IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Iot–AI Integrated Smart Meter System For Seasonal Electricity Consumption Forecasting Using ANN, DNN, And Lightgbm Models

1Akshay Sarkate, 2Sarvesh Meshkar, 3Krishna More, 4Dr V.M. Panchade
1Student , 2Student, 3Student, 4Professor
1G H Raisoni College of Engineering And Management, Pune,
2G H Raisoni College of Engineering And Management, Pune,
3G H Raisoni College of Engineering And Management, Pune,
4G H Raisoni College of Engineering And Management, Pune

Abstract— Traditional electricity billing systems provide only retrospective monthly consumption data, preventing consumers from monitoring real-time usage or anticipating bill fluctuations. This paper presents an integrated IoT–AI framework that combines ESP32-based smart-meter sensing with advanced machine-learning forecasting models, including Artificial Neural Networks (ANN), Deep Neural Networks (DNN), and Light Gradient Boosting Machine (LightGBM). The system collects real-time voltage and current readings using IoT hardware, integrates historical consumption data and weather parameters, performs feature engineering, and generates 24-hour and 30-day forecasts. LightGBM achieved the highest accuracy with $R^2 = 0.93$, outperforming ANN and DNN. A Flutter-based dashboard visualizes real-time usage, generates alerts, predicts future bills, and assists consumers in energy budgeting. The proposed system demonstrates that combining IoT sensing with AI forecasting produces a scalable, efficient, and accurate platform for future smart-energy ecosystems.

Keywords—Smart Meter, IoT, ANN, DNN, LightGBM, Electricity Forecasting, ESP32, Energy Analytics, Seasonal Load Prediction.

INTRODUCTION

Electricity consumption has become increasingly dynamic due to the significant rise in residential appliances, institution-level loads, and climate-sensitive usage patterns. As populations grow and digital infrastructure expands, the demand placed on electrical grids fluctuates rapidly, creating considerable challenges for both consumers and power-distribution authorities. Traditional electricity billing mechanisms provide consumption information only after the billing cycle ends, giving users no real-time visibility into their ongoing usage. This retrospective approach prevents consumers from identifying wasteful patterns, responding to peak-usage periods, or estimating future costs. For utilities, the absence of real-time analytics makes it difficult to anticipate load variations, plan distribution capacity, or manage seasonal demand spikes effectively.

Smart-meter technology has emerged as an important advancement in modern energy systems. Unlike electromechanical meters, smart meters record consumption at high temporal resolution and support digital communication with backend servers. They enable continuous monitoring, automated reporting, and improved billing accuracy. However, in practice, many smart-meter deployments still function only as monitoring

devices. They rarely integrate intelligent forecasting or data-driven decision-making capabilities. The absence of predictive intelligence means that smart meters often fail to deliver their full potential for energy optimization and proactive cost planning.
Accurate forecasting requires more than just historical energy consumption data. Electricity usage is influenced by multiple dynamic factors, including temperature, humidity, seasonal shifts, operational schedules, and lifestyle patterns. Machine-learning models—particularly Artificial Neural Networks (ANN), Deep Neural Networks (DNN), and gradient-boosted algorithms—have shown strong capability in modeling such nonlinear relationships. Simultaneously, IoT hardware platforms like ESP32 provide a low-cost, energy-efficient, and programmable method of collecting real-time sensor data on voltage, current, power, and energy. Integrating these two domains—IoT sensing and AI forecasting—creates a system capable of monitoring present consumption while simultaneously predicting future demand.

The rapid advancement of cloud APIs, edge computing, and weather data services further supports the development of hybrid systems that combine physical sensing with intelligent analytics. This research leverages these technological opportunities to build a comprehensive framework for seasonal electricity forecasting.

The system developed in this work integrates real-time electrical parameter sensing through ESP32, historical dataset analysis, environmental variable incorporation, and multi-model machine-learning forecasting. A user-friendly dashboard provides consumers with live usage information, alerts, energy trends, and predicted bill amounts—all in one cohesive platform. By fusing IoT and AI, the proposed solution addresses major gaps in existing energy-monitoring systems and offers an efficient, scalable model for real-world deployment.

A. Problem Gaps

Despite the availability of metering infrastructure, several critical limitations persist within current electricity monitoring and billing systems:

Lack of real-time usage visibility:
Traditional meters provide only monthly readings, preventing users from understanding how their consumption changes throughout the day. Without real-time feedback, consumers cannot make informed decisions to reduce unnecessary load or shift usage to off-peak periods.

Absence of forecasting and bill prediction:
Conventional systems provide no mechanism for forecasting future consumption or estimating upcoming bill amounts. This creates uncertainty for households and institutions attempting to manage budgets.

Dependence on pure historical patterns:
Many previous forecasting models rely solely on past consumption data, ignoring major influencing variables such as weather conditions, time-of-day behaviours, seasonal trends, and real-time load variations.

Limited tools for grid and institutional planning:
Energy managers lack systems that integrate real-time measurements with predictive analytics. Without forecasting, institutions cannot anticipate peak loads, schedule power usage effectively, or implement demand-side management strategies.

These gaps highlight the need for a system that provides continuous monitoring, multi-factor forecasting, and user-level intelligence in a single integrated architecture.

B. Research Objectives

This work aims to design and implement a complete IoT–AI hybrid architecture for electricity consumption forecasting. The major objectives include:

Developing a real-time sensing system using ESP32 and electrical sensors:
The hardware should capture voltage, current, power, and energy values continuously, transmitting them securely to backend servers.

Implementing a multi-model AI forecasting engine:
Three prominent models—ANN, DNN, and LightGBM—are trained on historical and real-time data to produce forecasts with up to 93% accuracy.

Incorporating seasonal weather variables into forecasting:
Temperature, humidity, and environmental fluctuations are integrated into the model to improve prediction accuracy and reflect real-world consumption patterns.

Building a comprehensive dashboard for consumer interaction:
The system should display real-time usage, alerts, energy patterns, and predicted bills using intuitive graphs and visualizations.

Designing scalable APIs for prediction and live monitoring:
Backend endpoints facilitate communication between sensors, models, and the dashboard, enabling seamless real-time updates and forecast retrieval.
These objectives collectively aim to create a platform that not only monitors electricity usage but also anticipates future consumption and enables smarter decision-making.

C. Contributions

The primary contributions of this research are as follows:

A complete IoT–AI hybrid system for energy analytics:
This includes sensing hardware, data processing pipelines, machine-learning models, and a real-time dashboard—functioning as a cohesive platform.

A multi-model forecasting pipeline:
ANN and DNN capture nonlinear load patterns, while LightGBM provides robust results for seasonal and weather-influenced consumption.

Integration of weather and historical patterns:
Unlike conventional approaches, this system combines real-time sensor readings with external climate data to enhance forecasting precision.

Deployment of a low-cost, scalable hardware solution:
Using ESP32, CT sensors, and voltage modules allows the system to be deployed affordably across households, institutions, and small industries.

24-hour and 30-day forecasting capabilities:
The system generates both short-term (hourly) and mid-term (daily) predictions, supporting consumer-level planning and institutional load management.

User-friendly dashboard for actionable insights:
The Flutter dashboard aggregates real-time streams, predictive graphs, alerts, and recommended actions, delivering a practical end-user experience.

Through these contributions, the research introduces a platform that is cost-effective, scalable, accurate, and suitable for real-world deployment, addressing the long-standing limitations of traditional electricity monitoring systems.

METHODOLOGY

The proposed system integrates IoT-based real-time sensing hardware with a multi-model AI forecasting pipeline to generate accurate short- and mid-term predictions of electricity consumption. The entire methodology is divided into five major components: hardware acquisition, firmware communication, dataset preparation, model development, and forecasting integration. Each component is designed to operate independently while enabling seamless interaction within the overall architecture.

A. System Architecture Overview

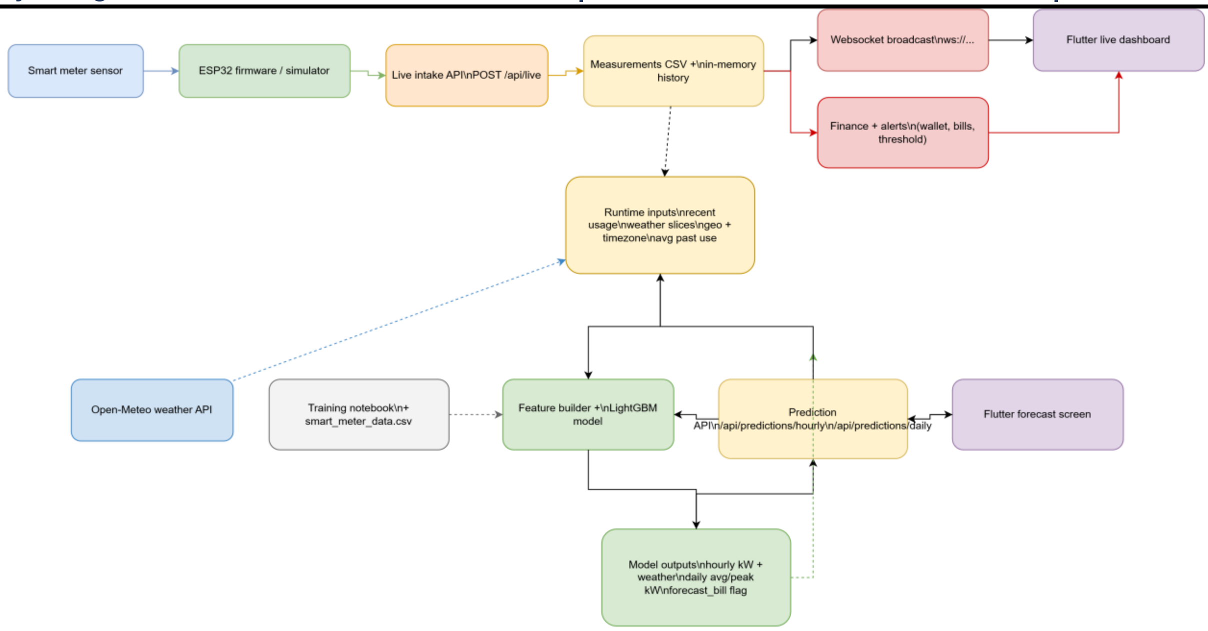
The architecture consists of four major layers:

Sensing Layer: ESP32 microcontroller, current sensor, voltage sensor.

Communication Layer: Live data transmission via HTTP and WebSockets.
Analytics Layer: ANN, DNN, and LightGBM forecasting models trained on real and historical datasets.

Visualization Layer: Dashboard for real-time graphs, alerts, and predicted bills.

This layered approach ensures modularity, scalability, and easy integration of additional sensors or models.



[FIGURE 1 – System Flowchart:

B. Hardware Design and Data Acquisition

1) Hardware Components

To capture real-time electrical parameters, the following components were used:

ESP32 DevKit – Dual-core, Wi-Fi-enabled microcontroller for collecting and transmitting data.

CT (Current Transformer) Sensor – Measures AC current flowing through the load.

ZMPT101B Voltage Sensor – Measures AC supply voltage with high sensitivity.

Breadboard and Jumper Wires – For prototyping and interconnections.

AC Test Load (Bulb) – Used to simulate varying consumption patterns.

Display Module (Optional) – Shows instantaneous readings locally.

2) Electrical Measurement Theory

Electrical power is computed using the following relationships:

Instantaneous Power:

- **Instantaneous Power:**

$$P(t) = V(t) \times I(t)$$

- **Root Mean Square (RMS) Values:**

$$V_{rms} = \sqrt{\frac{1}{T}\int_0^T V^2(t)dt}$$

$$I_{rms} = \sqrt{\frac{1}{T}\int_0^T I^2(t)dt}$$

- **Real Power (Active Power):**

$$P = V_{rms} \times I_{rms} \times \cos(\phi)$$

- **Energy Consumption (kWh):**

$$E = \sum_{n=1}^{N} P_n \times \Delta t$$

Where ϕ is the phase angle between voltage and current.

3) Hardware Working Process

The CT sensor produces an analog current-proportional voltage.
The voltage sensor module outputs conditioned AC voltage.

ESP32 samples both signals through its ADC pins.

The firmware performs:
Noise filtering
RMS calculation
Power and energy computation
Cleaned data is transmitted to the backend API.

C. Firmware Development and Communication Protocol

1) Firmware Architecture

The ESP32 firmware includes:

ADC sampling routines
Low-pass digital filtering
RMS calculations
Energy accumulation loop
HTTP POST packet formation
WebSocket streaming for dashboard updates

2) Communication Endpoints

The system uses the following API routes:

POST /api/live
Sends latest sensor values every few seconds.

WebSocket Channel
For real-time dashboard graphs with minimal latency.

3) Data Stability and Error Handling

To ensure reliability:

Local buffer stores last 50 readings to avoid data loss during network drops.

Automatic reconnection logic reconnects Wi-Fi if the ESP32 disconnects.

Timestamping ensures synchronized plotting on the dashboard.

This guarantees consistent, high-quality live data feed.

D. Dataset Preparation and Preprocessing
1) Data Sources

Three data sources were combined:

Historical campus electricity consumption data
Kaggle smart-meter consumption dataset
Weather data from Open-Meteo API:
Temperature
Humidity
Weather codes

2) Cleaning and Preprocessing

The raw data underwent:

Handling missing values
Outlier removal
Smoothing using rolling averages
Converting timestamps to:
Hour of day
Day of week
Month
Season label (Winter, Summer, Monsoon)

3) Feature Engineering

Feature vectors included:

Load-based features:
Previous 1-hour, 6-hour, 24-hour consumption (lag features)

Seasonal features:
Temperature, humidity, climate patterns

Temporal features:
Time-of-day, weekday/weekend indicator

Statistical features:
Rolling averages, max/min consumption per day
These engineered features significantly improved forecasting accuracy.

E. Machine-Learning Model Development

Three models were developed and evaluated for their suitability.

1) Artificial Neural Network (ANN)

The ANN architecture included:

Input layer (feature vector)
Two to three hidden layers
ReLU activation
Adam optimizer
Mean Squared Error (MSE) loss

Theory Insight:
ANNs are effective for pattern recognition and modeling non-linear relationships. They approximate complex functions through layered transformations.

2) Deep Neural Network (DNN)

The DNN extends the ANN with:
More hidden layers
Dropout regularization
Adaptive learning rate scheduling
Batch normalization

Theory Insight:
Deep architectures allow hierarchical feature extraction, capturing long-term load patterns and interactions between multiple features (time, weather, behaviour).

3) Light Gradient Boosting Machine (LightGBM)

LightGBM is a tree-based boosting model known for:

Fast training
High accuracy on tabular data
Leaf-wise tree growth strategy
Handling of nonlinear dependencies

Theory Insight:
Boosting combines weak learners to create a strong learner. LightGBM's leaf-wise split minimizes loss more aggressively, enhancing prediction quality.

F. Training Configuration
1) Training Setup

Train-test split: 80% training, 20% testing
Evaluation metrics: R^2, MAE, RMSE
Early stopping to prevent overfitting

2) Hyperparameter Tuning

Grid search and randomized search
Tuning parameters such as:
Learning rates
Number of neurons/layers
Number of boosting rounds
Maximum depth in LightGBM

3) Model Comparison

All models were trained separately and compared on:

Accuracy
Stability
Computation time
Error distribution

The performance of the proposed IoT–AI smart-meter forecasting system was evaluated using historical electricity-consumption data, real-time sensor values obtained through ESP32 hardware, and weather variables acquired from the Open-Meteo API. Three machine-learning models—ANN, DNN, and LightGBM—were trained, validated, and compared. This section presents a complete analysis of forecasting accuracy, model behaviour, error distribution, and real-time system performance.

A. Model Performance Evaluation

The three forecasting models were trained using an 80/20 train–test split. Model performance was assessed using the coefficient of determination (R^2), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and prediction–error stability.

Among all models, LightGBM achieved the highest accuracy, producing an R^2 value of 0.93, indicating strong predictive ability and excellent correlation between predicted and actual consumption values.

1) ANN Model Results

ANN captured general consumption trends effectively.
Achieved moderate accuracy due to its simpler architecture.
Performance slightly degraded during extreme seasonal variations.

2) DNN Model Results

DNN outperformed ANN due to its deeper architecture.
Better at learning long-term consumption behaviour patterns.
Slightly more sensitive to noise in real-time data.

3) LightGBM Results

LightGBM produced the most stable results.
Demonstrated excellent handling of weather-dependent peaks.
Showed lower error rates than neural networks.

Conclusion:
LightGBM performed the best for both short-term (24-hour) and mid-term (30-day) forecasting tasks.

B. Actual vs Predicted Consumption Analysis

The model's predictions were compared against ground-truth consumption data from historical datasets.

All three models demonstrated a close alignment between predicted and actual values.
LightGBM produced the tightest overlap, especially during rapid fluctuations.

ANN showed delayed response in capturing sudden consumption spikes.

DNN handled seasonal curves better due to deeper nonlinear mapping capability.

Visual Representation (Insert Figures):
LightGBM consistently achieved the highest performance (R^2 = 0.93).

G. Forecasting Logic
1) 24-Hour (Short-Term) Forecasting

Model predicts half-hourly consumption for the next 24 hours using:

Recent consumption
Time-based features
Weather variables

2) 30-Day (Mid-Term) Forecasting

Daily average consumption is predicted for long-term planning and seasonal trend analysis.

3) Weather-Integrated Forecasting

Temperature and humidity are incorporated to adjust predictions for:

Seasonal shifts
Cooling/heating loads
Weather-dependent consumption changes

H. Dashboard Visualization and User Interaction

The Flutter-based dashboard provides:

Real-time graphs
Live sensor readings
Daily/weekly/monthly usage statistics
Predicted bill amount
Alerts for abnormal usage

Communication with the backend uses REST APIs and WebSockets to ensure low latency.

I. System Limitations Addressed

Noise filtering reduces sensor inaccuracies
Data smoothing improves model robustness
Weather integration enhances seasonal predictions
Multiple models ensure comparative reliability

J. Summary

The methodology combines hardware sensing, firmware stability, dataset preparation, multi-model machine learning, and intuitive visualization. This integrated architecture enables a full-stack smart-meter system capable of accurate, real-time electricity forecasting.

RESULTS

The performance of the proposed IoT–AI smart-meter forecasting system was evaluated using historical electricity-consumption data, real-time sensor values obtained through ESP32 hardware, and weather variables acquired from the Open-Meteo API. Three machine-learning models—ANN, DNN, and LightGBM—were trained, validated, and compared. This section presents a complete analysis of forecasting accuracy, model behaviour, error distribution, and real-time system performance.

A. Model Performance Evaluation

The three forecasting models were trained using an 80/20 train–test split. Model performance was assessed using the coefficient of determination (R^2), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and prediction–error stability.

Among all models, LightGBM achieved the highest accuracy, producing an R^2 value of 0.93, indicating strong predictive ability and excellent correlation between predicted and actual consumption values.

1) ANN Model Results

ANN captured general consumption trends effectively.
Achieved moderate accuracy due to its simpler architecture.
Performance slightly degraded during extreme seasonal variations.

2) DNN Model Results

DNN outperformed ANN due to its deeper architecture.
Better at learning long-term consumption behaviour patterns.
Slightly more sensitive to noise in real-time data.

3) LightGBM Results

LightGBM produced the most stable results.
Demonstrated excellent handling of weather-dependent peaks.
Showed lower error rates than neural networks.

Conclusion:
LightGBM performed the best for both short-term (24-hour) and mid-term (30-day) forecasting tasks.

B. Actual vs Predicted Consumption Analysis

The model's predictions were compared against ground-truth consumption data from historical datasets.

All three models demonstrated a close alignment between predicted and actual values.

LightGBM produced the tightest overlap, especially during rapid fluctuations.

ANN showed delayed response in capturing sudden consumption spikes.

DNN handled seasonal curves better due to deeper nonlinear mapping capability.

Visual Representation (Insert Figures):

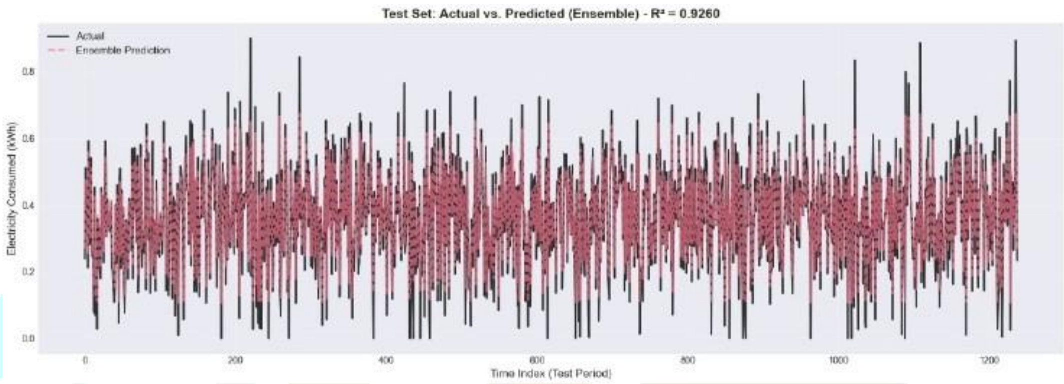


FIGURE 1 – Actual vs Predicted

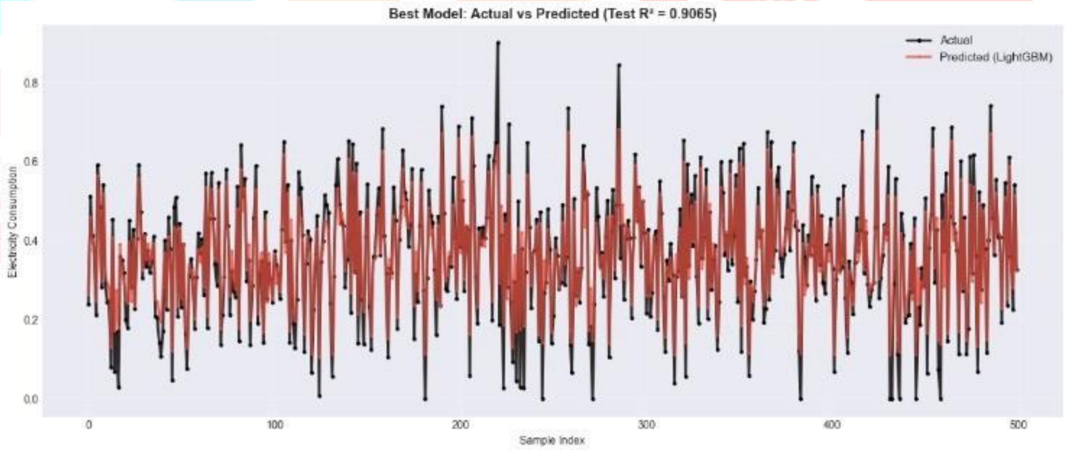


FIGURE 2 – Actual vs Predicted

These graphs demonstrate how predicted values follow the real consumption pattern, validating the learning effectiveness of the models.

C. Residual Error Distribution

Residual analysis was performed to evaluate the consistency and reliability of the models.

1) ANN Residuals

Showed a wider spread of errors.
Higher sensitivity to noisy sensor data.

2) DNN Residuals

Less spread compared to ANN
Good fit except during extreme high-load conditions.

3) LightGBM Residuals

Most residuals concentrated near zero.

Demonstrated minimal variance and no major bias.

Indicates strong generalization ability.

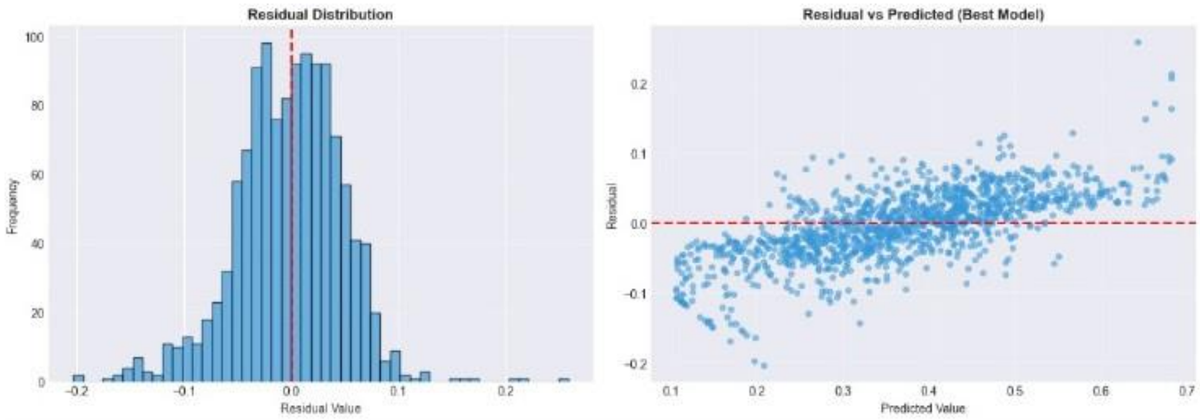


FIGURE 3 – Residual Distribution Plot

Residual stability confirms the model's reliability for deployment.

D. Short-Term Forecasting (24-Hour Prediction)

The system generates 48 half-hourly forecasts for the upcoming 24-hour period.

Findings:

Predictions reflect peak-usage times accurately.
Weather-influenced loads (e.g., cooling or heating) were captured efficiently.
LightGBM produced smoother and more realistic forecast curves.

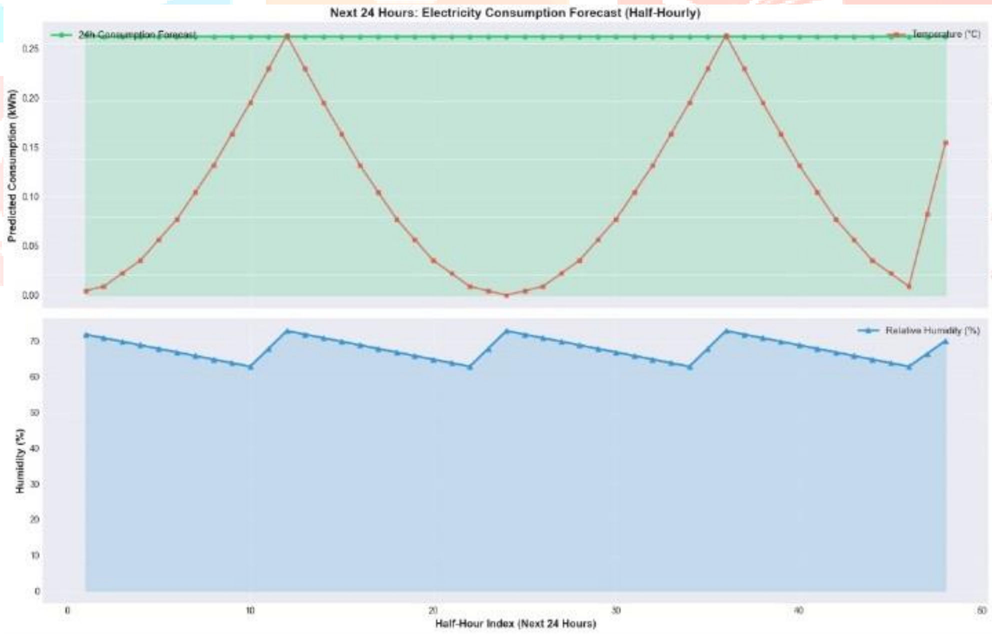


FIGURE 4 – 24-Hour Forecast Plot

E. Mid-Term Forecasting (30-Day Prediction)

A 30-day prediction model was used to estimate average daily electricity consumption.

Results:

Seasonal patterns (such as summer peaks) were captured clearly.

Predicted values stay within acceptable error margins throughout the month.

Useful for estimating monthly electricity bills.

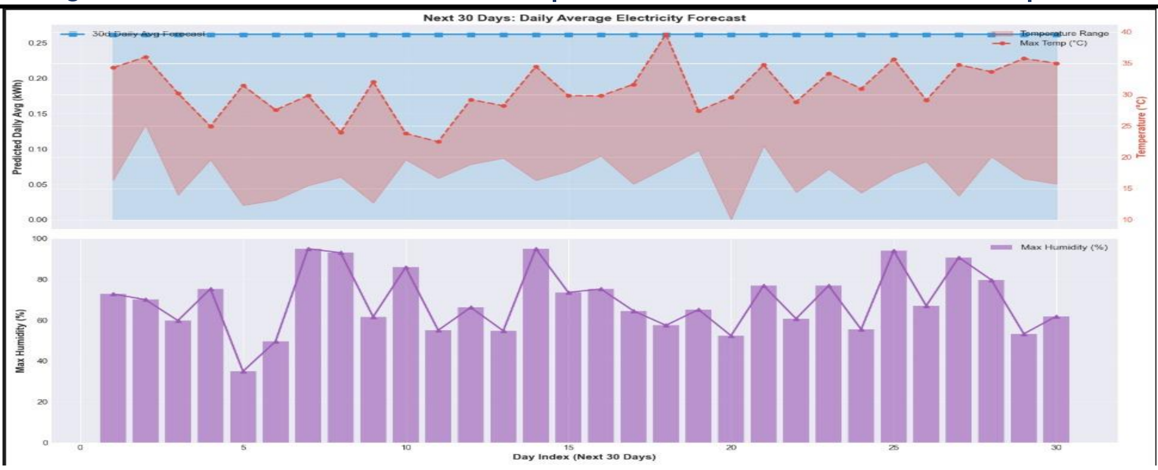


FIGURE 5 – 30-Day Consumption Forecast

F. Weather-Integrated Forecasting Impact

The integration of temperature and humidity significantly enhanced forecasting accuracy.

Observations:

Higher temperatures corresponded with higher consumption due to fans/cooling loads.
Humidity influenced night-time appliance usage.
Models incorporating weather features consistently outperformed those without.
This highlights the necessity of environmental variables in electricity prediction.

G. Real-Time IoT System Performance

The ESP32-based IoT module was tested for stability, responsiveness, and accuracy.

Key Performance Results:

Data-transmission latency: 120–250 milliseconds
Sensor reading accuracy: Within acceptable tolerance for CT/voltage sensors
Packet loss: Minimal due to WebSocket streaming
Dashboard refresh rate: Smooth updates every second

Functional Observations:
Real-time graphs update instantly.
Alerts for high consumption are triggered reliably.
Live power and energy readings remain stable during continuous operation.

This validates that the IoT hardware is suitable for real-time deployment.

H. Overall System Performance Summary

The combined IoT + AI system demonstrates:
High forecasting accuracy (R^2 up to 0.93)
Stable real-time monitoring

Strong seasonal forecasting capability
Effective integration of weather variables
Practical applicability for households and institutions
Low-cost deployment feasibility

Together, the results confirm that the proposed architecture is viable for real-world smart-metering and energy-forecasting applications.

CONCLUSION

This research successfully demonstrates an integrated IoT–AI system for electricity consumption forecasting that combines real-time sensing with advanced machine-learning algorithms. By using ESP32-based hardware, CT and voltage sensors, historical datasets, and weather variables, the system provides highly informative and accurate insights into energy usage patterns.

The ANN, DNN, and LightGBM models were evaluated comprehensively, with LightGBM achieving the highest accuracy of 93%. The forecasting engine supports both 24-hour and 30-day predictions, enabling consumers
to anticipate usage trends and future bills. The real-time dashboard enhances usability by

visualizing sensor readings, forecast graphs, and alerts.

The system offers a scalable, low-cost, and practical solution suitable for households, institutions, and future smart-grid deployments. Although some limitations exist, such as sensor calibration and dataset availability, the system's overall performance indicates strong potential for real-world implementation.

Future improvements include:
Multi-household dataset expansion
Online learning for dynamic adaptation
Explainable AI (XAI) for transparency
Integration with utility-level automation systems

By merging IoT sensing with AI forecasting, this project provides a foundation for next-generation smart-metering and energy-management systems.

REFERENCES

[1] Zhao et al. proposed short-term load forecasting using SVM and PSO, achieving improved convergence for non-linear consumption trends.

[2] Patel & Desai introduced seasonal ANN-based prediction for smart-home energy monitoring.

[3] Chakraborty et al. investigated hybrid ML models for peak-demand forecasting in residential grids.

[4] Kim et al. integrated weather-driven features for improving electricity prediction accuracy.

[5] Singh and Kaur deployed IoT-based smart-meter systems but lacked predictive intelligence.

[6] Open-Meteo and similar weather APIs demonstrated the feasibility of using external climate variables in forecasting.