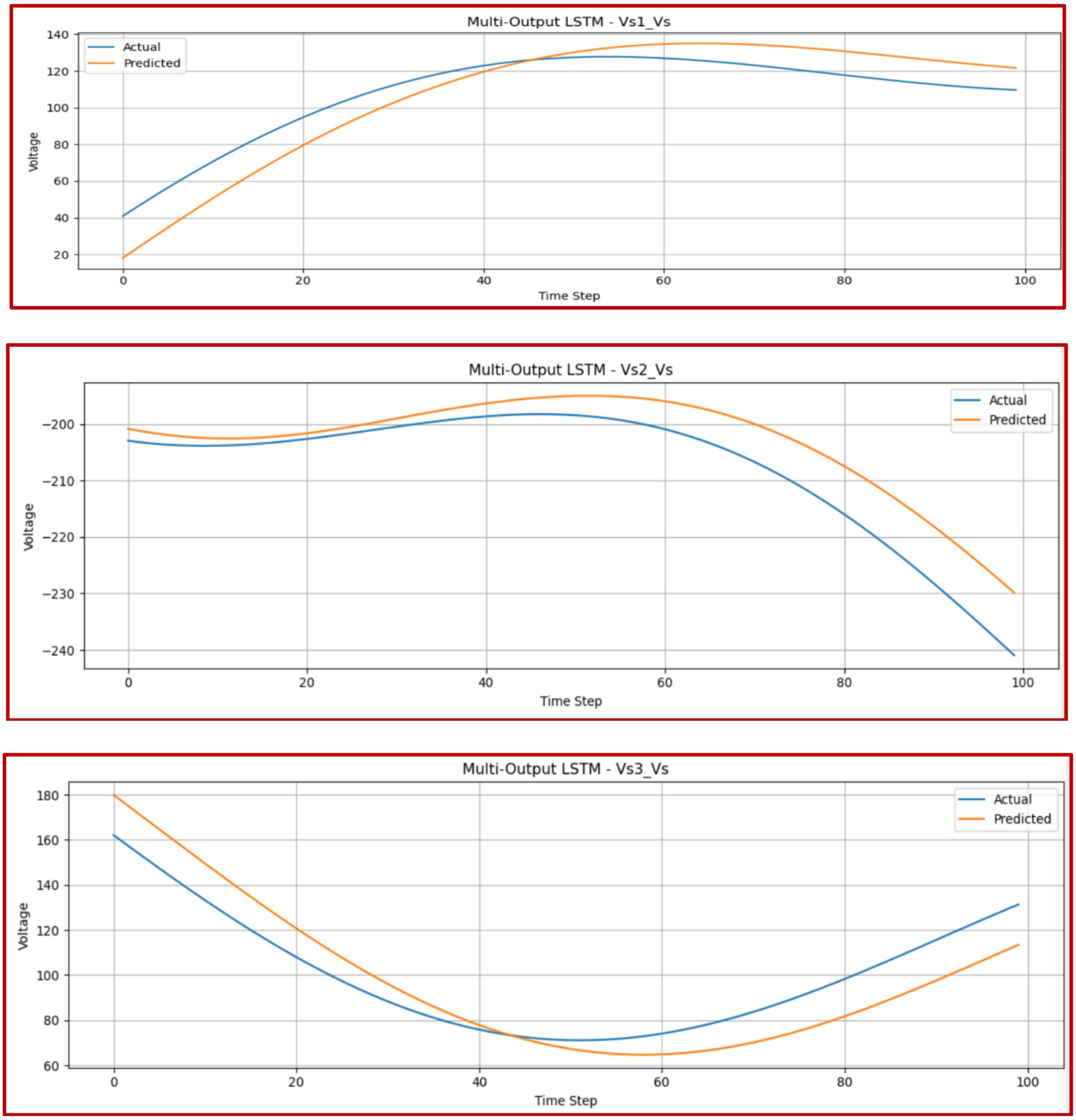


Figure 2: Actual and Predicted voltage plot for MOLSTM model for three phase voltage.

c) ANFIS: ANFIS-Based Multi-Target Voltage Prediction

This approach integrates fuzzy logic with neural network learning, providing both interpretability and adaptability. It uses fuzzy rules to model nonlinear relationships and an adaptive mechanism to fine-tune membership functions and consequent parameters during

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

training. This model is advantageous when rule-based decision-making is required alongside predictive capability.

```
Input:
    Dataset D (sheet 'AllData')
    Input feature set X_cols: Vabc1, Vabc2, Vabc3
    Target set T : Vs1, Vs2, Vs3
    num_mfs = 3 #per input variable#
    epochs = 50
Output:
    Performance metrics (MAE, MSE, R²) for each target signal
    Prediction plots (Actual vs Predicted)

Start
Load dataset D; remove the Time column.
Form input matrix X ← D[X_cols].
Form target matrix Y ← D[T].
Normalize X and Y using Min-Max scaling → X_norm, Y_norm.
Split X_norm, Y_norm into Training (80%) and Testing (20%) sets.
num_inputs ←: number of columns in X_cols.
Generate fuzzy rule base:
rules ← CartesianProduct({0.num_mfs-1} repeated num_inputs times).
       num_rules ← |rules|.
Initialize premise (membership) parameters:
       For each input i and MF j: centers c[i,j] ~ U(0,1); spreads σ[i,j] ~ U(0,1).
Initialize consequent parameters θ[num_rules][num_inputs + 1].  // bias + linear terms

For each target t in T Do
y_train ← training column of Y_norm for t
y_test  ← testing  column of Y_norm for t

 #Training#
 For epoch = 1 To epochs Do
     (ŷ_train, NormFire) ← ForwardPass(X_train, c, σ, θ, rules)
     err ← y_train − ŷ_train
     loss ← MeanSquare(err)

  #Update consequents by batch Least Squares#
  Build design matrix A:
            For each sample s:
                For each rule r:
                    row_extend NormFire[s,r] * [1, X_train[s,1], ..., X_train[s,num_inputs]]
  Solve θ ← Reshape( PseudoInverse(A) * y_train ).
  IF epoch mod 10 == 0: print loss.
```

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

```
  End For

#Testing #
(ŷ_test, _) ← ForwardPass(X_test, c, σ, θ, rules)
Rescale ŷ_test and y_test back to original units using Y scaler.
Compute metrics:
        MAE ← MeanAbsoluteError(y_test_actual, y_pred_actual)
        MSE ← MeanSquareError(y_test_actual, y_pred_actual)
        R²  ← CoefficientOfDetermination(y_test_actual, y_pred_actual)
Append (t, MAE, MSE, R²) to Results.

Plot the first 100 samples: Actual vs Predicted for target t.
End For

Display Results table (all targets).
End
```

The pseudocode for the multi-output LSTM method derives the value of MAE, MSE, and R^2 value for the three-phase supply voltage as shown in the Table 3.

Table 3: Evaluation metrics for ANFIS model

Three phase voltage	MAE	MSE	R^2
Vs1	0.025215	0.001324	1
Vs2	0.022459	0.001150	1
Vs3	0.022281	0.001094	1

The plot representing the actual value and predicted value across timestamps is shown in the figure. It represents the plot of actual and predicted values for three three-phase voltages independently.

Power System Technology
ISSN:1000-3673

Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

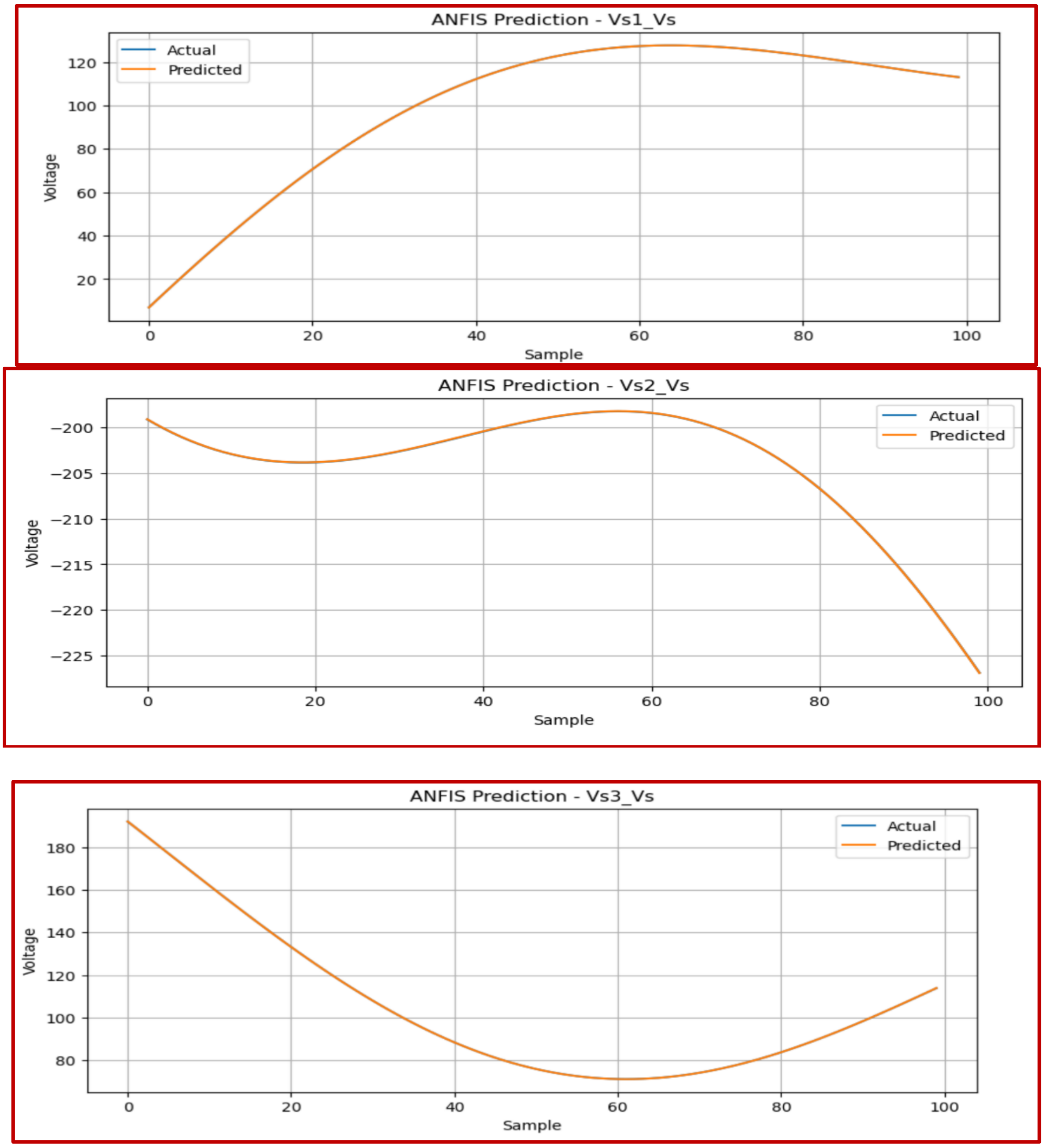


Figure 3: Actual and Predicted voltage plot for ANFIS model

Comparative Performance Analysis of MLP, Random Forest, and XGBoost Models

Algorithm: Model Comparison (MLP, Random Forest, XGBoost)

Multilayer Perceptron (MLP): MLP is a feed-forward neural network that consists of multiple fully connected layers, where each neuron applies a nonlinear activation function such as

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

ReLU. It excels in capturing complex nonlinear relationships between input and output variables, but does not handle temporal dependencies since it lacks recurrent connections. Training is performed using backpropagation with optimizers like Adam.

Ensemble Models (Random Forest and XGBoost): These models combine predictions from multiple decision trees to improve accuracy and robustness. Random Forest uses bagging, where multiple trees are trained on random subsets of data, and predictions are averaged for regression tasks. XGBoost employs gradient boosting, sequentially building trees that correct the errors of previous trees. These methods are efficient for structured data and offer strong predictive performance for non-sequential problems, but cannot model long-term dependencies inherent in time-series data.

```
Input:
   Dataset D with columns: Time, Vabc1, Vabc2, Vabc3, Vs1
Output: Vs1
   Performance metrics: MAE, MSE, R² for each model
   Summary table of comparison

Start
Load dataset D from the Excel sheet 'AllData'.
Remove column 'Time' from D.
Define:
      X ← {Vabc1, Vabc2, Vabc3}
      y ← {Vs1}
Normalize features and target using MinMaxScaler:
      X_scaled ← scale(X)
      y_scaled ← scale(y)
Split X_scaled and y_scaled into:
      Training set (80%) and Testing set (20%)

Define evaluation function Evaluate_Model(name, y_true, y_pred_scaled):
      Inverse transform y_true and y_pred_scaled to the original scale
      Compute MAE, MSE, and R² Score
      Print metrics
      Return (MAE, MSE, R²)

Phase 1: Train the MLP Neural Network
Define MLP with layers:
      Input → Dense(32, ReLU) → Dense(32, ReLU) → Dense(16, ReLU) → Dense(1,
Linear)
Compile MLP with 'adam' optimizer and 'mse' loss.
Train MLP on (X_train, y_train) for 50 epochs with batch size = 64.
```

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

```
Predict y_pred_mlp ← MLP(X_test)
Evaluate_MLP ← Evaluate_Model("MLP", y_test, y_pred_mlp)

Phase 2: Train Random Forest Regressor
Define RandomForestRegressor with n_estimators = 100
Train RF on (X_train, y_train)
Predict y_pred_rf ← RF(X_test)
Evaluate_RF ← Evaluate_Model("Random Forest", y_test, y_pred_rf)

Phase 3: Train XGBoost Regressor
Define XGBRegressor with n_estimators = 100, learning_rate = 0.1, max_depth = 4
Train XGB on (X_train, y_train)
Predict y_pred_xgb ← XGB(X_test)
Evaluate_XGB ← Evaluate_Model("XGBoost", y_test, y_pred_xgb)

Create results table:
    Model | MAE | MSE | R² Score
    MLP   | Evaluate_MLP
    RF    | Evaluate_RF
    XGB   | Evaluate_XGB
Print Model Comparison Summary.
End
```

The above code gives the output in terms of MAE, MSE & R^2 as tabulated in Table 4 for MLP model, Table 5 for RF model and Table 6 for XG boost model. It is observed that RF model has best prediction performance compared to MLP and XGBoost model, since it has lower value of MAE and MSE. The R^2 value also equals to one, which defines the best prediction model.

Table 4: Evaluation metrics for MLP model

	MAE	MSE	R^2
Vs1_	0.45457	0.270441	0.99999
Vs2_	0.30953	0.151271	0.99999
Vs3_	0.29449	0.112273	0.99999

Table 5: Evaluation metrics for RF model

	MAE	MSE	R^2
Vs1	0.01358	0.001594	1.00
Vs2	0.01456	0.001828	1.00
Vs3	0.01480	0.001545	1.00

Table 6: Evaluation metrics for XGBoost model

	MAE	MSE	R^2
Vs1_Vs	0.687878	0.961183	0.999965
Vs2_Vs	0.674199	0.906099	0.999967
Vs3_Vs	0.682151	0.921466	0.999967

Power System Technology
ISSN:1000-3673

Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

The figure 4, figure 5 and figure 6 indicates the graph plotted in terms of actual value and predicted value for all three phase voltages separately.

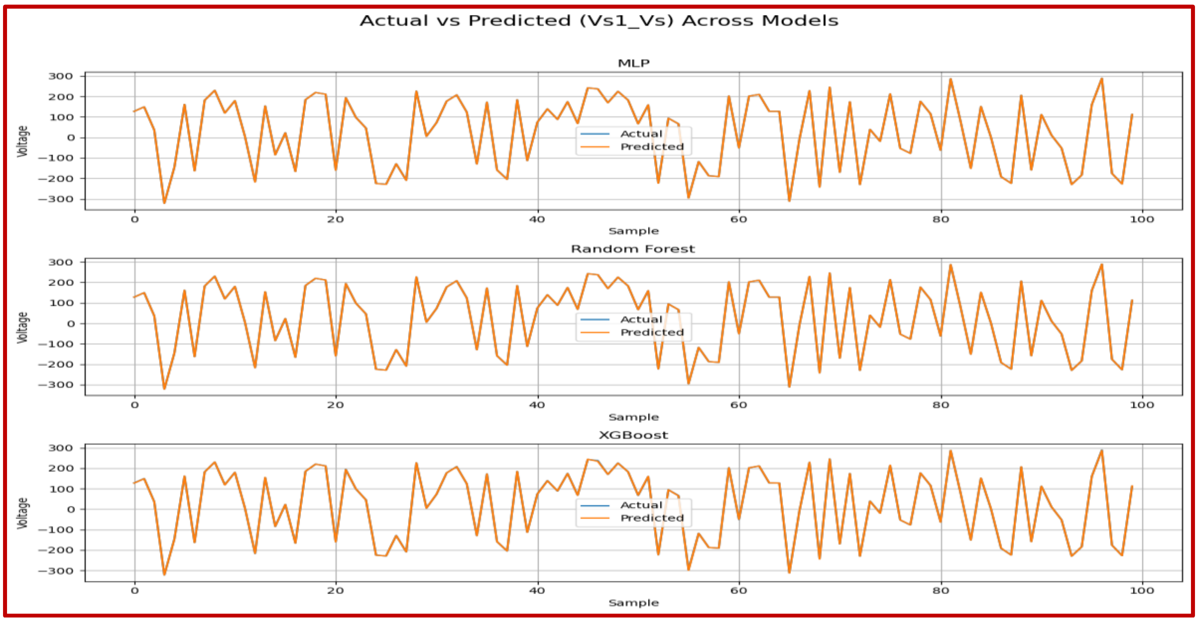


Figure 4: Actual and Predicted voltage plot for MLP, RF & XGBoost model for Vs1.

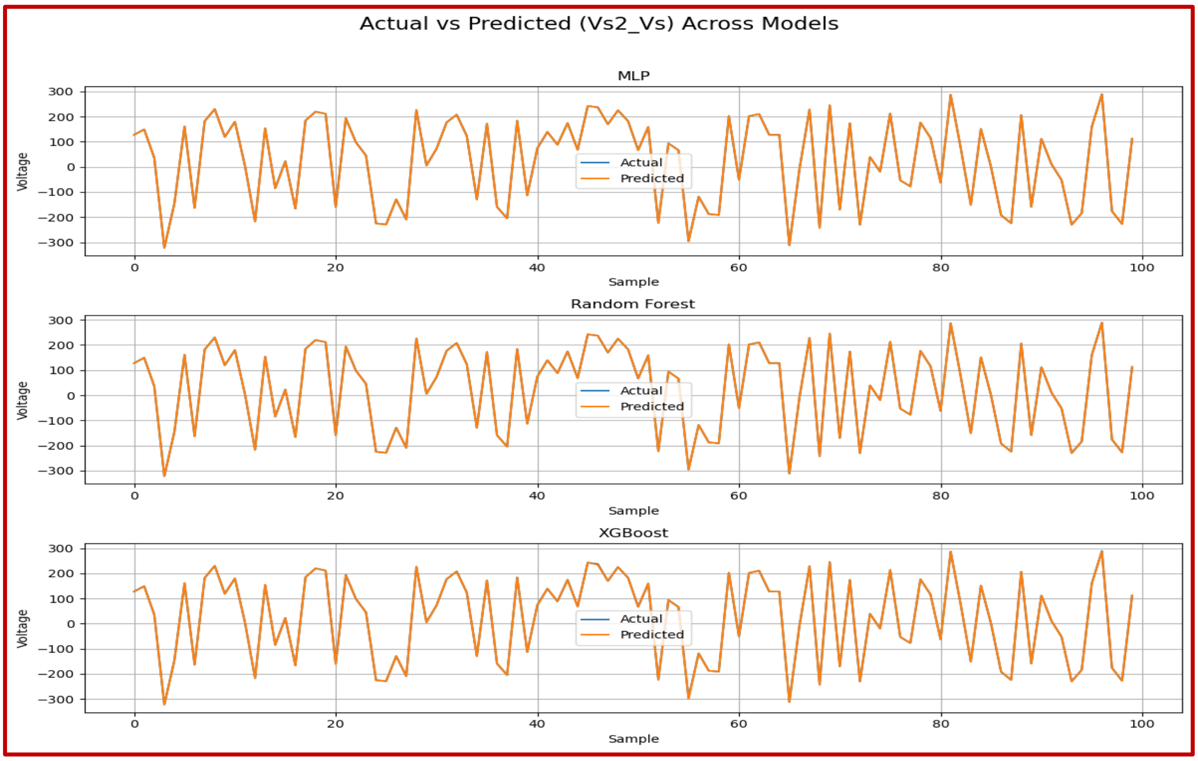


Figure 5: Actual and Predicted voltage plot for MLP, RF & XGBoost model for Vs2.

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

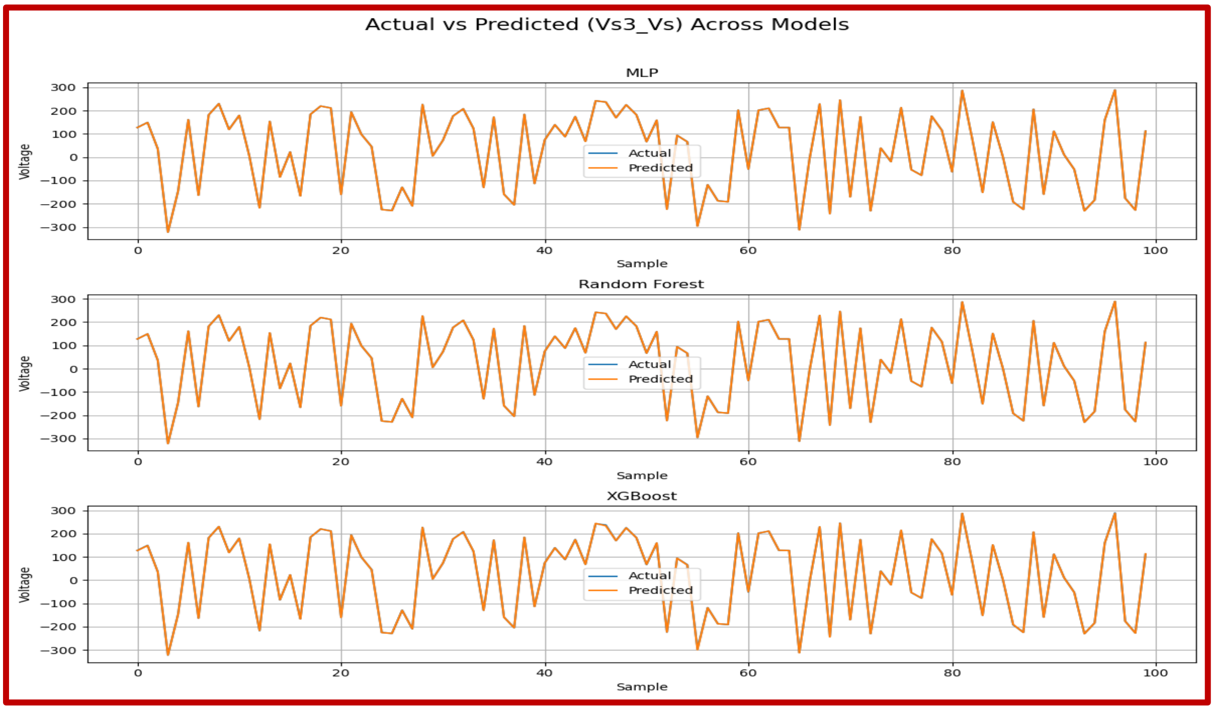


Figure 6: Actual and Predicted voltage plot for MLP, RF & XGBoost model for Vs3.

The Random Forest model exhibited the most accurate results across all voltage targets, outperforming deep learning models in terms of error metrics. However, its suitability for time-dependent data is limited compared to LSTM, which can better capture temporal patterns.

5. Discussion

In this section, the results of five different methods, such as multi-target LSTM, multi-output LSTM, ANFIS, MLP, random forest, and XGBoost algorithms, are consolidated in terms of mean absolute error, mean squared error, and coefficient of determination. The bar chart represents the comparison of all 5 methods under three different metrics. Figure 7 shows the representation of MAE, which indicates that the random forest algorithm has the least value of MAE. As per the standards, the lower the value of MAE, the better the prediction model. Hence RF model is considered as a bench mark model for determining the prediction accuracy out of LSTM and ANFIS. By comparing the output of LSTM and ANFIS model with RF model output, it is observed that ANFIS has least MAE error and proves to be the best predictive model compared to LSTM.

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

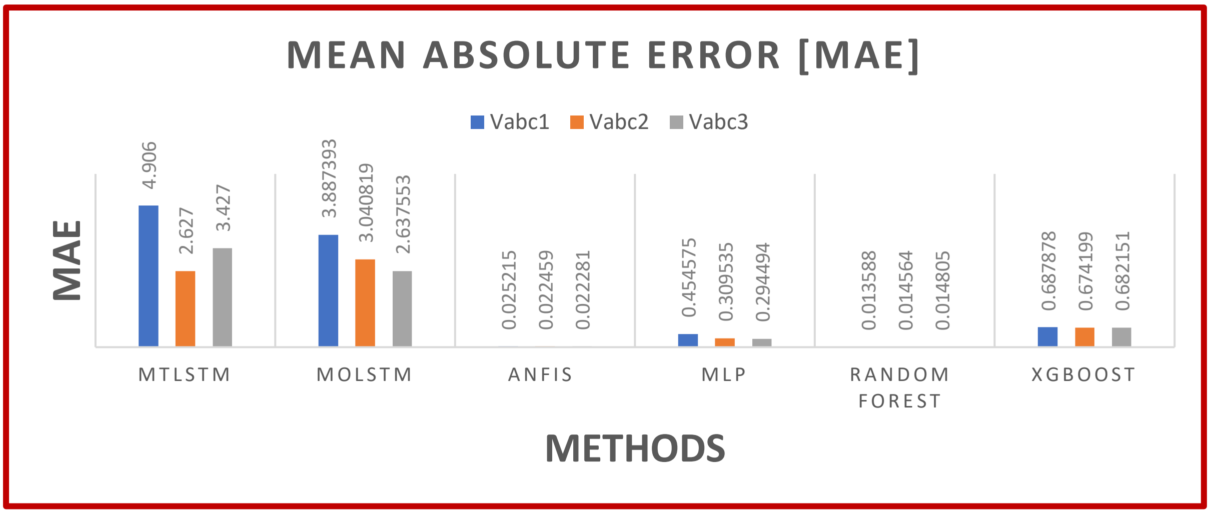


Figure 7: Comparison chart for Mean Absolute Error.

Figure 8 shows the representation of MSE, from which we can conclude that ANFIS has less value of MSE compared to LSTM model. The value of MSE is almost nearer to the benchmark value obtained from RF model. Hence even in terms of MSE, Anfis model proves to be the best prediction model.

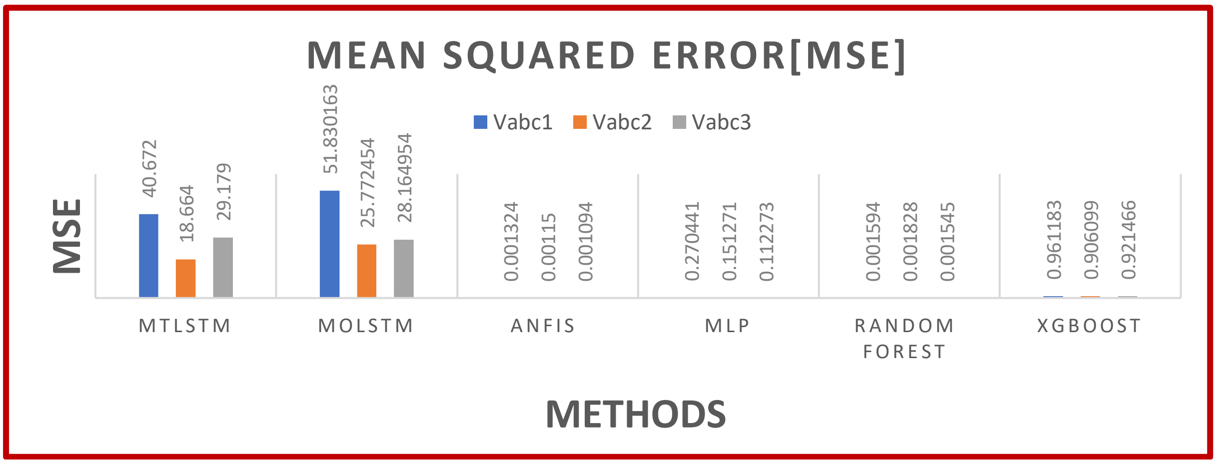


Figure 8: Comparison chart for Mean Squared Error.

Figure 9 shows the representation of R^2 value, which indicates that the random forest algorithm has the highest value of R^2, i.e, 1. As per the standards, if $R^2 = 1$, it is considered to be the best prediction model. ANFIS model also has the R^2 value equal to 1. This proves that ANFIS is best prediction model in terms of R^2 also.

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

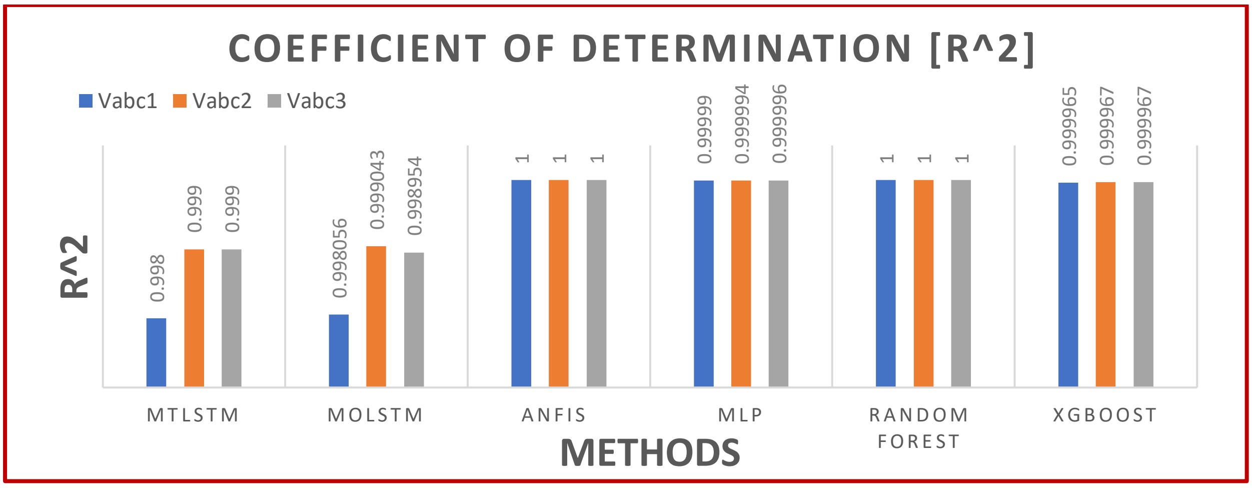


Figure 9: Comparison chart for coefficient of determination value.

Hence, the study justifies that ANFIS model have the best prediction capability compared to LSTM model in terms of all evaluation metrics. This prediction can be further used to generate control signal to provide suitable compensation for the voltage related PQ issues in terms of reactive power injection.

6. Conclusion and future scope

This study presents an intelligent power quality forecasting framework using LSTM and ANFIS models, benchmarked against conventional machine learning techniques. Through simulation-based data generation and structured time-series analysis, multiple models were trained and evaluated under identical conditions to ensure fair comparison.

The ANFIS models demonstrated high predictive accuracy, confirming their effectiveness in handling sequential PQ data. LSTM model, both multi-target and multi-output model, provided an interpretable alternative with competitive performance. Among the benchmark algorithms, Random Forest emerged as the top performer in terms of numerical accuracy, though it lacks the ability to model temporal dependencies.

Overall, the results validate the potential of deep learning and neuro-fuzzy systems in enhancing proactive power quality management in smart grids. The proposed approach can be extended to include additional PQ parameters and real-time deployment for advanced fault detection and grid optimization. Future work may involve integrating real sensor data, optimizing model parameters further, and exploring ensemble strategies that combine temporal learning with feature interpretability.

Power System Technology
ISSN:1000-3673

Received: 16-06-2025 *Revised: 05-07-2025* *Accepted: 20-08-2025*

Refrences

[1] R. Machlev, A. Chachkes, J. Belikov, Y. Beck, and Y. Levron, "Open source dataset generator for power quality disturbances with deep-learning reference classifiers," *Electric Power Systems Research*, vol. 195, p. 107152, Mar. 2021, doi: 10.1016/j.epsr.2021.107152. Available: https://doi.org/10.1016/j.epsr.2021.107152.

[2] N. Mostafa, H. S. M. Ramadan, and O. Elfarouk, "Renewable energy management in smart grids by using big data analytics and machine learning," *Machine Learning with Applications*, vol. 9, Art. no. 100363, 2022, doi: 10.1016/j.mlwa.2022.100363.

[3] S. Paran and C. S. Edrington, "Improved Power Hardware-in-the-Loop Interface Methods via Impedance Matching," Florida State University Center for Advanced Power Systems, 2015. [Online].Available:https://www.researchgate.net/profile/SanazParan/publication/284031308_Improved_Power_Hardware_in_the_Loop_Interface_Methods_via_Impedance_Matching/links/564b566908aeab8ed5e741be/Improved-Power-Hardware-in-the-Loop-Interface-Methods-via-Impedance-Matching.pdf

[4] Vijayshree Gopal and Sumathi Srinivasan, "Power quality improvement in distributed generation system using intelligent control methods," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 32, no. 1, p. 33, Aug. 2023, Doi:10.11591/ijeecs.v32.i1.pp33-42.

[5] Y. Guo, Z. Yang, S. Feng, and J. Hu, "Complex Power System Status Monitoring and Evaluation Using Big Data Platform and Machine Learning Algorithms: A Review and a Case Study," *Complexity*, vol. 2018, Art. no. 8496187, Sep. 2018. [Online]. Available: https://doi.org/10.1155/2018/8496187

[6] B. Zayas, A. Espinosa, V. Sanchez, and J. Perez, "Getting ready for data analytics of electric power distribution systems," *International Journal of Computers*, vol. 2, pp. 179–186, 2017. [Online]. Available: https://iaras.org/iaras/filedownloads/ijc/2017/006-0027%282017%29.pdf

[7] Y. Gu, *Renewable Energy Integration in Distribution System – Synchrophasor Sensor Based Big Data Analysis, Visualization, and System Operation*, Ph.D. dissertation, School of Engineering and Computer Science, Univ. of Denver, Denver, CO, USA, Jun. 2018. Advisor: J. J. Zhang. [Online]. Available: arXiv:1803.06076

[8] C. A. Gómez Tapias, A. A. Cuevas Murcia, and H. E. Rojas Cubides, "Characterization and Evaluation of Power Quality Disturbances Using Python," *Pistas Educativas*, vol. 15, no. 45, pp. 1–15, Apr. 2024. [Online]. Available: https://pistaseducativas.celaya.tecnm.mx/index.php/pistas/article/viewFile/3393/2536

[9] F. Milano, "A Python-Based Software Tool for Power System Analysis," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 119–127, Feb. 2011. [Online]. Available: http://faraday1.ucd.ie/archive/papers/dome.pdf

[10] J.-M. Sierra-Fernández, O. Florencias-Oliveros, M.-J. Espinosa-Gavira, J.-J. González-de-la-Rosa, A. Agüera-Pérez, and J.-C. Palomares-Salas, "Online System for Power Quality Operational Data Management in Frequency Monitoring Using Python and

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

Grafana," *Energies*, vol. 14, no. 24, Art. no. 8304, Dec. 2021, doi: 10.3390/en14248304. [Online]. Available: MDPI Energies Journal

[11] S. Boopathy, P. S. Kumar, and M. Karaaslan, "Predictive Analytics With Data Visualization," *Research Square*, preprint, Nov. 2022. [Online]. Available: https://www.researchsquare.com/article/rs-803205/latest.pdf

[12] J. Zhu, E. Zhuang, C. Ivanov, and Z. Yao, "A Data-Driven Approach to Interactive Visualization of Power Systems," *IEEE Trans. Power Syst.*, vol. 26, no. 4, pp. 2094–2103, Oct. 2011, doi: 10.1109/TPWRS.2011.2119499.

[13] R. Huang and S. Yu, "Data Collection and Visual Analysis of New Power System Hot Words Based on Python," in *Proc. ICIDC 2023*, Nanchang, China, Jun. 2023. [Online]. Available: https://eudl.eu/pdf/10.4108/eai.2-6-2023.2334604

[14] P. Liu, Z. Wang, S. Wei, Y. Bo, and S. Pu, "Recent Developments of Modulation and Control for High-Power Current-Source-Converters Fed Electric Machine Systems," *CES Trans. Electr. Mach. Syst.*, vol. 4, no. 3, pp. 215–226, Sep. 2020, doi: 10.30941/CESTEMS.2020.00027. [Online]. Available: CES Transactions PDF

[15] R. Itagi, "Data Analytics in Solar Photovoltaics Power Forecasting for Smart Grid Applications," in *Proc. 2021 Int. Conf. on Intelligent Technologies (CONIT)*, Hubli, India, Jun. 2021, pp. 1–6, doi: 10.1109/CONIT51480.2021.9498299.

[16] N. F. Akhter, A. Mia, and M. J. Talukder, "Python-Based Hybrid AI Models for Real-Time Grid Stability Analysis in Solar Energy Networks," *Frontiers in Applied Engineering and Technology*, vol. 1, no. 1, pp. 139–156, Dec. 2024, doi: 10.70937/faet.v1i01.24. [Online]. Available: Academia.edu PDF

[17] A. Jain and S. C. Gupta, "Optimal placement of distributed generation in power distribution system and evaluating the losses and voltage using machine learning algorithms," *Frontiers in Energy Research*, vol. 12, Art. no. 1378242, Apr. 2024. [Online]. Available: https://www.frontiersin.org/journals/energyresearch/articles/10.3389/fenrg.2024.1378242/full

[18] S. Preda, S.-V. Oprea, A. Bâra, and A. Belciu (Velicanu), "PV Forecasting Using Support Vector Machine Learning in a Big Data Analytics Context," *Symmetry*, vol. 10, no. 12, Art. no. 748, Dec. 2018, doi: 10.3390/sym10120748. [Online]. Available: MDPI Symmetry Journal

[19] L. Thurner, A. Scheidler, F. Schäfer, J.-H. Menke, J. Dollichon, F. Meier, S. Meinecke, and M. Braun, "pandapower—An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6510–6521, Nov. 2018, doi: 10.1109/TPWRS.2018.2829021. [Online]. Available: arXiv:1709.06743

[20] M. Sabahi, A. Safari, and M. Nazari-Heris, "Design and implementation of a cost-effective practical single-phase power quality analyzer using pyboard microcontroller and python-to-python interface," *J. Eng.*, vol. 2024, Art. no. e12360, Feb. 2024. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/tje2.12360

[21] P. K. Persis and R. Geetha, "Visualization and Interpretation of Lithium Iron Phosphate Cell Variable Characteristics by Adapting Python Data Science Packages," in *Proc. 2024*

Power System Technology
ISSN:1000-3673
Received: 16-06-2025 Revised: 05-07-2025 Accepted: 20-08-2025

IEEE Silchar Subsection Conference (SILCON), Silchar, India, Jun. 2024, pp. 1–6. ISBN: 979-8-3315-4082-1. [Online]. Available: IEEE TIB Portal
[22] Z. Liang, Z. Liang, Y. Zheng, B. Liang, and L. Zheng, "Data Analysis and Visualization Platform Design for Batteries Using Flask-Based Python Web Service," *World Electr. Veh. J.*, vol. 12, no. 4, Art. no. 187, Oct. 2021, doi: 10.3390/wevj12040187. [Online]. Available: MDPI Journal Link
[23] H. Jiang, Y. Zhang, E. Muljadi, Y. Gu, J. Zhang, and T. Gao, "Knowledge Discovery for Smart Grid Operation, Control, and Situation Awareness – A Big Data Visualization Platform," in *Proc. 2016 North American Power Symposium (NAPS)*, Denver, CO, USA, Sep. 2016, pp. 1–6, doi: 10.1109/NAPS.2016.7747892. [Online]. Available: NREL Research Hub
[24] Q.-k. Sun, J.-b. Wang, T.-j. Jing, and Y.-h. Guo, "Research on Voltage Quality Analysis System Based on Kernel Density Estimation for Power Grid," in *Proc. IEEE Conf. on Energy Internet and Energy System Integration (EI2)*, Beijing, China, Nov. 2017, pp. 1–6.
[25] G. Van Houdt, C. Mosquera, and G. Nápoles, "A review on the long short-term memory model," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5929–5955, May 2020, doi: 10.1007/s10462-020-09838-1.