International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 11 | Nov - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Electricity Load and Price Forecasting Using Deep Learning

Sharath Kumar S R
[Akanksha B U] | [Kalpitha S] | [Lavanya M G] | [Sinchana R T]
Department of Information Science and Engineering, JNN College of Engineering, Shivamogga, India

ABSTRACT

Accurate forecasting of electricity load and electricity price supports grid operation, market bidding, demand response and household energy management, yet both series are volatile, non-linear and strongly seasonal. This paper presents the project implementation of a deep-learning system, Smart Energy Forecasting, that predicts electricity load with a Long Short-Term Memory (LSTM) network and electricity market price with a Gated Recurrent Unit (GRU) network. Each model takes a 24-step sequence of 17 input features covering calendar, weather, economic, generation-capacity, lagged load and price, and rolling-average variables. Four recurrent architectures (LSTM, GRU, BiLSTM and CNN-LSTM) were compared on the test set. LSTM gave the best load forecast, with MAE 39.6754, RMSE 48.7313 and RZ 0.8882, while GRU gave the best price forecast, with MAE 2.6210, RMSE 3.3600 and RZ 0.6548. The trained models are deployed in a Streamlit dashboard that provides single-point prediction, 24-hour forecasts, peak-demand alerts, appliance-shifting recommendations and a slab-based electricity bill generator with PDF output. The evaluated dataset is [electricity load data.csv].

Index Terms— electricity load forecasting, electricity price forecasting, deep learning, LSTM, GRU, time series, smart grid, Streamlit.

I. INTRODUCTION

Electricity is difficult to store economically and must be balanced with demand at every instant. Deregulated markets have therefore turned both demand (load) and price into quantities that producers, retailers, grid operators and large consumers must predict ahead of time [1]. Load forecasts support dispatch, reliability and infrastructure planning, while price forecasts support bidding strategies, bilateral contracts and risk control [2], [6]. Both series are affected by weather, calendar effects, fuel prices, generation outages, congestion and market behaviour, which makes them highly volatile and non-linear [1].

Classical statistical models such as ARIMA and its seasonal variants capture trend and seasonality but struggle with non-linearity and price spikes [13], [15]. Machine-learning models such as Random Forest and gradient boosting improve on them [15], [16], and deep neural networks, in particular recurrent networks, convolutional networks and attention-based models, have recently become the dominant approach because they learn temporal features directly from raw sequences [1], [3], [6].

This project develops a deep-learning forecasting system for electricity load and price, named Smart Energy Forecasting. The final implementation trains LSTM, GRU, BiLSTM and CNN-LSTM models with TensorFlow/Keras on [electricity load LSTM for load and GRU for price according to test error, saves the selected models together with their input and output scalers, and serves them through a Streamlit dashboard.

The main contributions of this project are as follows. First, load and price are both forecast, with a separate best model chosen for each target after comparing four recurrent architectures under the same data and metrics [16]. Second, each model uses 17 input features that combine calendar, weather, economic, generation-capacity and lagged load and price information, which the literature identifies as themain drivers of demand and price [1], [6], [12]. Third, the forecasts are turned into practical outputs through a dashboard with peak-demand alerts, appliance-shifting advice and an estimated electricity bill, in the spirit of demand-response work [20]. Finally, the models are evaluated with MAE, RMSE and RZ, and their strengths and limitations are discussed.

The rest of the paper is organized as follows. Section II reviews related work. Section III describes the proposed methodology, Section IV presents the results and the dashboard, Section V discusses them, Section VI concludes the paper, and Section VII outlines its research significance and practical considerations.

II. RELATED WORK

The project literature survey covers 20 recent studies on electricity load and price forecasting. A comprehensive survey by Mustafa *et al.* reviews machine-learning and deep-learning methods for both tasks, lists the datasets used and compares performance measures, and identifies volatility, non-stationarity and non-linearity as the central challenges [1].

Recurrent models. LSTM networks are the most common deep baseline. Gaber and Ibrahim forecast hourly ERCOT settlement prices with an Adam-trained LSTM using load and temperature as inputs [5]. Mohamed *et al.* tune an LSTM with Bayesian optimization (BOL-LPP) for day-ahead load-price forecasting in ERCOT and report a 32% MAE reduction over the strongest deep baseline, a BiLSTM [4]. Qureshi *et al.* apply LSTM to hospital electricity consumption and examine several optimizers [19], and Khan *et al.* use an LSTM day-ahead load forecast inside a multi-agent reinforcement-learning scheme for demand response [20]. For smart-meter data, Fekri *et al.* propose an online adaptive RNN that keeps learning from new samples and handles concept drift [17].

Attention, decomposition and hybrid models. Azam and Younis combine BiLSTM, multi-head self-attention and ensemble empirical mode decomposition to forecast day-ahead load and locational marginal price (LMP) with interpretability analysis [6]. Zhang *et al.* build a four-module framework with isolation-forest and Lasso preprocessing, a DBN-LSTM-CNN point predictor, error compensation and quantile-regression intervals on PJM data

[2]. Mir *et al.* propose ABMF-Net, which joins attention-based data selection, an encoder-decoder with masked prediction and a Bayesian neural network, and report a MAPE as low as 1.89% on Australian market data [3]. Hu *et al.* combine FFT denoising with a GWO-optimized CNN-LSTM-Attention model for medium-to-long-term price forecasting [14].

Convolutional, graph and optimized models. Hong *et al.* use a genetic algorithm to design a CNN for 24-hour-ahead LMP forecasting on PJM data across four seasonal subsets [7]. Cheng *et al.* introduce a graph neural network that learns coupling between European price zones from Nord Pool data [8]. Almazroi and Ayub optimize an AlexNet model with the Dwarf Mongoose algorithm for price forecasting on IESO Ontario data [9], and Dash *et al.* explore an adiabatic-quantum-computing forecaster on a synthetic microgrid dataset [10].

Statistical, machine-learning and risk-aware models. Alkawaz *et al.* combine ARD regression with Extra Trees for day-ahead price forecasting on Nord Pool [11]. Farhoumandi *et al.* fuse ML and DL models to forecast non-conforming net loads and hourly LMPs and estimate spike probabilities [12]. Yousefi *et al.* use feature selection with S-ARIMA to predict monthly electricity prices five years ahead [13]. Lin and Feng pair Random Forest, XGBoost and LSTM with value-at-risk measures for hedging in renewable-heavy markets [15]. Fesseha *et al.* benchmark ARIMA, VAR, XGBoost, Random Forest, LSTM and TCN for multi-step industrial load forecasting and find that LSTM minimizes typical error while XGBoost better bounds large errors [16]. Qin *et al.* combine federated learning with differentiable architecture search for building-level load forecasting [18].

Table I summarizes the surveyed work. Recurring limitations are dependence on a single market or dataset, limited comparison under identical conditions, difficulty with price spikes, and little validation on independent data.

TABLE I. SUMMARY OF SELECTED RELATED WORK

Ref.	Target	Method	Data
[2]	Price	IF-Lasso, DBN-LSTM-CNN, quantile regression	PJM
[3]	Price, load	Attention, encoder-decoder, Bayesian NN, MSSA	Australia
[4]	Price	Bayesian-optimized LSTM	ERCOT
[5]	Price	LSTM with Adam	ERCOT
[6]	Price, load	BiLSTM, multi-head attention, EEMD	LMP, system load
[7]	Price	GA-optimized CNN	PJM
[8]	Price	Graph neural network	Nord Pool
[14]	Price	GWO-CNN-LSTM-Attention, FFT	Medium/long term
[15]	Price	RF, XGBoost, LSTM, VaR/CVaR	3 yr hourly
[16]	Load	ARIMA, VAR, XGB, RF, LSTM, TCN	Smelting plant
[17]	Load	Online adaptive RNN	Smart meters
[9]	Price	AlexNet with Dwarf Mongoose optimization	IESO Ontario
[11]	Price	ARD regression with Extra Trees	Nord Pool
[12]	Price, net load	ML and DL fusion, spike probability	LMP data
[13]	Price	Featureselectionwith S-ARIMA	California, monthly
[18]	Load	Federated learningwith DARTS	Building loads
[20]	Load	LSTM inside multi-agent RL	Household loads

Challenges Identified in the Literature

The surveyed work points to several recurring challenges. *Volatility and spikes:* electricity prices change rapidly and can spike, so models that fit average behaviour well may still make large errors at extreme points [1], [12]. *Non-stationarity and concept drift:* consumption patterns change when equipment, tariffs or user behaviour change, which degrades a model trained once on old data [17]. *Multiple time scales:* short-term, medium-term and long-term forecasting need different inputs and models, and long horizons need drivers such as fuel prices [13], [14]. *Data quality and dimensionality:* outliers, missing values and large numbers of candidate features motivate preprocessing, feature selection and denoising [2], [14].

Hybrid and optimized models usually report better accuracy than single models, but they are evaluated on one market or dataset and cannot easily be compared with each other because the data, horizons and metrics differ [1]. Moreover, a single accuracy figure can hide differences in behaviour: Fesseha *et al.* show that the model with the lowest MAE is not always the one with the smallest worst-case error [16]. For these reasons, this project compares several architectures with several metrics instead of reporting one number.

III. Proposed Methodology

A. System Architecture

The proposed pipeline, shown in Fig. 1, starts from historical load, price, weather and calendar data and builds 17 input features per time step. The features are scaled with a fitted input scaler (X_scaler), and consecutive steps are stacked into 24-step input sequences. An LSTM model predicts electricity load and a GRU model predicts electricity price. The scaled outputs are converted back to real units with an output scaler (y_scaler), and the resulting 24-hour load and price forecast is presented in a Streamlit dashboard.

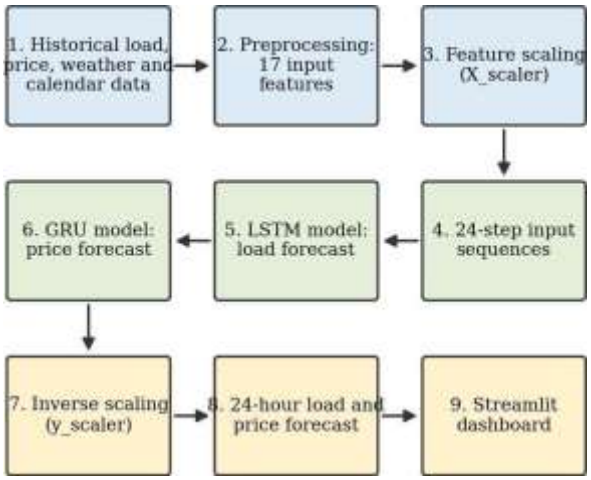


Fig. 1. End-to-end system design of the Smart Energy Forecasting system.

B. Dataset and Input Features

The dataset used in this study is *Electricity Load Data.csv*, obtained from Kaggle. It contains 1,000 hourly records covering January 1, 2024, to February 11, 2024, at 15:00. Each time step is represented by 17 input features, including calendar variables, weather variables, economic and generation-capacity variables, and lagged and rolling load and price values. These features help the models learn temporal patterns in electricity demand and price. The input features are scaled using X_scaler, while the load and price targets are scaled using y_scaler. Both scalers are used consistently during prediction. The dataset was checked for missing values and duplicate rows, and none were found. The records were sorted chronologically and divided into 80% training data and 20% testing data. Additionally, 10% of the training sequences were used for validation during model training.

TABLE II. THE 17 INPUT FEATURES

Group	Features	Count
Calendar	Hour, day of week, month, weekend flag, holiday flag	5
Weather	Temperature, humidity, wind speed, solar radiation	4
Economic / system	GDP index, population index, generation capacity	3
Load history	Previous-day load, previous-week load, rolling average load (24 h)	3
Price history	Previous-day price, rolling average price	2

C. Deep Learning Models

Both forecasting models are recurrent networks. The LSTM, used for load, has input, forget and output gates that let it retain information over long sequences [4], [5], [20]. For input x_t, hidden state h_t and cell state c_t,

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f),\ i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c[h_{t-1}, x_t] + b_c)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o),\ h_t = o_t \odot \tanh(c_t)$$

The GRU, used for price, merges the cell and hidden statesand uses only an update gate z_t and a reset gate r_t, which gives fewer parameters and faster training:

$$z_t = \sigma(W_z[h_{t-1}, x_t]),\ r_t = \sigma(W_r[h_{t-1}, x_t])$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tanh(W[r_t \odot h_{t-1}, x_t])$$

For comparison, a bidirectional LSTM (BiLSTM), which reads the sequence in both directions [6], and a CNN-LSTM, which adds convolutional feature extraction before the recurrent layer [2], were also trained. The load model takes an input of shape 24 × 17, and the network configuration is The load model uses an LSTM architecture with 128 and 64 units in the two LSTM layers, while the price model uses a GRU architecture with 64 and 32 units in the two GRU layers. Both models use a dropout rate of 0.2 to reduce overfitting.

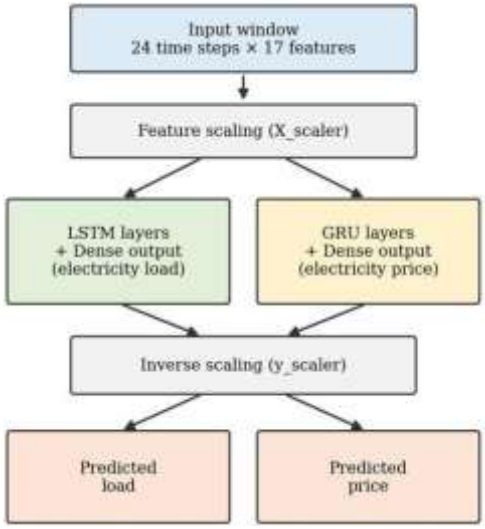


Fig. 2. Model architecture: LSTM for load and GRU for price.

D. Problem Formulation

Let l_t and p_t denotethe electricity load and price at time step t, and let z_t be the 17-element feature vector at that step. The model input is a window of the 24 most recent steps, $X_t = [z_{t-23}, \ldots, z_t]$, and the objective is to learn mappings

$$9l = f_LSTM(X_t; \theta_1),\ p9 = f_GRU(X_t; \theta_2)$$

where θ_1 and θ_2 are the learnable parameters of the two networks. Training adjusts them so that the difference between the predictions and the true values is minimized. Because the input uses only past values and calendar information known in advance, no future measurements leak into the forecast.

E. Training Procedure and Configuration

All four architectures are trained on the same training split and evaluated on the same test split. The selected LSTM and GRU models are saved in Keras format (.h5) and loaded by the dashboard without recompilation, since only inference is needed. Table III lists the configuration.

TABLE III. TRAINING CONFIGURATION

Parameter	Value
Framework	TensorFlow / Keras
Input sequence length	24 time steps
Input features	17
Load model	LSTM
Price model	GRU

Parameter	Value
Layers / units / dropout	LSTM: 128, 64; GRU: 64, 32; dropout: 0.2
Optimizer / learning rate	Adam / 0.001
Batch size / epochs	32 / LSTM 10, GRU 20
Loss function	Mean Squared Error (MSE)

F. Evaluation Metrics

For N test samples with actual values y_i, predictions $\hat{y}_i$ and mean $\bar{y}$, the following metrics are used:

$$MAE = (1/N)\,\Sigma\,|y_i - \hat{y}_i|,\quad RMSE = \sqrt{[(1/N)\,\Sigma\,(y_i - \hat{y}_i)^2]}$$

$$R^2 = 1 - \Sigma\,(y_i - \hat{y}_i)^2 / \Sigma\,(y_i - \bar{y})^2$$

MAE reflects typical error, RMSE penalizes large errors such as price spikes, and RZ shows the share of the variance of the target that the model explains. MAE and RMSE are in the units of the target, so load and price errors cannot be compared with each other directly, whereas RZ is scale-free.

IV. Experimental Results

A. Overall Performance

Table IV compares the four architectures on the test set. For load forecasting, LSTM achieves the lowest error, with MAE 39.6754, RMSE 48.7313 and RZ 0.8882. For price forecasting, GRU achieves the lowest error, with MAE 2.6210, RMSE 3.3600 and RZ 0.6548. These two models are therefore selected for the dashboard.

TABLE IV. TEST-SET COMPARISON OF MODELS

Model	Load MAE	Load RMSE	Load R^2	Price MAE	Price RMSE	Price R^2
LSTM	39.6754	48.7313	0.8882	2.6223	3.3660	0.6536
GRU	59.9807	72.6567	0.7515	2.6210	3.3600	0.6548
BiLSTM	47.2709	58.5462	0.8387	2.6918	3.4616	0.6336
CNN-LSTM	42.4036	54.1858	0.8618	3.0707	3.7820	—

Best valuein eachcolumn: LSTM for load; GRUfor price. The price R^2 of CNN-LSTM was not reported.

B. Load Forecasting Results

Fig. 3 shows the load errors. LSTM reduces the RMSE by about 32.9% compared with GRU (72.6567), 16.8% compared with BiLSTM (58.5462) and 10.1% compared with CNN-LSTM (54.1858). Its RZ of 0.8882 means that the model explains about 89% of the variance of the load. The ranking is LSTM, CNN-LSTM, BiLSTM and GRU, so the added convolutional or bidirectional components did not improve on a plain LSTM for this dataset.

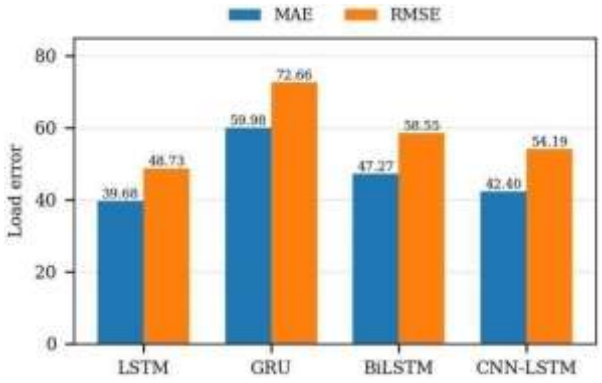


Fig. 3. Load forecasting error (MAE and RMSE) of the four models.

C. Price Forecasting Results

Fig. 4 shows the price errors. The three recurrent models are close: GRU (MAE 2.6210, RMSE 3.3600) and LSTM (2.6223, 3.3660) differ by only 0.0013 in MAE and 0.0060 in RMSE, and BiLSTM is slightly behind (2.6918, 3.4616). CNN-LSTM is the weakest (3.0707, 3.7820). GRU is therefore selected as the price model, but its advantage over LSTM is marginal and should not be over-interpreted. The price RZ of 0.6548 is clearly lower than the load RZ of 0.8882, which shows that price is harder to predict than load, in line with the literature [1], [12].

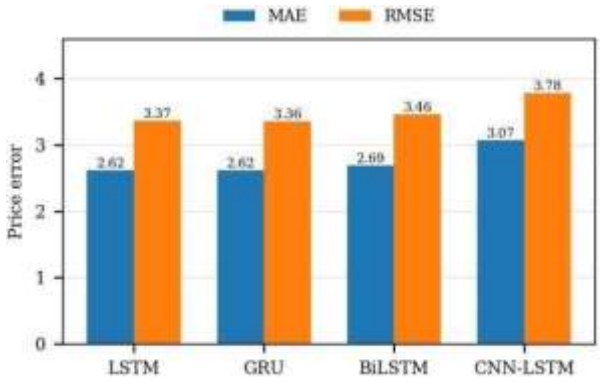


Fig. 4. Price forecasting error (MAE and RMSE) of the four models.



Fig. 5. RZ score of the four models forload and price (CNN-LSTM price RZ not reported).

D. Web-Based Dashboard

The trained models are deployed in a Streamlit dashboard with seven pages. *Dashboard* shows key indicators of the next 24 hours: predicted peak load and the hour it occurs, the lowest predicted price and its hour, the average load, a rule-based energy health

24-Hour Electricity Forecasts

Forecast values exported from the Streamlit dashboard

Fig. 7. 24-Hour Electricity Load Forecast (LSTM)

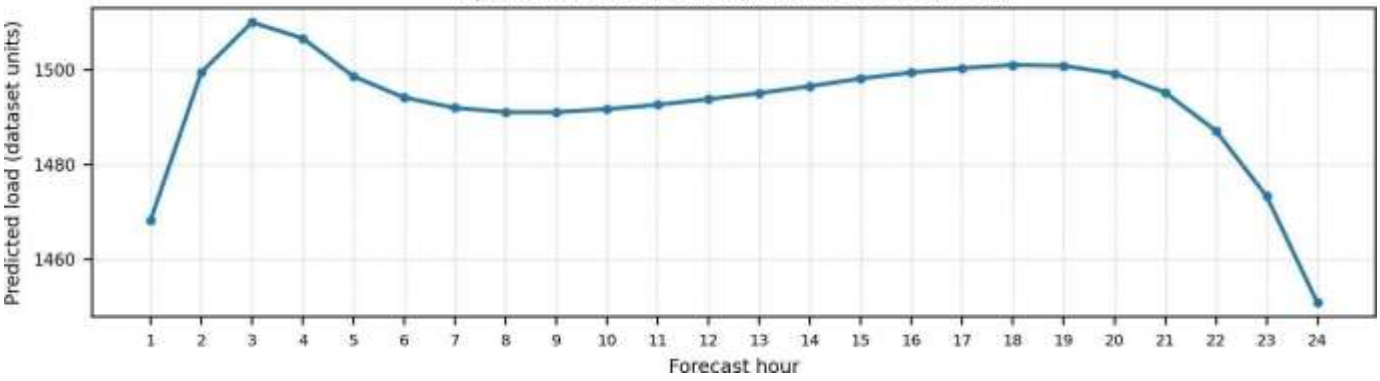
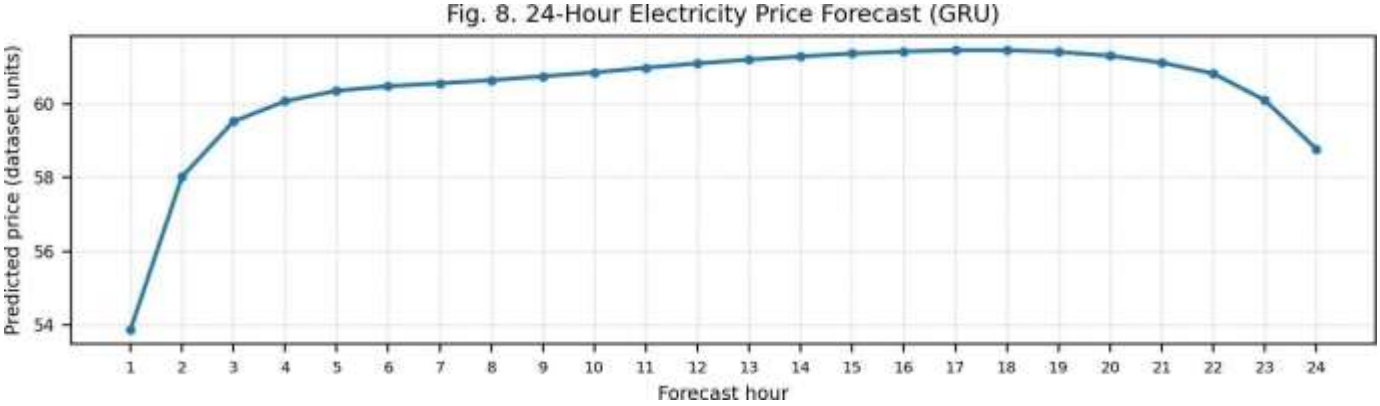


Fig. 8. 24-Hour Electricity Price Forecast (GRU)



The dashboard CSV contains 24 hourly predictions. The LSTM load forecast ranges from 1450.82 to 1510.00 dataset units. The GRU price forecast ranges from 53.87 to 61.45 dataset units.

These figures show the forecast values over the 24-hour horizon; they are not actual-versus-predicted comparisons.

score from 0 to 100 computed from the peak-to-average load ratio and price opportunity, and a smart appliance scheduler that compares the cost of running an appliance at the peak-load hour with the cost at the cheapest-price hour. *Single Prediction* lets the user enter the 17 feature values and returns the predicted load from the LSTM and the predicted price from the GRU, with a comparison against the current load and the rolling average price. *24-Hour Forecast* plots the load and price forecasts, marks peak and lowest hours, and offers the forecast table as a CSV download.

AI Recommendations classifies each forecast hour as high, moderate or low consumption using the 75th and 25th percentiles of the predicted load, lists the hours to avoid and the hours best suited to heavy appliances, and estimates the saving from shifting one appliance run from the peak hour to the cheapest hour. *Model Performance* presents the results in Table IV, and *Model Information* describes the models, the 17 features and the technologies used (Python, TensorFlow/Keras, Streamlit, Pandas, NumPy, Scikit-learn, Matplotlib and ReportLab). *Bill Generator* estimates a monthly bill from a slab tariff (0–100, 101–200, 201–300 and above 300 kWh) with editable rates, a fixed charge and tax, optionally using the forecast to estimate monthly consumption, splits the bill across common appliances, and exports the bill as a PDF.



Fig. 6. Illustrative Smart Energy Forecasting dashboard mockup.

The dashboard separates model inference from presentation: models and scalers are loaded once and cached, and a clear message is shown if a required file is missing. In Single Prediction, the 24-step input sequence is formed from the entered values with the hour of day advancing at each step, so the other features are assumed constant over the window.

V. Discussion

A. Interpretation of Results

The results show that recurrent networks can forecast load well and price moderately well from the 17-feature input. LSTM clearly outperforms the other architectures on load, consistent with its wide use for load forecasting [5], [20], while the price models are nearly tied. Fesseha *et al.* likewise find LSTM strong for load but note that the best model depends on the error measure and horizon [16], which is why several metrics are reported here.

B. Load Versus Price Forecasting

Load follows regular daily, weekly and seasonal cycles and depends on weather and calendar effects, so it is largely explained by the lagged and calendar features. Price shares these cycles but is also driven by fuel costs, outages, congestion and bidding behaviour, which are not among the 17 inputs and which produce spikes that are rare in training data [1], [12]. This explains why the price RZ (0.65) is much lower than the load RZ (0.89), and it suggests that adding supply-side and fuel-price variables is the most promising way to improve price forecasts.

C. Strengths and Limitations

The main strengths are a complete path from data to a usable dashboard, a like-for-like comparison of four architectures, separate best models for load and price, and scalers saved with the models so that deployment matches training. Several limitations should be noted. The evaluation uses one dataset and one train–validation–test split, so the results do not by themselves show accuracy in other markets or periods [16]. Only MAE, RMSE and RZ are reported, MAPE and prediction intervals are not, and no repeated runs or statistical tests are given. The price RZ is modest, and the LSTM–GRU price difference is too small to be conclusive. In the dashboard, the energy health score and the consumption levels are rule-based indicators and not model outputs, the bill is an estimate with user-set tariff rates, and its monthly estimate from the forecast assumes that the predicted load is in units that can be multiplied by 24 hours and 30 days.

D. Practical Implications

Better forecasts support concrete decisions. Price forecasts help generators and retailers prepare bidsand manage risk [15], and load forecasts support scheduling and demand response; Khan *et al.* use a day-ahead LSTM load forecast to drive a reinforcement-learning demand-response scheme that lowered aggregated load and cost [20]. The dashboard brings a small version of this to a household user by pointing out the peak hour, the cheapest hour and the cost difference for a chosen appliance. Consumption data from individual buildings is sensitive, so any use on real customer data would need data governance, and approaches such as federated learning [18] could keep raw data local.

VI. Conclusion and Future Work

This paper presented Smart Energy Forecasting, a deep-learning system for electricity load and price forecasting deployed as a Streamlit dashboard. Using 24-step sequences of 17 features, LSTM gave the best load forecast (MAE 39.6754, RMSE 48.7313, RZ 0.8882) and GRU gave the best price forecast (MAE 2.6210, RMSE 3.3600, RZ 0.6548) among LSTM, GRU, BiLSTM and CNN-LSTM. Future work includes adding MAPE and prediction intervals [2], [3], fuel-price and supply-side inputs for price, attention and decomposition components [6], [14], hyperparameter optimization [4], online adaptation to concept drift [17], repeated runs and external validation on other markets, and generating the dashboard forecast directly from live data instead of a stored forecast file.

VII. Research Significance and Practical Considerations

The project demonstrates an end-to-end approach in which data preparation, deep-learning training, model comparison, deployment and user-facing recommendations are treated as connected stages. Rather than treating accuracy as an isolated result, it shows how a forecast can be turned into advice about when to use electricity and what that use is likely to cost.

From a research perspective, the reported results are a baseline for future experiments and not a final conclusion about market behaviour. They describe the behaviour of four recurrent architectures on the specified test set, and they identify price forecasting as the weaker component. From a practical perspective, the system should be viewed as an educational prototype: real deployment would need independent validation, monitoring for concept drift [17], secure data handling and review by domain experts.

ACKNOWLEDGMENT

The authors acknowledge the guidance of Sharath Kumar S R B.E, M.Tech(PhD), Department of Information Science and Engineering, JNN College of Engineering, Shivamogga, for guidance and support during the project work.

REFERENCES

[1] K. Mustafa, S. Khan, S. Aslam, H. Herodotou, N. Ashraf, A. Daraz, and T. Alkhalifah, "Toward holistic energy management by electricity load and price forecasting: A comprehensive survey," *IEEE Access*, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3334164.

[2] R. Zhang, G. Li, and Z. Ma, "Adeep learningbased hybrid framework for day-ahead electricity price forecasting," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3014241.

[3] M. N. H. Mir, A. K. Biswas, M. S. A. Bhuiyan, M. G. R. Abir, M. F. Mridha, and M. J. Hossen, "ABMF-Net: An attentive Bayesian multi-stage deep learning model for robust forecasting of electricity price and demand," *IEEE Open J. Comput. Soc.*, 2025, doi: 10.1109/OJCS.2025.3579522.

[4] Y. Mohamed, M. M. Fouda, Z. M. Fadlullah, R. Abdelfattah, and M. I. Ibrahem, "BOL-LPP: A Bayesian-optimized LSTM modelfor day-ahead load price forecasting in the ERCOT market," *IEEE Open J. Comput. Soc.*, 2025, doi: 10.1109/OJCS.2025.3580107.

[5] I. M. Gaber and R. A. Ibrahim, "Hourly electricity price prediction applying deep learning for electricity market management," in *Proc. IEEE EEEIC / I&CPS Europe*, 2023, doi: 10.1109/EEEIC/ICPSEUROPE57605.2023.10194867.

[6] M. F. Azam and M. S. Younis, "Multi-horizon electricity load and price forecasting using an interpretable multi-head self-attention and EEMD-based framework," *IEEE Access*, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3086039.

[7] Y.-Y. Hong, J. V. Taylar, and A. C. Fajardo, "Locational marginal price forecasting using deep learning network optimized by mapping-based genetic algorithm," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.2994444.

[8] X. Cheng, I. Ilieva, B. Bremdal, S. Redhu, and S. Ø. Ottesen, "A novel spatial-temporal deep neural network for electricity price forecasting," in *Proc. Int. Conf. Applied Artificial Intelligence (ICAPAI)*, 2023, doi: 10.1109/ICAPAI58366.2023.10193970.

[9] A. A. Almazroi and N. Ayub, "Multi-task learning for electricity price forecasting and resource management in cloud based industrial IoT systems," *IEEE Access*, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3280857.

[10] R. Dash, A. Sinha, K. J. Reddy, C. Dhanamjayulu, and I. Kamwa, "Dynamic modeling and quantum-enhancedforecasting of multi-seasonal energy prices in simulated microgrid environments," *IEEE Access*, vol. 13, 2025, doi: 10.1109/ACCESS.2025.3570719.

[11] A. N. Alkawaz, A. Abdellatif, J. Kanesan, A. S. M. Khairuddin, and H. M. Gheni, "Day-ahead electricity price forecasting based on hybrid regression model," *IEEE Access*, vol. 10, 2022, doi: 10.1109/ACCESS.2022.3213081.

[12] M. Farhoumandi, S. Bahramirad, A. Alabdulwahab, M. Shahidehpour, F. Rahimi, A. Ipakchi, F. Albuyeh, and S. Mokhtari, "Fusion of deep learning and machine learning methods for hourly locational marginal price forecast in power systems," *iEnergy*, 2025, doi: 10.23919/IEN.2025.0019.

[13] A. Yousefi, O. A. Sianaki, and D. Sharafi, "Long-term electricity price forecast using machine learning techniques," in *Proc. IEEE PES Innovative Smart Grid Technologies Asia*, 2019.

[14] J. Hu, S. Cai, F. Li, and Y. Chen, "A data driven modelbased approach for medium-to-long-term electricity price forecasting in power markets," *Sci. Rep.*, vol. 15, art. 37046, 2025, doi: 10.1038/s41598 -025-20776-2.

[15] K. Lin and P. Feng, "Machine learning-based price forecasting and risk management in renewable energy markets," *Sustain. Energy Res.*, vol. 13, art. 13, 2026, doi: 10.1186/s40807-025-00234-3.

[16] S. B. Fesseha, B. Li, B. Qi, S. Chen, and F. Gong, "A comparative study of multi-step forecasting algorithms for real-time load profiling in smelting plant electricity demand," *IEEE Access*, 2026, doi: 10.1109/ACCESS.2026.3666755.

[17] M. N. Fekri, H. Patel, K. Grolinger, and V. Sharma, "Deep learning for load forecasting with smart meter data: Online adaptive recurrent neural network," *Appl. Energy*, 2021.

[18] D. Qin, C. Wang, Q. Wen, W. Chen, L. Sun, and Y. Wang, "Personalized federated DARTS for electricity load forecasting of individualbuildings," *IEEE Trans. Smart Grid*, 2023.

[19] M. Qureshi, M. A. Arbab, and S. ur Rehman, "Deep learning-based forecasting of electricity consumption," *Sci. Rep.*, vol. 14, art. 6489, 2024, doi: 10.1038/s41598-024-56602-4.

[20] D. A. Khan, A. Arshad, M. Lehtonen, and K. Mahmoud, "Combined DR pricing and voltage control using reinforcement learning based multi-agents and load forecasting," *IEEE Access*, vol. 10, 2022, doi: 10.1109/ACCESS.2022.3228836.