Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Electricity Consumption Forecasting and Bill Estimation Using Machine Learning: A Full-Stack XGBoost-Driven Web Application for TNEB Residential Tariff Prediction

Janani K • Nivethitha M • Vaishnavi S
Supervised by: Dr. C.M.T. Karthigeyan, M.Tech., Ph.D.
Department of Computer Science and Engineering
Government College of Engineering Bargur, Krishnagiri - 635 104, Tamil Nadu, India|April 2026

Abstract - Residential electricity management in India is hindered by the opacity of multi-slab tariff structures and the absence of accessible, predictive consumer tools. This paper presents an intelligent, full-stack web application — the Electricity Consumption Predictor — designed to forecast bi-monthly electricity usage and estimate TNEB-compliant utility bills for residential households in Tamil Nadu. The system integrates four principal modules: (1) a household appliance profiling interface capturing granular usage parameters; (2) an XGBoost regression engine trained on synthesized domestic consumption data to predict electricity units (kWh); (3) a deterministic TNEB tariff calculator mapping predicted units through the revised 2024 multi-slab pricing model; and (4) a dynamic budget analysis and personalised energy-insights dashboard. Built on a React.js frontend communicating with a FastAPI/Python backend via RESTful JSON endpoints, the platform delivers predictions with an RMSE of 8.1 units under baseline conditions and an R^2 score of up to 0.965 for typical residential profiles. Response latency remains below 160 ms under full computational load. Experimental evaluation across five household profiles and four seasonal conditions confirms that the system substantially outperforms generic online calculators and static TNEB portals in predictive granularity, tariff accuracy, and actionable user feedback. The open-architecture design supports future extension to IoT smart-meter integration, multi-regional tariff databases, and solar energy offset modelling.

Keywords: ***Electricity Consumption Prediction; XGBoost Regression; TNEB Tariff Estimation; FastAPI; React.js; Residential Energy Management; Machine Learning; Smart Home Analytics; Budget Forecasting; Energy Dashboard***

1. INTRODUCTION

Electricity is the foundational resource of modern civilization, driving domestic appliances, industrial processes, and critical public infrastructure. In recent years, residential energy demand in India has escalated dramatically, fuelled by rising household incomes, the proliferation of air-conditioning units, and an expanding base of energy-intensive electronics. According to the Central Electricity Authority of India (2023), domestic consumption now constitutes nearly 25% of national electricity generation — a share projected to grow steadily through the decade. Despite this surge, most residential consumers lack actionable insight into the relationship between their daily appliance usage and the resultant utility bills, particularly under complex multi-slab pricing regimes such as those enforced by the Tamil Nadu Electricity Board (TNEB).

The TNEB Domestic Tariff IA structure applies a progressive, bi-monthly multi-slab rate that penalises high-consumption households exponentially. Households consuming up to 100 units pay nothing (zero-tariff benefit), while successive slabs up to and beyond 500 units attract sharply escalating rates. This non-linearity renders simple arithmetic estimates systematically inaccurate and leaves consumers unprepared for the final bill. Existing tools — ranging from generic online wattage calculators to the TNEB self-service portal — are fundamentally retrospective: they either compute a past bill or apply a flat per-unit rate that entirely ignores slab transitions, fixed charges, and seasonal variation.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

This paper addresses the resulting gap by presenting the Electricity Consumption Predictor, a full-stack web application that leverages supervised machine learning and structured tariff logic to provide proactive, personalised electricity bill forecasts. The system enables users to define a detailed household appliance profile — including device count, rated wattage, daily usage hours, and seasonal context — and receive an immediate, tariff-accurate monetary estimate alongside interactive visualisations of budget adherence and tailored energy-saving recommendations.

1.1 Problem Statement

The core challenge addressed by this work is threefold. First, existing tools apply a single flat rate that fails to replicate the exponential cost escalation inherent in TNEB's tier-based structure, producing estimates that may diverge from actual bills by 20–40% for high-consumption households. Second, current tools operate as stateless single-query calculators with no capacity for budget tracking, profile storage, or progress monitoring. Third, no freely accessible, appliance-granular tool exists that simultaneously forecasts consumption, computes a slab-accurate bill, and generates personalised load-reduction insights within a unified, interactive interface accessible without hardware installation.

1.2 Objectives

- To design and implement an XGBoost regression model capable of predicting bi-monthly residential electricity consumption (kWh) from a structured household appliance profile.
- To integrate a deterministic TNEB Domestic Tariff IA calculator that maps predicted units through current multi-slab pricing rules to produce accurate monetary bill estimates.
- To develop a responsive, full-stack web application (React.js + FastAPI) that abstracts the prediction pipeline into an intuitive consumer-facing interface.
- To implement an interactive Budget Analysis module enabling users to define financial limits and receive real-time visual feedback on projected cost versus budget.
- To generate personalised, appliance-level energy-saving recommendations derived from the user's own consumption profile.

1.3 Principal Contributions

- A validated XGBoost regression pipeline trained on synthesised domestic household data achieving RMSE $\leq$ 8.1 units and $R^2 \geq 0.965$ under baseline residential conditions.
- A modular TNEB tariff engine that accurately handles all 2024 slab thresholds, fixed charges, and zero-tariff exemptions for the first 100 units.
- A complete open-architecture full-stack web application delivering sub-160 ms end-to-end API response latency and supporting concurrent multi-user access.
- A comparative evaluation demonstrating measurable superiority over four existing tool categories across predictive accuracy, tariff fidelity, and interface utility.

2. BACKGROUND AND RELATED WORK

A systematic review of existing literature was conducted spanning household energy prediction, regional tariff modelling, non-intrusive load monitoring, and behaviour-change dashboard design. The following subsections survey the most relevant prior contributions and identify the gaps addressed by the proposed system.

2.1 Tariff-Based Cost Optimisation in Smart Homes

Sharma and Gupta [1] presented a linear programming framework to minimise household electricity costs against Time-of-Use (ToU) and tier-based tariff schedules. While mathematically rigorous, the model operated as a static backend optimiser lacking any user-facing interactive interface. Budget tracking, seasonal sensitivity, and appliance-level profiling were entirely absent — limitations directly addressed by the proposed Budget Analysis and appliance intake modules.

2.2 Rule-Based Expert Systems for Energy Auditing

Thompson et al. [2] introduced a digital energy-auditing expert system driven by static heuristic rules mapped to survey questionnaire responses. Although accessible to non-technical users, the rigid rule base made regional tariff updates and appliance

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

catalogue extensions prohibitively complex. The architecture provided no dynamic recalculation on input change, a shortcoming resolved by the React-driven real-time frontend of the proposed system.

2.3 Regional Tariff Calculation Engines

Krishnan and Iyer [3] engineered a deterministic algorithm that accurately computes Indian regional electricity board tariffs, handling edge cases such as connected-load penalties and fixed charges. The proposed system directly adapts the mathematical rigour of this approach, integrating it into an accessible web interface rather than a command-line utility.

2.4 Non-Intrusive Load Monitoring (NILM)

Wang and Zheng [4] established a Hidden Markov Model (HMM)-based framework for disaggregating individual appliance loads from a single metered power signal. While accurate, NILM requires continuous access to high-frequency smart-meter readings unavailable to most Indian residential consumers. The proposed system circumvents this hardware dependency by achieving comparable appliance-level granularity through structured user-input profiling.

2.5 Machine Learning for Residential Energy Prediction

Chen et al. [5] applied Long Short-Term Memory (LSTM) networks to historical smart-grid data, achieving 92% accuracy on macroscopic grid-level load forecasting. Despite high accuracy, LSTM deployment requires GPU infrastructure and historical meter data unavailable to individual consumers. The proposed system employs XGBoost — computationally lightweight at inference time and trainable on synthetic household data — to achieve comparable predictive utility without hardware prerequisites.

2.6 Behavioural Interventions through Energy Dashboards

Davis and Miller [6] demonstrated that users exposed to interactive energy dashboards reduced their consumption by up to 12% over six months through goal-setting and progress visualisation. The proposed system operationalises this finding through its Budget Analysis module and Insights Engine, ensuring that the financial framing — not merely the raw kWh figure — is the primary behavioural lever presented to the user.

2.7 Summary of Prior Work

Table 1: Summary of related work — key contributions and residual limitations

Ref.	Authors (Year)	Technique	Limitation	Addressed by Proposed System
[1]	Sharma & Gupta (2021)	Linear programming optimisation	No interactive UI; no budget tracking	Interactive Budget Analysis module
[2]	Thompson et al. (2021)	Rule-based heuristic expert system	Rigid architecture; hard to update tariffs	Modular FastAPI backend; configurable tariff engine
[3]	Krishnan & Iyer (2021)	Deterministic slab algorithm	Command-line only; no consumer frontend	Full-stack React web application
[4]	Wang & Zheng (2022)	NILM with Hidden Markov Models	Requires inaccessible high-frequency meter data	User-input appliance profiling
[5]	Chen et al. (2023)	LSTM deep learning	GPU required; no manual input interface	CPU-deployable XGBoost; interactive input form
[6]	Davis & Miller (2023)	Persuasive energy dashboards	Generic tips; no appliance-level specificity	Personalised insights engine

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

3. SYSTEM ARCHITECTURE AND DESIGN

The Electricity Consumption Predictor is built on a three-tier client-server architecture that cleanly separates the user-facing presentation layer, the orchestration and computation backend, and the machine learning and tariff data layer. This separation of concerns ensures that the predictive model can be retrained independently of the user interface, and that tariff rules can be updated without modifying the frontend codebase.

3.1 Architectural Overview

- Tier 1 — Presentation Layer (React.js SPA): Renders the appliance configuration form, budget input controls, prediction results, dynamic chart visualisations, and personalised insights. All user interactions trigger asynchronous API calls, enabling real-time feedback without page reload.
- Tier 2 — Application Layer (FastAPI/Python): Exposes a RESTful POST /predict endpoint that validates incoming JSON payloads, invokes the XGBoost inference engine, applies the TNEB tariff calculator, performs budget analysis, and assembles the structured JSON response.
- Tier 3 — Model and Tariff Layer: Contains the pre-trained XGBoost regression model (xgboost_tneb_model.json) and the deterministic TNEB tariff function. The model layer communicates with the application layer through direct Python function calls, not through a network boundary.

3.2 Data Flow

A single prediction request traverses the following processing pipeline: the user submits a completed household profile through the React frontend; the JavaScript client serialises the form data as a JSON object and dispatches a POST request to the FastAPI /predict endpoint; the backend Pydantic model validates all 19 input features and rejects malformed requests with structured HTTP 422 errors; the validated feature dictionary is passed to the XGBoost inference function, which loads the persisted model and returns predicted bi-monthly units; the predicted units are forwarded to the TNEB tariff calculator, which applies slab thresholds and fixed charges to produce an INR estimate; the budget analysis module computes the variance between the predicted bill and the user's declared budget; finally, the insights engine scans the feature vector to identify high-drain appliances and assembles targeted recommendations; the assembled response is returned as JSON to the frontend, which updates the dashboard visualisations via React state updates.

3.3 Input Feature Schema

The prediction model accepts 19 structured input features, summarised in the table below. Features were selected through domain analysis of TNEB consumption patterns and validated against publicly available Tamil Nadu residential energy survey data.

Table 2: Prediction model input feature schema

Feature Name	Data Type	Description
house_size_sqft	Integer	Total conditioned floor area (sq. ft.)
num_occupants	Integer	Number of permanent household residents
num_acs / avg_ac_hours_per_day	Int / Float	Air-conditioner count and average daily runtime (hrs)
num_fans / fan_hours_per_day	Int / Float	Ceiling/table fan count and average daily runtime (hrs)
num_led_bulbs / num_old_bulbs	Integer	Count of LED and incandescent lighting units
num_tvs / tv_hours_per_day	Int / Float	Television count and average daily viewing hours
has_refrigerator	Binary (0/1)	Presence of a refrigerator (assumed always-on)
has_washing_machine / loads_per_week	Binary / Int	Washing machine flag and weekly cycle count

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

has_geyser / geyser_hours_per_day	Binary / Float	Geyser flag and average daily heating duration (hrs)
solar_water_heater	Binary (0/1)	Presence of solar water heating (reduces geyser load)
laptop_hours_per_day	Float	Average daily laptop/desktop usage hours
has_ev	Binary (0/1)	Presence of an electric vehicle charging at home
season	Categorical (0/1/2)	Seasonal context: Winter=0, Summer=1, Monsoon=2

4. MACHINE LEARNING MODEL

4.1 Algorithm Selection: XGBoost Regression

XGBoost (eXtreme Gradient Boosting) was selected as the predictive engine following a comparative evaluation against linear regression, Random Forest, and LSTM networks. XGBoost's ensemble of gradient-boosted decision trees handles the non-linear, interaction-heavy structure of residential electricity consumption without requiring feature normalisation, tolerates missing values natively, and achieves state-of-the-art performance on tabular prediction tasks at CPU-level inference speed. Critically for deployment, the trained model is serialised to a compact JSON file (< 2 MB) that loads in under 50 ms, ensuring the sub-160 ms API response latency target is met consistently.

4.2 Training Configuration

The model was trained on a synthesised dataset of 5,000 household profiles generated to reflect the statistical distribution of Tamil Nadu residential energy consumption patterns. An 80/20 train-test split was applied with a fixed random seed (42) to ensure reproducibility. Key hyperparameters were set as follows: objective = reg:squarederror; eval_metric = rmse; max_depth = 4; learning_rate = 0.1; subsample = 0.8; colsample_bytree = 0.8; num_boost_round = 150 with early stopping at 10 non-improving rounds. The shallow tree depth (max_depth=4) was deliberately chosen to prevent overfitting to individual synthetic household patterns, ensuring generalisability to real-world inputs.

4.3 TNEB Tariff Calculation Engine

The tariff module implements the 2024 TNEB Domestic Tariff IA structure deterministically. The first 100 bi-monthly units are provided free of charge (zero tariff). Units 101–200 are billed at ₹1.50/unit; 201–500 at ₹3.00/unit; 501–1000 at ₹5.00/unit; and units beyond 1000 at ₹7.00/unit. Fixed charges, meter rental, and applicable surcharges are applied as additive constants dependent on the consumption slab. This deterministic function accepts the XGBoost-predicted unit count and returns an INR estimate that replicates the actual TNEB billing computation, including the slab-crossing recalculation that generic calculators routinely omit.

5. SYSTEM MODULES AND IMPLEMENTATION

The application is structured into six cohesive functional modules, each encapsulating a distinct stage of the data and user experience pipeline.

5.1 Data Aggregation and Collection Module

This module renders the household appliance configuration form within the React frontend. It captures 19 structured input features through validated form controls — including numeric spinners for device counts and daily hours, binary toggles for optional appliances, and a seasonal dropdown. Input validation is enforced client-side (range checks, required fields) before submission to prevent redundant API calls for malformed data.

5.2 Data Pre-Processing and Feature Engineering Module

Upon receipt of the JSON payload, the FastAPI backend performs server-side validation through Pydantic model binding, which enforces strict type constraints and raises structured HTTP 422 errors on any schema violation. Derived features — such as effective

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

geyser load (reduced by solar water heater presence) and normalised appliance energy estimates — are computed within the feature engineering pipeline before the feature vector is passed to the model.

5.3 Machine Learning Model Training Module

The XGBoost model was trained offline using the Python xgboost library and persisted to disk as a JSON artefact. The training pipeline reads the synthetic household dataset, performs the train-test split, configures hyperparameters, trains with early stopping on the validation RMSE, and saves the resulting model. This module is executed as a standalone script; the inference path in production loads the persisted model file and performs no retraining at runtime.

5.4 Prediction and Tariff Calculation Engine Module

The core prediction pipeline constructs an XGBoost DMatrix from the validated feature dictionary, calls the loaded Booster's predict method to obtain bi-monthly kWh estimates, and immediately forwards the result to the TNEB tariff function. The combined output — predicted units and estimated INR bill — is returned as a structured JSON object to the application layer for downstream processing.

5.5 Interactive Budget Tracking Module

The frontend Budget Analysis panel accepts a user-defined bi-monthly financial target and computes the variance between the predicted bill and that target in real time. A dynamic bar chart (rendered via a JavaScript charting library) visually encodes the comparison, applying colour-coded thresholds: green for under-budget, amber for within 10% of budget, and red for projected overspend. This immediate visual feedback is intended to motivate proactive behavioural adjustment before the billing cycle closes.

5.6 Personalised Insights and Dashboard Module

The insights engine performs a rule-based scan of the feature vector to identify the three highest-consuming appliance categories based on wattage estimates and reported daily hours. For each identified high-drain device, a contextually specific recommendation is generated — for example, recommending ISEER-5-rated inverter ACs for households reporting more than four hours of daily air-conditioning usage. The dashboard aggregates predicted units, INR estimate, budget variance, and insights into a single-screen display, providing the user with a complete analytical picture without requiring navigation between views.

6. PERFORMANCE EVALUATION

6.1 Evaluation Metrics

The system was evaluated using five complementary metrics spanning predictive accuracy, model fit, classification correctness, and engineering performance. Root Mean Squared Error (RMSE) serves as the primary accuracy indicator, penalising large deviations proportionally and providing a unit-consistent error measure in kWh. Mean Absolute Error (MAE) offers a linear error measure directly interpretable as the average unit-count discrepancy between prediction and ground truth. The R^2 (coefficient of determination) score quantifies the proportion of variance in electricity consumption explained by the feature set. Budget Accuracy measures the percentage of predictions correctly classified as over- or under-budget relative to a declared target. Response Latency captures end-to-end API processing time in milliseconds.

Table 3: Evaluation metrics — definitions and formulas

Metric	Formula	Interpretation
RMSE	$\sqrt{[\Sigma(y_i - \hat{y}_i)^2 / N]}$	Std. deviation of prediction errors (kWh). Lower is better.
MAE	$\Sigma\lvert y_i - \hat{y}_i\rvert / N$	Average absolute unit error. Lower is better.
R^2 Score	$1 - [\Sigma(y_i - \hat{y}_i)^2 / \Sigma(y_i - \bar{y})^2]$	Variance explained. Closer to 1.0 is better.
Budget Accuracy	Correct classifications / Total × 100	% of over/under-budget flags correctly assigned. Higher is better.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

Response Latency	Request End Time − Request Start Time	End-to-end API processing time (ms). Lower is better.

6.2 Experimental Results — Seasonal Variation

The test set comprised 20% of the synthesised dataset (approximately 1,000 profiles), stratified across seasonal conditions, household size categories, and appliance complexity tiers. Results are presented by seasonal segment in Table 4.

Table 4: Prediction accuracy by seasonal condition

Dataset Segment	RMSE (units)	MAE (units)	Latency (ms)	Remarks
Clean Baseline (Average Household)	8.1	5.4	142	Best performance — stable base load
Winter Season	9.2	6.8	155	Near-ideal; minimal AC contribution
Monsoon Season	12.4	8.2	148	Acceptable; moderate unpredictability
Summer Season (High AC Usage)	16.8	12.5	160	Highest variance; AC behavioural noise

6.3 Experimental Results — Household Profile Variation

Table 5: Predictive accuracy by household profile category

Household Profile	True Units (Avg)	Predicted Units (Avg)	R^2 Score	MAE (units)
Small Apartment (< 1,000 sq. ft.)	210.5	214.2	0.965	4.8
Mid-Size Home (1,000–2,500 sq. ft.)	480.2	473.5	0.932	11.2
Luxury Home (> 2,500 sq. ft.)	840.6	858.9	0.884	22.4
High Appliance Load (No AC)	320.1	315.6	0.941	6.1
Heavy AC Usage (> 8 hrs/day)	750.8	768.4	0.852	26.8

The results demonstrate consistent model performance across the profile spectrum. Small-apartment profiles yield the highest fidelity ($R^2 = 0.965$, MAE = 4.8 units), attributable to the predictable, schedule-driven operation of baseline appliances such as refrigerators and LED lighting. Performance degrades gracefully as household complexity increases: luxury home profiles introduce multi-zone AC units and premium appliances whose stochastic usage patterns are harder to model from single-day averages, producing an MAE of 22.4 units — still within practical utility for household budgeting purposes. Heavy AC usage remains the most challenging segment, with MAE rising to 26.8 units under sustained high-load summer conditions, consistent with findings in the LSTM-based forecasting literature.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

6.4 Comparison with Existing Systems

Table 6: Feature and performance comparison with existing tool categories

Feature / Capability	TNEB Portal	Generic Calculator	Smart Meter	Proposed System
Predicts future bills (proactive)	No	Partial	No	**Yes**
Appliance-level granularity	None	Limited	None	**Yes (19 features)**
Accurate multi-slab tariff calculation	Yes (past bills)	No (flat rate)	N/A	**Yes (2024 TNEB slabs)**
Budget tracking and visualisation	No	No	No	**Yes**
Personalised energy-saving insights	No	No	No	**Yes**
R^2 / Accuracy metric	N/A	Variable	100% (hardware read)	**$R^2 \approx 0.94$ (avg)**
Deployment requirement	Web browser	Desktop app	IoT hardware + hub	**Pure Web SPA (zero install)**
Open REST API	No	No	Paid subscription	**Yes (FastAPI)**

7. DISCUSSION

7.1 Interpretation of Results

The XGBoost regression model's strong baseline performance (RMSE = 8.1 units, $R^2 = 0.965$) confirms that a structured 19-feature appliance profile carries sufficient information to predict bi-monthly residential electricity consumption with engineering-grade accuracy for the majority of household types in the Tamil Nadu context. The model's accuracy degradation under summer high-AC-usage conditions is expected and consistent with prior machine learning energy forecasting studies: stochastic human behaviour around cooling comfort — varying thermostat settings, intermittent usage, and occupancy changes — introduces variance that cannot be fully captured by a static daily-average feature. This limitation is partially mitigated by the seasonal flag feature, but would benefit from dynamic real-time sensor input in future iterations.

7.2 Limitations

Several limitations bound the present work. First, the training dataset is synthesised rather than empirically collected, meaning the model's statistical distribution may not perfectly reflect the full heterogeneity of real Tamil Nadu residential consumption. Ground-truth validation against actual TNEB meter readings is planned as a near-term priority. Second, the current system captures a static household profile snapshot rather than continuously updated usage data; real-world consumption exhibits significant intra-day and intra-week variation that a single daily-average entry cannot fully represent. Third, the TNEB tariff engine implements the 2024 slab structure and will require periodic updates as the Board revises pricing. The modular architecture is designed to accommodate such updates with minimal code disruption.

7.3 Ethical and Equity Considerations

The system collects no personally identifiable information: household profiles are processed transiently and are not persisted server-side unless explicitly saved by the user in a future profile-management feature. No demographic data is collected or used in the prediction pipeline, eliminating the risk of encoding socioeconomic biases into the tariff estimates. The application is designed to run as a zero-cost, browser-accessible tool with no licensing requirement, ensuring equitable access for consumers at all income levels.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

8. CONCLUSION AND FUTURE WORK

8.1 Conclusion

This paper presents the Electricity Consumption Predictor, a full-stack, machine-learning-driven web application that provides Tamil Nadu residential consumers with accurate, proactive electricity bill forecasts based on a detailed appliance usage profile. The system integrates an XGBoost regression engine, a deterministic TNEB 2024 multi-slab tariff calculator, an interactive budget analysis module, and a personalised energy-insights generator into a unified, zero-installation browser interface.

Experimental evaluation across five household profiles and four seasonal conditions confirms that the XGBoost pipeline achieves RMSE $\leq$ 8.1 units and $R^2 \geq 0.965$ under baseline residential conditions, with graceful performance degradation to RMSE = 16.8 units under the most challenging high-AC-usage summer scenario. End-to-end API response latency consistently remains below 160 ms, meeting the interactive-dashboard usability threshold. A structured comparison against four existing tool categories — the TNEB portal, generic online calculators, spreadsheet tools, and commercial smart meters — demonstrates that the proposed system uniquely combines predictive forecasting, slab-accurate tariff computation, budget visualisation, and personalised recommendations in a single freely accessible platform.

These results confirm that accessible, data-driven applications represent an effective mechanism for bridging the information asymmetry between residential energy consumers and complex utility pricing structures, and that machine learning-augmented tools can meaningfully support sustainable household energy management at scale.

8.2 Future Work

- IoT Smart Meter Integration: Direct interfacing with IoT-enabled smart meters to automate real-time consumption data ingestion, eliminating manual profile entry and substantially improving prediction accuracy through continuous learning.
- Historical Pattern Learning: Implementation of a user profile persistence layer with session-over-session ML model fine-tuning to capture individual household behavioural patterns and seasonal trends for progressively more accurate personal forecasts.
- Multi-Regional Tariff Support: Extension of the tariff engine to a dynamically updatable multi-state database, enabling the platform to serve residential consumers across all Indian electricity boards and international tariff structures.
- Solar Energy Offset Modelling: Integration of a solar generation estimator using location-based irradiance data to compute net consumption after residential PV generation, providing eco-conscious users with ROI and payback-period analytics.
- Mobile Application: Development of native iOS and Android applications using React Native, incorporating push notifications for budget-threshold alerts and daily consumption summaries.
- Explainability Visualisations: Integration of SHAP (SHapley Additive exPlanations) value decomposition to overlay feature-importance scores on the prediction output, enabling users to understand precisely which appliances contributed most to their forecasted bill.
- Carbon Footprint Tracking: Extension of the dashboard to convert predicted kWh consumption into estimated CO_2 emissions based on the Tamil Nadu grid's reported generation mix, providing a dual financial-environmental motivation for load reduction.

REFERENCES

[1] Sharma, P., & Gupta, M. (2021). Tariff-based energy cost optimisation for smart homes using linear programming. Energy and Buildings, 245, 111032.
[2] Thompson, E., et al. (2021). A rule-based expert system for residential energy auditing. Energy Policy, 149, 112089.
[3] Krishnan, V., & Iyer, S. (2021). Regional electricity tariff calculation engine for Indian utility boards. International Journal of Electrical Power & Energy Systems, 124, 106382.
[4] Wang, Z., & Zheng, L. (2022). Non-intrusive load monitoring using hidden Markov models for residential appliance disaggregation. IEEE Transactions on Smart Grid, 13(4), 3120–3131.
[5] Chen, Y., et al. (2023). LSTM-based deep learning for residential energy consumption forecasting at the smart-grid level. Applied Energy, 310, 118456.
[6] Davis, S., & Miller, K. (2023). Behavioural impacts of energy information dashboards on domestic consumption: A six-month longitudinal study. Journal of Environmental Psychology, 85, 101934.
[7] Ahmad, T., & Chen, H. (2018). Utility company strategy for short-term energy consumption forecasting using machine learning. Sustainable Cities and Society, 39, 401–417.
[8] Amasyali, K., & El-Gohary, N. M. (2018). A review of data-driven building energy consumption prediction studies. Renewable and Sustainable Energy Reviews, 81, 1192–1205.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

[9] Raza, M. Q., & Khosravi, A. (2015). A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. Renewable and Sustainable Energy Reviews, 50, 1352–1372.

[10] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794.

[11] Deb, C., et al. (2017). A review on time series forecasting techniques for building energy consumption. Renewable and Sustainable Energy Reviews, 74, 902–924.

[12] Zhao, H. X., & Magoulès, F. (2012). A review on the prediction of building energy consumption. Renewable and Sustainable Energy Reviews, 16(6), 3586–3592.

[13] Central Electricity Authority, Government of India. (2023). Annual Report on Growth of Electricity Sector in India 2022–23. New Delhi: Ministry of Power.

[14] Tamil Nadu Electricity Board. (2024). Tariff Order — Domestic Tariff IA: Revised Slab Structure and Fixed Charges. Chennai: TNEB Regulatory Commission.

[15] Wang, Z., & Srinivasan, R. S. (2017). A review of artificial intelligence based building energy use prediction. Renewable and Sustainable Energy Reviews, 75, 796–808.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)