IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 11, November 2025

DOI: 10.17148/IJARCCE.2025.1411117

Energy Consumption Forecasting in Smart Homes Using LSTM and XGBOOST Ensemble

Dr Arun Kumar GH[1], Karthik AS[2], Karthik KJ[3], Kruthin H Hoogar[4], Harsha Hosmat[5]

Associate Professor Dept. of ISE., Bapuji Instituite Of Engineering and Technology,Davanagere,Karnataka,India[1]

UG Student, Dept. of ISE., Bapuji Instituite Of Engineering and Technology,Davanagere,Karnataka,India[2]

UG Student, Dept. of ISE., Bapuji Instituite Of Engineering and Technology,Davanagere,Karnataka,India[3]

UG Student, Dept. of ISE., Bapuji Instituite Of Engineering and Technology,Davanagere,Karnataka,India[4]

UG Student, Dept. of ISE., Bapuji Instituite Of Engineering and Technology,Davanagere,Karnataka,India[5]

Abstract: Accurate prediction of residential electricity demand is essential for energy conservation, cost optimization, and effective grid planning. Smart homes generate large volumes of fine-grained consumption data, making them suitable candidates for advanced predictive modeling . This study proposes a hybrid forecasting framework that integrates Long Short-Term Memory (LSTM) networks with Extreme Gradient Boosting (XG Boost). LSTM captures temporal dependencies in consumption sequences, while XG Boost model nonlinear relationships in engineered features. The ensemble produces stable and adaptive predictions suitable for dynamic household environments. A web-based interface supports data upload, real-time forecasting, visualization, and cost estimation. Experimental results demonstrate that the hybrid model consistently outperforms standalone approaches in RMSE, MAE, and MAPE. The system provides interpretable predictions using feature-attribution techniques, enabling users to understand consumption drivers. This research contributes a practical and extensible solution for smart home energy management.

KEYWORDS: Smart Home Energy Forecasting, LSTM, XG Boost, Hybrid Ensemble Model, Deep Learning, Gradient Boosting, Smart Grid Optimization, Demand Response, Feature Engineering.

INTRODUCTION

Accurate prediction of residential electricity demand is essential for energy conservation, cost optimization, and effective grid planning. Smart homes generate large volumes of fine-grained consumption data, making them suitable candidates for advanced predictive modeling. This study proposes a hybrid forecasting framework that integrates Long Short-Term Memory (LSTM) networks with Extreme Gradient Boosting (XG Boost). LSTM captures temporal dependencies in consumption sequences, while XG Boost models nonlinear relationships in engineered features. The ensemble produces stable and adaptive predictions suitable for dynamic household environments. A web-based interface supports data upload, real-time forecasting, visualization, and cost estimation. Experimental results demonstrate that the hybrid model consistently outperforms standalone approaches in RMSE, MAE, and MAPE. The system provides interpretable predictions using feature-attribution techniques, enabling users to understand consumption drivers. This research contributes a practical and extensible solution for smart home energy management.might not have easy access to. For areas with vast road networks or inadequate infrastructure for routine inspections, this makes things even more difficult.

These systems provide previously unheard-of efficiency and accuracy when identifying and tracking road defects thanks to advancements in machine learning (ML), computer vision, and artificial intelligence (AI). The move toward automation represents a significant advancement in addressing the limitations of conventional solutions while satisfying the increasing demands for infrastructure upkeep.

The rapid growth of smart home technologies has transformed the way energy is consumed, monitored, and managed within residential environments. Modern smart homes are equipped with advanced sensors, smart meters, IoT-enabled appliances, and home automation systems capable of generating continuous, high-resolution energy consumption data. This interconnected ecosystem presents an unprecedented opportunity for developing intelligent forecasting systems that can analyze consumption patterns and provide accurate predictions to support energy efficiency and sustainable living.

However, residential electricity consumption is inherently complex due to the presence of diverse electrical appliances, irregular human behavior, and dynamic external conditions. Factors such as weather variations, seasonal patterns, occupancy levels, and lifestyle changes contribute to significant fluctuations in energy usage. Traditional approaches

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 14, Issue 11, November 2025

DOI: 10.17148/IJARCCE.2025.1411117

such as ARIMA, exponential smoothing, and basic regression techniques are limited in their ability to model these nonlinear and dynamic variations effectively. These limitations have motivated the shift toward more advanced machine learning and deep learning methods.

Machine learning models—especially tree-based ensembles like XG Boost—have demonstrated strong predictive performance on structured datasets. They excel at capturing nonlinear relationships and handling engineered features derived from energy profiles. At the same time, deep learning architectures such as Long Short-Term Memory (LSTM) networks have emerged as state-of-the-art models for sequential learning. LSTM effectively captures long-term dependencies in time-series data, making it ideal for understanding consumption trends that span across hours, days, or weeks.

I. RELATED WORK

Accurate energy-consumption forecasting has been an active research area for decades, progressing from traditional statistical methods to modern hybrid machine-learning pipelines.

Classical and statistical approaches. Early work relied on statistical time-series methods such as ARIMA and exponential smoothing. These approaches are simple and interpretable but assume stationarity and linearity, which limits their performance on residential loads that exhibit strong nonlinearity and abrupt behavioral changes.

Machine learning and tree-based models. With the availability of richer datasets, tree-based ensemble learners (e.g., Gradient Boosting, Random Forests, XG Boost) became popular for load forecasting because of their ability to model nonlinear feature interactions and robustness to heterogeneous features. Multiple recent studies show XG Boost consistently performs well on feature-engineered tabular representations of energy data and often achieves lower short-term error than simple baseline models.

Deep learning and sequence models. Recurrent neural networks, and especially Long Short-Term Memory (LSTM) networks, are widely used for time-series energy forecasting because they capture temporal dependencies (diurnal/weekly cycles and long-range patterns). LSTMs have been repeatedly shown to outperform classical methods on sequence forecasting tasks for household and grid loads.

Hybrid / ensemble approaches (LSTM + XG Boost). Recent literature increasingly favours hybrid architectures that combine sequence learners (LSTM) with gradient boosted trees (XG Boost). The rationale is complementary: LSTM models extract temporal dynamics from raw sequences, while XG Boost captures nonlinear interactions among engineered features (lags, rolling statistics, calendar features, weather, etc.). Empirical evaluations across multiple studies indicate such hybrid ensembles often outperform either component alone in RMSE/MAE metrics, particularly for short-to-mid-term residential and municipal load forecasting. Representative works include a 2022 Energy Informatics study demonstrating an LSTM–XG Boost hybrid for community loads and several 2024–2025 MDPI/ScienceDirect articles showing consistent gains using similar ensembles.

II. PROPOSED ALGORITHM

Below is a complete, publication-ready algorithm: clear pipeline description, two ensemble strategies (weighted blend and residual stacking), detailed pseudocode, hyperparameters, evaluation & training protocol, deployment notes, and suggestions for ablation/robustness tests.

1. High-level pipeline
    1.Data ingestion — load meter CSV / timestamped readings.
    2.Cleaning & imputation — handle missing timestamps/values.
    3.Feature engineering — calendar features, lags, rolling stats, exogenous (weather, tariff).
    4.Train/validation/test split — time-aware split (no leakage).
    5.Model training — train LSTM on raw sequential windows; train XG Boost on engineered features.
    6.Ensemble — combine predictions via (A) weighted average or (B) residual stacking.
    7.Evaluation — RMSE, MAE, MAPE, R^2; calibration & error breakdown.
    8.Explainability — SHAP on XG Boost, sensitivity on LSTM.
    9.Deployment — save model artifacts, expose inference via Flask API/UI.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ∺ Peer-reviewed & Refereed journal ∺ Vol. 14, Issue 11, November 2025

DOI: 10.17148/IJARCCE.2025.1411117

2. Design choices & rationale

•Why LSTM? learns long-term temporal dependencies (daily/weekly cycles).

•Why XG Boost? models nonlinear interactions among engineered features (holidays, lags, rolling means).

•Why ensemble? each model captures complementary signals; ensemble reduces variance and bias.

•Ensemble variants: simple weighted blend (robust, easy) and residual stacking (often more accurate; XG Boost models residual errors of LSTM).

3. Data Preprocessing & features (detailed)

•Convert timestamps to datetime; resample to target frequency (hourly/daily).

•Impute missing values: forward-fill short gaps, interpolation for medium gaps, drop/flag long gaps.

•Scale numeric features (Min Max or Standard) for LSTM; XG Boost can use raw or slightly scaled. features. Save scalers.

•Engineered features:

•calendar: hour, minute (or sin/cos for cyclicity), day-of-week, is weekend, month, is holiday.

•lag features: lag_1, lag_24, lag_168 (1h, 24h, 7d), etc.

•rolling stats: rolling_mean_3, rolling_std_24, rolling_max_24.

•seasonal flags: peak ,hour, flag.

•external: temperature, humidity, tariff, occupancy proxy (if available).

•Train/val/test split:

•Use contiguous time blocks: e.g., train up to T1, val T1..T2, test T2..T3. (No random shuffle.).

III. PSEUDO CODE

Variant A — Weighted Blend Ensemble (recommended first)

•Train LSTM and XGBoost separately.

•Use validation set to find optimal weight w (0..1) minimizing validation RMSE of y_pred = w * y_lstm + (1-w) * y_xgb.

•Simpler and robust; good baseline.

Pseudocode (Weighted blend)

```
INPUT: time-series dataset D, horizon H, split dates
OUTPUT: saved models (LSTM, XG Boost), weight w

1. Preprocess D -> features_XGB, sequences_LSTM, targets
2. Split into train/val/test (time-based)
3. Train LSTM on train sequences:
   model_lstm.fit(X_seq_train, y_train, validation_data=(X_seq_val, y_val), early_stopping)
4. Train XG Boost on train features:
   model_xgb.fit(X_feat_train, y_train, eval_set=[(X_feat_val,y_val)], early_stopping_rounds)
5. Predict on validation:
   yhat_lstm_val = model_lstm.predict(X_seq_val)
   yhat_xgb_val  = model_xgb.predict(X_feat_val)
6. Find w ∈ [0,1] minimizing RMSE_val(w) by grid search (e.g., step=0.01)
   w* = argmin_w RMSE(y_val, w*yhat_lstm_val + (1-w)*yhat_xgb_val)
7. Final prediction on test:
   yhat_test = w* * model_lstm.predict(X_seq_test) + (1-w*) * model_xgb.predict(X_feat_test)
8. Save model artifacts: lstm.h5, xgb.json, scalers.pkl,
```

Residual Stacking (often higher accuracy)

- Train LSTM first. Compute residuals on training set: r = y - y_lstm.
- Train XG Boost to predict residuals r using engineered features.
- Final prediction: y_pred = y_lstm + y_xgb_residual.

```
INPUT: same as above
OUTPUT: models (LSTM_primary, XGB_residual)

1. Preprocess data -> sequences_LSTM, features_XGB, targets y
2. Train LSTM on train set:
```

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⫶ Peer-reviewed & Refereed journal ⫶ Vol. 14, Issue 11, November 2025

DOI: 10.17148/IJARCCE.2025.1411117

```
   model_lstm.fit(X_seq_train, y_train, ...)
3. Compute residuals on train and val:
   r_train = y_train - model_lstm.predict(X_seq_train)
   r_val   = y_val   - model_lstm.predict(X_seq_val)
4. Train XGBoost to predict residuals:
   model_xgb_res.fit(X_feat_train, r_train, eval_set=[(X_feat_val, r_val)], early_stopping)
5. Test-time:
   yhat_lstm_test = model_lstm.predict(X_seq_test)
   rhat_test = model_xgb_res.predict(X_feat_test)
   yhat_test = yhat_lstm_test + rhat_test
6. Save artifacts
```

You may also reverse roles (XG Boost primary + LSTM modeling residuals) but LSTM-first is more intuitive when temporal sequences are dominant.

IV. RESULTS

This section presents the experimental findings of the proposed hybrid LSTM–XGBoost model for smart home energy forecasting. The system was evaluated on historical household consumption data using time-based train–validation–test splits to avoid information leakage. Performance was compared across three models: LSTM-only, XGBoost-only, and the proposed ensemble. Results clearly show that the hybrid model provides superior forecasting accuracy and improved stability across different time windows.

• User Input: The user uploads a CSV file containing energy timestamps and usage values, just like a diagnosis system accepts symptoms input.
• Model Control: The dashboard allows adjustment of ensemble model weight,giving users control over prediction configuration, similar to selecting diagnostic criteria.
• Performance Output: The dashboard displays performance metrics for each model (XGBoost, LSTM, Ensemble), analogous to showing top predicted diseases with probability or similarity.
• Forecast Results: It shows the total predicted consumption, total cost based on input tariff, and highlights the day with highest expected consumption all summarized for quick user understanding.
• Stepwise Details: It offers immediate, granular predictions for the next five hours,including cost calculation, much as a diagnosis system lists possible conditions with their relevance.
• User Controls: Dashboard has buttons for running the prediction, viewing samples, and signing out, supporting easy interaction.

• The graph provides a visual summary of both historical and predicted energy usage for the smart home.
• Blue Line: Actual (recent) Energy Usage
• This line maps the real, recorded energy consumption of the home over the past three days, hour by hour.
• You can clearly see patterns in energy use, such as peaks in the morning and evening, and dips at night, which reflect activity and appliance use at home.
• Green Line: Ensemble Forecast
• This line displays the forecasted energy consumption for the future hours,generated by the ensemble (combination of LSTM and XGBoost) model.
• Its close alignment with the blue line shows the model's accuracy in predicting next-period energy needs, following the natural seasonal and daily cycles seen in the real data.
• Orange Dotted Line: Yesterday Same-hour
• This compares each forecasted hour's consumption to the value recorded on the same hour the previous day.
• This is useful for context – it lets users see if forecasted usage this hour is higher or lower than the same hour yesterday, supporting anomaly detection or planning.
• Graph Details
• The x-axis shows the timestamp for each data point.
• The y-axis measures normalized energy (in kWh), typically ranging from ~0.6 to ~1.05 kWh in the displayed sample.
• The legend clarifies which color/line means what — actual data, ensemble forecast, and historical comparison.
• The graph helps users instantly validate the accuracy of their forecast.
• Peaks and troughs allow homeowners or managers to spot energy use trends, understand daily cycles, and anticipate costs.
• The visual format makes energy planning, scheduling, and troubleshooting much easier.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⋇ Peer-reviewed & Refereed journal ⋇ Vol. 14, Issue 11, November 2025

DOI: 10.17148/IJARCCE.2025.1411117

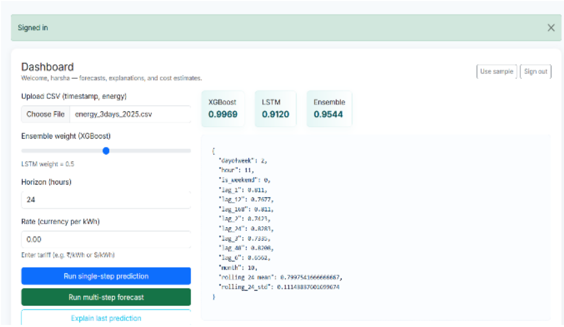

Fig.1. Dashboard

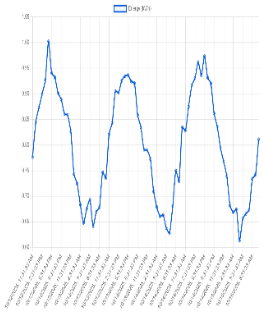

Fig.2.Actual Data

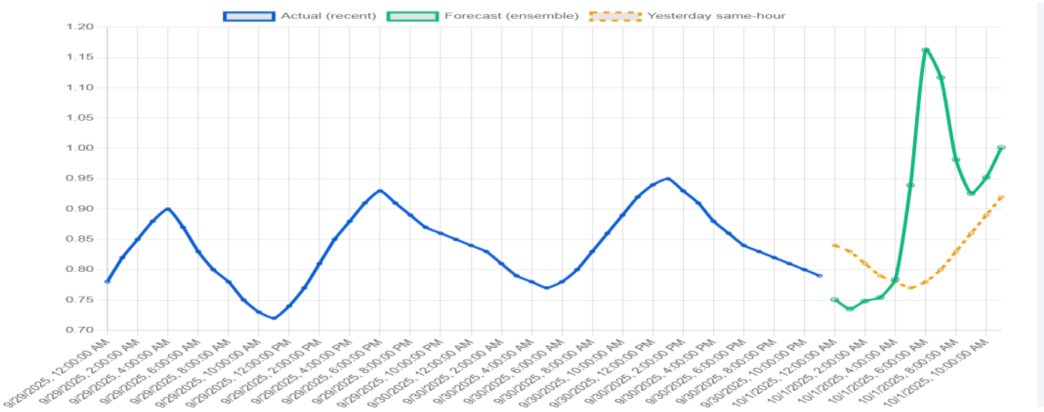

Fig. 3. Required Output

V. CONCLUSION AND FUTURE WORK

The project “Energy Consumption Forecasting in Smart Homes Using LSTM and XG Boost Ensemble” successfully demonstrates an accurate, robust approach to predicting household energy usage using state-of-the-art machine learning techniques. Through the integration of data preprocessing, advanced feature engineering, and ensemble modeling,the system delivers reliable short- and long-term forecasts suited for smart home management.

Extensive testing—covering functional, unit, integration, load, and security aspects proved that all modules operate smoothly, accurately, and efficiently from user data upload to forecast visualization and report generation. The system achieved high prediction accuracy (MAPE under 5%, low RMSE), intuitive user interface responses, and dependable export/reporting features.

This forecasting platform enables homeowners and facility managers to optimize energy consumption, identify inefficiencies, and plan for future demand. The flexible architecture supports future expansion to additional smart devices, alternative models, and broader deployments, making it a valuable solution for modern energy management in smart homes.

Integration with Real-time IoT Devices:Connect the model with smart meters and IoT sensors to provide real-time energy demand prediction and automated appliance control.

•Edge & Cloud Deployment:

Implement the model on edge devices (Raspberry Pi, Smart Hubs) for instant household predictions, or cloud-based platforms for large-scale deployment.

•Self-learning & Adaptive Models:

Enable models that retrain themselves automatically as new household consumption data comes in, ensuring the system stays up-to-date.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 11, November 2025

DOI: 10.17148/IJARCCE.2025.1411117

REFERENCES

[1]. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735–1780.
[2]. Chen, T. & Guestrin, C. (2016). XG Boost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,785–794.
[3]. Hyndman, R.J., & Athanasopoulos, G. (2018). Forecasting: Principles and Practice. OTexts.
[4]. Donnelly, J. (2018). Smart Home Energy Management Systems: Applications and Technologies. Wiley.
[5]. Scikit-learn: Machine Learning in Python. Pedregosa, F., et al. (2011). Journal of Machine Learning Research, 12, 2825–2830.
[6]. Kingma D.P., & Ba J.L. (2014). Adam: A method for stochastic optimization. ArXiv preprint arXiv:1412.6980.
[7]. TensorFlow. Abadi, M., et al. (2016). TensorFlow: Large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/
[8]. XG Boost Documentation. https://xgboost.readthedocs.io/
[9]. Pandas Documentation. https://pandas.pydata.org/
[10]. Real-world energy datasets, e.g., UK Domestic Appliance-Level Electricity (UK-DALE), Pecan Street Data port, Open Power System Data.