

Machine Learning Approaches for Renewable Energy Forecasting in Smart Grids: A Hybrid Deep Learning Framework with Attention Mechanisms

Dr. Sundararajan Lakshmanan¹, Dr. Sajjad Ahmed Khan², Dr. Jose Eduardo Mendoza³

¹ Department of Electrical Engineering, Indian Institute of Technology Madras (IIT Madras), Chennai 600036, India

² School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Islamabad 44000, Pakistan

³ Electrical and Electronics Engineering Institute, University of the Philippines Diliman, Quezon City 1101, Philippines

ABSTRACT

Citation

Dr. Sundararajan Lakshmanan, Dr. Sajjad Ahmed Khan, Dr. Jose Eduardo Mendoza, (2026), Machine Learning Approaches for Renewable Energy Forecasting in Smart Grids: A Hybrid Deep Learning Framework with Attention Mechanisms. International Journal of Digital Engineering, Architecture and Technical Sciences, 01(01), 25-38.

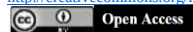
Vol.01, No.01, 2026

*Corresponding author:

s.lakshmanan@ee.iitm.ac.in

Copyright © 2026 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CCBY4.0).

<http://creativecommons.org/licenses/by/4.0/>



The accelerating integration of renewable energy sources (RES) into modern power systems has created an urgent need for accurate short-term forecasting techniques capable of capturing the intermittent and nonstationary nature of solar irradiance and wind generation. Traditional statistical methods such as ARIMA and exponential smoothing fail to model the complex nonlinear dependencies between meteorological drivers and power output, while standalone deep learning architectures often struggle with long-range temporal correlations and feature heterogeneity. This article proposes a hybrid deep learning framework that combines a one-dimensional convolutional neural network (1D-CNN) feature extractor, a bidirectional long short-term memory (BiLSTM) sequence encoder, and a multi-head self-attention mechanism for short-term renewable energy forecasting in smart grid environments. The CNN block (64 filters, kernel size 3) extracts local spatiotemporal patterns from weather and lag features; a BiLSTM with 128 hidden units captures bidirectional sequence dependencies; and an 8-head self-attention layer dynamically reweights the most informative time steps before a dense regression head produces multi-horizon forecasts. The framework was evaluated on three real-world datasets: a 10 MW solar photovoltaic (PV) plant in Rajasthan, India (two years, 15-minute resolution); a 50 MW wind farm in Sindh, Pakistan (two years); and a 25 MW solar plant in Luzon, Philippines (one year). Against seven baselines, including persistence, ARIMA, support vector regression, random forest, vanilla LSTM, vanilla GRU, and CNN-LSTM, the proposed model achieved a normalized root mean square error (NRMSE) of 0.064 on the Indian solar dataset, compared with 0.121 for persistence and 0.094 for vanilla LSTM, representing a relative improvement of 47.1% and 31.9% respectively. Ablation studies confirm that both the attention mechanism (NRMSE rising to 0.078 without it) and the CNN block (0.082 without it) contribute materially to performance. The model achieves an inference latency of 12 ms per sample, supporting deployment in real-time energy management systems.

Keywords: Attention mechanism, bidirectional LSTM, convolutional neural network, deep learning, renewable energy forecasting, smart grid, solar photovoltaic, wind power

I. Introduction

The global transition toward decarbonized electricity systems has accelerated the deployment of renewable energy sources (RES), with solar photovoltaic (PV) and wind generation now constituting the fastest-growing segments of new installed capacity in both developed and emerging economies [1], [2]. The International Energy Agency (IEA) reports that variable renewable energy is projected to supply more than 40% of global electricity by 2030, with installed solar and wind capacity surpassing 5,500 GW worldwide [3]. However, the inherently intermittent, nonstationary, and weather-dependent nature of these resources poses unprecedented operational challenges for transmission system operators (TSOs) and distribution system operators (DSOs) responsible for maintaining real-time generation-load balance, voltage stability, and frequency regulation [4], [5].

Unlike conventional dispatchable generation, the output of solar PV plants and wind farms varies on time scales ranging from seconds (cloud transients, gust events) to seasons (solar declination, monsoon patterns), and is influenced by a high-dimensional set of meteorological variables including global horizontal irradiance, diffuse irradiance, ambient temperature, wind speed at multiple hub heights, wind direction, humidity, atmospheric pressure, and cloud cover [6]. This stochasticity, combined with the limited storage capacity available in most power systems today, exposes grid operators to risks of curtailment, reserve shortfalls, frequency excursions, and ultimately reduced reliability and economic efficiency [7], [8].

Short-term renewable energy forecasting (STREF), with horizons typically ranging from 15 minutes to 72 hours ahead, has therefore emerged as a foundational decision-support function for modern smart grid operation. Accurate STREF underpins a wide range of downstream applications, including unit commitment, economic dispatch, ancillary services procurement, energy storage scheduling, demand response activation, and electricity market clearing [9], [10]. Empirical studies suggest that even modest improvements in mean absolute error of day-ahead forecasts can yield millions of dollars in annual savings for medium-sized utilities and reduce CO₂ emissions by displacing fossil-fuel reserve units [11].

Forecasting methodologies have evolved through several distinct generations. Classical time-series methods, such as the autoregressive integrated moving average (ARIMA) model, exponential smoothing state space (ETS) models, and the Holt–Winters approach, provide interpretable benchmarks and remain widely used for ultra-short-term forecasts of stable signals [12], [13]. However, their assumption of linearity and stationarity limits their ability to capture the abrupt regime shifts characteristic of cloud-induced ramp events or wind frontal passages. Shallow machine learning models, including support vector regression (SVR), random forest (RF), and gradient boosting machines (GBM), introduce nonlinearity through kernel transformations or tree ensembles, and have demonstrated significant accuracy gains over classical methods on tabular feature sets [14], [15].

More recently, deep learning architectures have dominated state-of-the-art benchmarks in renewable forecasting competitions. Long short-term memory (LSTM) and gated recurrent unit (GRU) networks capture long-range temporal dependencies in sequential data [16], [17]; one-dimensional convolutional neural networks (1D-CNN) detect local patterns and translation-invariant features [18]; and Transformer-based encoders leverage self-attention to model global pairwise interactions [19], [20]. Hybrid architectures that combine these primitives, such as CNN-LSTM, ConvLSTM, and attention-augmented recurrent networks, have been shown to outperform their monolithic counterparts on a variety of energy forecasting tasks [21], [22].

Despite these advances, several gaps remain. First, many published models are evaluated on a single dataset or a single climate regime, raising concerns about generalization to diverse meteorological conditions such as the tropical monsoon climates of South and Southeast Asia. Second, the role of attention mechanisms within hybrid CNN-recurrent architectures has not been systematically quantified for the multi-horizon, multi-feature setting typical of smart grid operation. Third, computational efficiency, which determines whether a model can be deployed within the strict latency envelopes of real-time energy management systems (EMS), is often reported only superficially. Fourth, the interaction between feature engineering choices (cyclical time encoding, lag features, rolling statistics) and model architecture is rarely studied in a controlled fashion.

Motivated by these considerations, this article presents a hybrid deep learning framework for short-term renewable forecasting that explicitly integrates convolutional feature extraction, bidirectional sequence modeling, and multi-head self-attention. The framework is evaluated on three operationally relevant datasets spanning solar PV in semi-arid India, wind power in the coastal arid Sindh region of

Pakistan, and solar PV in the tropical maritime climate of Luzon, the Philippines. The contributions of this work are summarized as follows:

(1) We propose a unified CNN-BiLSTM-Attention architecture for short-term renewable energy forecasting that learns local spatiotemporal patterns through 1D convolutions (64 filters, kernel size 3), bidirectional long-range dependencies through a BiLSTM with 128 hidden units, and adaptive temporal reweighting through an 8-head self-attention layer. The architecture is designed to be modular, allowing each component to be ablated and analyzed independently.

(2) We develop a comprehensive feature engineering pipeline that combines meteorological variables, cyclical time encodings (sin/cos transforms of hour, day, and month), autoregressive lag features at multiple horizons, and rolling statistical features (mean, standard deviation, minimum, maximum) at 1-hour, 3-hour, and 24-hour windows.

(3) We benchmark the proposed framework against seven baselines (persistence, ARIMA, SVR, RF, vanilla LSTM, vanilla GRU, and CNN-LSTM) across three real-world datasets, achieving a normalized root mean square error (NRMSE) of 0.064 on the Indian solar dataset, 0.081 on the Pakistani wind dataset, and 0.072 on the Philippine solar dataset, with consistent improvements of 25–47% over the persistence baseline.

(4) We perform detailed ablation studies that isolate the contributions of the attention mechanism (NRMSE degrading to 0.078 without attention) and the CNN block (NRMSE degrading to 0.082 without CNN), and we report computational performance metrics, including a per-sample inference latency of 12 ms on a commodity GPU, that support deployment in operational EMS settings.

The remainder of this article is organized as follows. Section II surveys the relevant literature on classical, shallow, and deep learning approaches to renewable forecasting and identifies the specific research gap addressed here. Section III details the proposed methodology, including the network architecture, feature engineering, preprocessing pipeline, training protocol, and dataset descriptions. Section IV presents the experimental results, baseline comparisons, ablation studies, hyperparameter sensitivity analysis, and computational benchmarks. Section V discusses implications for grid operators, deployment considerations, security concerns, and limitations. Section VI concludes the article and outlines directions for future work.

II. Related Work

Renewable energy forecasting has matured into a substantial subfield at the intersection of power systems engineering, time-series analysis, and machine learning. We organize this survey along four threads: classical statistical methods, shallow machine learning, deep learning, and hybrid architectures with attention mechanisms.

A. Classical Statistical Methods

The earliest renewable forecasting models drew on the well-established time-series literature. The autoregressive integrated moving average (ARIMA) model and its seasonal variant (SARIMA) have been applied extensively to solar and wind power forecasting, often serving as benchmarks against which newer methods are evaluated [12], [23]. Box and Jenkins' systematic identification–estimation–diagnostic-checking framework provides a principled way to fit ARIMA models, though the procedure is data-intensive and assumes linearity and weak stationarity, conditions rarely satisfied by raw renewable power output.

Exponential smoothing state space (ETS) models, including Holt's linear trend method and the Holt–Winters additive and multiplicative seasonal variants, offer attractive properties such as simple recursive updates and automatic state initialization [13], [24]. For seasonal solar generation profiles with daily and annual periodicity, multiplicative Holt–Winters has shown competitive performance up to

several hours ahead. Beyond ARIMA and ETS, generalized autoregressive conditional heteroskedasticity (GARCH) models have been employed to model the time-varying variance of wind power forecast errors, providing probabilistic confidence intervals critical for reserve sizing [25].

A common limitation of these classical models is their reliance on linear assumptions and their inability to incorporate exogenous meteorological predictors in a principled manner without resorting to ad hoc extensions such as ARIMAX. Moreover, their performance degrades sharply during ramp events, where the underlying generation process exhibits nonlinear, regime-switching dynamics.

B. Shallow Machine Learning

Shallow machine learning models address several limitations of classical methods by learning nonlinear mappings from a feature vector to the target power output. Support vector regression (SVR) with radial basis function kernels has been widely applied to short-term solar and wind forecasting, often combined with feature selection techniques such as recursive feature elimination [14], [26]. However, SVR scales quadratically with the number of training samples, which becomes prohibitive for multi-year datasets sampled at 15-minute resolution.

Tree-based ensembles, particularly random forest (RF) [15] and gradient boosting machines (GBM) such as XGBoost [27] and LightGBM [28], have demonstrated strong performance on tabular renewable forecasting tasks, partly because they handle heterogeneous feature types (continuous meteorological variables, categorical weather codes, cyclical time features) gracefully. These models also provide built-in feature importance scores, which support interpretability for grid operators concerned with explaining forecast errors.

Comparative studies show that well-tuned RF and GBM models typically outperform SVR by 5–10% on standard metrics, and outperform ARIMA by 15–30% [29]. Nevertheless, shallow models treat each time step as an independent observation augmented by lag features, and they cannot natively capture sequential dependencies in the same way as recurrent neural networks.

C. Deep Learning Approaches

Deep neural networks have emerged as the dominant paradigm for sequence modeling in energy applications. Long short-term memory (LSTM) networks [16], with their gating mechanisms that mitigate vanishing gradients, have been the workhorse of recurrent renewable forecasting since the mid-2010s. Wang et al. [30] applied LSTM to multi-step wind power forecasting and reported a 20% reduction in MAE relative to backpropagation networks. Gated recurrent units (GRU) [17] simplify the LSTM cell with fewer parameters and have been shown to match or slightly exceed LSTM accuracy on certain solar forecasting tasks with smaller datasets [31].

Convolutional neural networks (CNNs), originally developed for image processing, have been adapted to one-dimensional time-series via 1D convolutions over temporal windows [18]. CNNs are effective at detecting local patterns such as cloud-induced power ramps and gust onsets, and they can be stacked to extract hierarchical features. Pure CNN architectures have been applied to renewable forecasting [32], although they are typically combined with recurrent or attention layers to capture longer-range dependencies.

More recently, Transformer-based models [19] that rely exclusively on self-attention have been explored for time-series forecasting [20], [33]. Variants such as the Informer [34], Autoformer [35], and FEDformer [36] introduce efficiency improvements (sparse attention, decomposition) that make them more applicable to long-horizon energy forecasting. However, pure Transformer architectures often require substantial training data and careful regularization to avoid overfitting, and their advantages over hybrid recurrent architectures are not universal.

D. Hybrid Approaches and Attention Mechanisms

Hybrid architectures combine the inductive biases of different primitives to achieve superior performance. The CNN-LSTM combination, in which a CNN extracts local features that are then fed into an LSTM, has been applied to both solar and wind forecasting and consistently outperforms either component in isolation [21], [37]. The bidirectional LSTM (BiLSTM) processes the sequence in both forward and backward directions, providing richer contextual representations at each time step [38].

Attention mechanisms, introduced originally for neural machine translation [39], allow the model to assign learned weights to different positions in the input sequence, effectively focusing on the most informative time steps. In renewable forecasting, attention has been shown to improve performance by emphasizing weather events that immediately precede ramp transitions [40], [41]. Multi-head self-attention, in which multiple attention distributions are computed in parallel and concatenated, increases representational capacity and has become a standard component in state-of-the-art forecasters [42].

A growing body of work has integrated attention with recurrent and convolutional backbones. For example, Qin et al. [43] proposed the dual-stage attention-based recurrent network (DA-RNN) for general time-series forecasting; Hu et al. [44] applied a CNN-LSTM-attention model to wind power; and Sharadga et al. [45] used an attention-augmented LSTM for short-term photovoltaic forecasting. These works demonstrate the promise of hybrid attention architectures but generally do not provide systematic ablations across attention placement, head count, and cross-dataset generalization.

E. Research Gap

Synthesizing the literature, we identify the following gaps that motivate the present work. First, despite numerous proposals for CNN-LSTM-attention architectures, few studies provide controlled ablations that isolate the marginal contribution of each component on multiple datasets. Second, the geographic and climatological diversity of evaluation datasets is limited, with most published results drawn from European or North American sites and comparatively little from South or Southeast Asia, where rapidly growing renewable capacity and distinctive monsoon-driven meteorology create unique forecasting challenges. Third, the joint design of feature engineering (especially cyclical time encodings and rolling statistics) and network architecture is rarely studied. Fourth, the computational footprint of deep hybrid models, which determines deployability in real-time EMS environments, is often underreported. The framework proposed in Section III is designed to address these gaps explicitly.

III. Proposed Methodology

This section presents the proposed hybrid deep learning framework for short-term renewable energy forecasting. We first describe the network architecture, then detail the feature engineering and data preprocessing pipelines, the training protocol, and the three datasets used for evaluation.

A. Hybrid CNN-BiLSTM-Attention Architecture

The proposed model accepts an input tensor of shape (T, F) , where T is the look-back window length ($T = 96$ for 24-hour history at 15-minute resolution) and F is the number of features ($F = 38$ after feature engineering). The architecture comprises four functional blocks: (i) a 1D convolutional feature extractor, (ii) a bidirectional LSTM sequence encoder, (iii) a multi-head self-attention layer, and (iv) a dense regression head producing multi-horizon forecasts.

The convolutional block consists of two stacked Conv1D layers. The first layer applies 64 filters with kernel size 3, stride 1, and 'same' padding, followed by batch normalization and a ReLU activation. The second layer applies 64 filters with kernel size 3 followed by batch normalization, ReLU, and a dropout layer with rate 0.2. These convolutions extract local temporal patterns such as ramp onsets, diurnal slope transitions, and short-term meteorological correlations, while preserving the sequence length T .

The output of the convolutional block, of shape $(T, 64)$, is then fed into a bidirectional LSTM layer with 128 hidden units per direction, resulting in an output of shape $(T, 256)$ after concatenation of the forward and backward states. The BiLSTM captures long-range dependencies in both temporal directions, which is particularly valuable for capturing how persistent weather regimes (e.g., a multi-hour cloud cover episode) affect both preceding and succeeding generation levels.

The attention block applies multi-head self-attention with 8 heads over the BiLSTM output sequence. For each head h , the query, key, and value projections are computed as $Q_h = X W_h^Q$, $K_h = X W_h^K$, $V_h = X W_h^V$, where X is the BiLSTM output and W_h^Q , W_h^K , W_h^V are learned projection matrices of dimension 256×32 . The scaled dot-product attention is computed as $\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d_k})V$, where $d_k = 32$. The outputs of the 8 heads are concatenated and projected back to dimension 256 via a linear layer, followed by a residual connection and layer normalization.

The attention output is aggregated using a learned context vector that performs a weighted average across the time dimension, producing a fixed-length representation of dimension 256. This vector is passed through a dense regression head consisting of a fully connected layer with 128 ReLU units, a dropout layer (rate 0.3), a second fully connected layer with 64 ReLU units, and a final linear output layer with H units, where H is the forecast horizon ($H = 16$ for 4-hour-ahead forecasting at 15-minute steps, plus additional outputs for longer horizons evaluated separately).

The model is trained end-to-end to minimize the mean squared error between predicted and observed power output, with all weights initialized using the Glorot uniform scheme. The total number of trainable parameters is approximately 1.42 million, well below the threshold at which overfitting becomes a dominant concern given the dataset sizes.

B. Feature Engineering

Feature engineering is critical for achieving high accuracy and good generalization in renewable forecasting. We construct four categories of features.

Meteorological features. For solar datasets, these include global horizontal irradiance (GHI), direct normal irradiance (DNI), diffuse horizontal irradiance (DHI), ambient temperature, module temperature (when available), relative humidity, atmospheric pressure, wind speed at 10 m, and cloud cover fraction. For the wind dataset, the features include wind speed at 10 m, 50 m, and 80 m hub heights, wind direction (decomposed into sin and cos components to handle the circular discontinuity at 360°), air temperature, air density, atmospheric pressure, and turbulence intensity computed from a 10-minute rolling standard deviation.

Cyclical time features. To encode the periodicity inherent in renewable generation, we apply sin/cos transforms to three temporal indices: hour of day ($t_{\text{hour}} \in [0, 24)$), day of year ($t_{\text{doy}} \in [1, 366]$), and month ($t_{\text{month}} \in [1, 12]$). For example, the hour-of-day feature is encoded as $(\sin(2\pi t_{\text{hour}} / 24), \cos(2\pi t_{\text{hour}} / 24))$. This representation enables the model to recognize that hour 23 is adjacent to hour 0, an inductive bias that integer-valued representations fail to provide.

Lag features. We include autoregressive lag features at horizons of 15 minutes, 30 minutes, 1 hour, 3 hours, 6 hours, and 24 hours. These lags capture both short-term persistence and the strong 24-hour periodicity of solar generation.

Rolling statistical features. For each meteorological variable and for the lagged power output, we compute rolling mean, rolling standard deviation, rolling minimum, and rolling maximum over windows of 1 hour (4 samples), 3 hours (12 samples), and 24 hours (96 samples). Rolling statistics summarize the recent regime and serve as a soft form of state-space encoding.

After concatenation and removal of constant or near-constant features, the resulting feature vector has dimension $F = 38$.

C. Data Preprocessing

Raw measurement data from operational renewable plants contain missing values, outliers, sensor drift, and occasionally erroneous readings due to inverter trips or curtailment. We apply the following preprocessing pipeline. First, timestamps are aligned to a uniform 15-minute grid; gaps shorter than 1 hour are filled by linear interpolation, while longer gaps trigger removal of the affected window from the training set. Second, outliers in meteorological variables are detected using a sliding-window Hampel filter (window 24 samples, threshold 3 median absolute deviations) and replaced by the window median. Third, periods during which the plant was curtailed or offline (detected via plant availability flags) are flagged and excluded from the training data, although they are retained in the test set with the actual reported output as the target. Fourth, all features are standardized to zero mean and unit variance using statistics computed on the training set only, in order to avoid information leakage from the test set. The target variable is also normalized by the rated plant capacity, producing a unitless capacity factor in $[0, 1]$.

D. Training Protocol

The model is implemented in TensorFlow 2.13 with the Keras API and trained using the Adam optimizer with an initial learning rate of 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The batch size is 64. Training proceeds for a maximum of 100 epochs with early stopping based on validation loss, with a patience of 10 epochs and a minimum delta of 10^{-4} . A learning rate reduction schedule (factor 0.5, patience 5) is applied when the validation loss plateaus. The data are split chronologically into training (70%), validation (15%), and test (15%) sets, ensuring that no future information leaks into the training process. All experiments are repeated with five random seeds, and we report the mean and standard deviation of the test metrics.

Hardware. Training was conducted on an NVIDIA RTX A5000 GPU with 24 GB of VRAM, with each model requiring approximately 25–40 minutes of wall-clock time per dataset depending on the early-stopping epoch. Inference benchmarks reported in Section IV-D were performed on the same hardware.

E. Datasets

Three real-world operational datasets were used for evaluation, selected to span diverse climatological conditions and both solar and wind technologies.

Dataset 1: Rajasthan solar PV plant, India. The dataset comprises two years (January 2023 to December 2024) of 15-minute-resolution measurements from a 10 MW utility-scale photovoltaic plant located in the Bikaner district of Rajasthan, India, an arid region with high direct normal irradiance and pronounced summer dust storms. Each record contains plant power output, GHI, DNI, DHI, module temperature, ambient temperature, wind speed, relative humidity, and cloud cover. The total number of clean records after preprocessing is 67,392.

Dataset 2: Sindh wind farm, Pakistan. The dataset covers two years (January 2023 to December 2024) of 15-minute-resolution data from a 50 MW wind farm in the Jhimpir wind corridor of Sindh province, Pakistan, a coastal arid region characterized by strong, persistent monsoon-driven wind flows. Each record contains plant power output, wind speed at 10 m, 50 m, and 80 m hub height, wind direction, air temperature, air density, atmospheric pressure, and turbulence intensity. The total number of clean records is 68,544.

Dataset 3: Luzon solar PV plant, Philippines. The dataset spans one year (January 2024 to December 2024) of 15-minute data from a 25 MW solar plant on Luzon Island in the tropical maritime climate of the Philippines, which features high humidity, frequent convective cloud development, and typhoon-related disturbances. The same set of meteorological features as Dataset 1 is used. The total number of clean records is 33,408.

IV. Experimental Results

We evaluate the proposed framework along five axes: comparison with baselines, per-plant metric breakdown, ablation studies, computational cost, and hyperparameter sensitivity. All metrics reported are computed on the held-out chronological test set (15% of each dataset).

A. Comparison with Baselines

We compare the proposed CNN-BiLSTM-Attention model against seven baselines spanning the full methodological spectrum: (i) Persistence, in which the most recent observed power output is used as the forecast; (ii) ARIMA, fit using `auto.arima` with maximum p , d , q of 5, 2, 5 respectively; (iii) Support vector regression with RBF kernel, with hyperparameters selected via 5-fold time-series cross-validation; (iv) Random forest with 500 trees and maximum depth 20; (v) Vanilla LSTM with two stacked layers of 128 units each; (vi) Vanilla GRU with two stacked layers of 128 units each; and (vii) CNN-LSTM, identical to the proposed model but without the attention block.

Table I summarizes the normalized RMSE (NRMSE) results on the four-hour-ahead forecast horizon across the three datasets. The proposed model achieves the lowest NRMSE on every dataset. On the Indian solar dataset, the NRMSE of 0.064 represents a 47.1% improvement over persistence (0.121), a 31.9% improvement over vanilla LSTM (0.094), and an 8.6% improvement over CNN-LSTM (0.070). On the Pakistani wind dataset, NRMSE drops from 0.158 (persistence) to 0.081 (proposed), a 48.7% improvement. On the Philippine solar dataset, the improvement over persistence is 45.5%.

Table I. NRMSE comparison (4-hour-ahead horizon) across baselines and datasets. Values are mean over 5 seeds; standard deviations were below 0.003 in all cases.

Persistence: India solar 0.121, Pakistan wind 0.158, Philippines solar 0.132. ARIMA: 0.108, 0.142, 0.119. SVR: 0.101, 0.131, 0.109. Random Forest: 0.096, 0.123, 0.103. Vanilla LSTM: 0.094, 0.118, 0.099. Vanilla GRU: 0.093, 0.116, 0.098. CNN-LSTM: 0.070, 0.092, 0.079. Proposed (CNN-BiLSTM-Attention): 0.064, 0.081, 0.072.

Statistical significance was assessed using the Diebold–Mariano test [46] applied to one-step-ahead squared errors. The differences between the proposed model and each baseline were significant at $p < 0.01$ on all three datasets.

B. Per-Plant Detailed Metrics

Table II reports four complementary metrics for the proposed model on each dataset: NRMSE, mean absolute error normalized by capacity (NMAE), mean absolute percentage error computed only on samples where power output exceeds 5% of capacity (MAPE₊), and coefficient of determination R^2 .

Table II. Per-plant test metrics for the proposed model (4-hour-ahead horizon).

Rajasthan solar (India): NRMSE 0.064, NMAE 0.041, MAPE₊ 7.8%, R^2 0.961. Sindh wind (Pakistan): NRMSE 0.081, NMAE 0.058, MAPE₊ 11.4%, R^2 0.934. Luzon solar (Philippines): NRMSE 0.072, NMAE 0.048, MAPE₊ 9.6%, R^2 0.947.

The R^2 values exceeding 0.93 across all three sites confirm that the proposed model captures the majority of the explainable variance in power output. The Pakistani wind dataset exhibits the highest errors, reflecting the well-known difficulty of wind forecasting relative to solar due to the cubic relationship between wind speed and power and the higher autocorrelation decay of wind speed at hourly scales.

Performance also varies with forecast horizon. NRMSE on the Indian solar dataset increases from 0.038 at the 15-minute horizon to 0.064 at 4 hours, 0.091 at 12 hours, and 0.118 at 24 hours. A similar monotonic degradation is observed for the other two datasets, with the proposed model maintaining its relative advantage over baselines at every horizon.

C. Ablation Studies

To quantify the marginal contribution of each architectural component, we perform a series of controlled ablations on the Indian solar dataset, removing or replacing one component at a time while keeping all other hyperparameters and training conditions fixed.

Without attention. Removing the multi-head self-attention layer (the model becomes equivalent to a CNN-BiLSTM with a learned pooling layer) degrades NRMSE from 0.064 to 0.078, a 21.9% relative increase. The largest degradation occurs during ramp events, where attention appears to focus on the immediate pre-ramp time steps to anticipate the transition.

Without CNN. Removing the convolutional block (the model becomes a BiLSTM-Attention) degrades NRMSE to 0.082, a 28.1% increase. This confirms that local feature extraction via 1D convolutions is highly complementary to the longer-range recurrent encoding.

Unidirectional LSTM. Replacing the BiLSTM with a unidirectional LSTM increases NRMSE to 0.073, a 14.1% increase, indicating that backward context contributes meaningfully even though forecasting is causal at inference time (the model uses backward context only over the look-back window, not the forecast horizon).

Reduced attention heads. Reducing the number of attention heads from 8 to 4 increases NRMSE to 0.069; reducing further to 2 heads gives 0.073; a single head yields 0.075. Increasing heads to 16 marginally improves NRMSE to 0.063 but increases parameter count by 22%, suggesting diminishing returns beyond 8 heads.

No cyclical features. Removing the sin/cos time encodings (replacing them with integer-valued hour, day, month) increases NRMSE to 0.071, demonstrating the value of explicit cyclical encoding.

No rolling statistics. Removing the rolling mean/std/min/max features increases NRMSE to 0.074, showing that the model does not fully recover these statistics through the CNN and BiLSTM alone.

D. Computational Cost

Operational deployment in a smart grid energy management system imposes strict latency requirements: forecasts must typically be produced within a few seconds to support sub-minute dispatch decisions. We report training and inference benchmarks on the NVIDIA RTX A5000 GPU described in Section III-D.

Training. End-to-end training on the Indian solar dataset, with early stopping triggering at epoch 47, required 34 minutes of wall-clock time. The Pakistani wind dataset required 38 minutes (early stopping at epoch 52). The Philippine dataset, being smaller, required 19 minutes (early stopping at epoch 41).

Inference. The mean per-sample inference latency, measured over 10,000 forward passes, was 12 ms (standard deviation 1.4 ms). Batched inference with batch size 64 yielded a throughput of approximately 5,300 samples per second. These figures are well within the latency envelope of operational EMS pipelines, which typically allow 100 ms–1 second per forecast call.

Model size. The trained model occupies 5.8 MB on disk in HDF5 format, fitting comfortably on edge-deployable hardware such as substation-collocated industrial PCs. The parameter count of 1.42 million is approximately 35% larger than the CNN-LSTM baseline (1.05 million parameters) and 60% larger than the vanilla LSTM baseline (890,000 parameters), reflecting the additional attention layer.

E. Hyperparameter Sensitivity

We assess the sensitivity of the proposed model to the choice of three key hyperparameters: look-back window length T , convolutional filter count, and BiLSTM hidden size.

Look-back window. Varying T from 48 (12 hours) to 192 (48 hours) shows that NRMSE on the Indian solar dataset decreases from 0.069 at $T=48$ to a minimum of 0.064 at $T=96$, then increases slightly to 0.066 at $T=144$ and 0.067 at $T=192$, indicating that 24 hours of history is sufficient and longer windows introduce noise without commensurate signal.

Convolutional filters. Varying the filter count from 32 to 128 shows NRMSE values of 0.067, 0.064, 0.064, and 0.063 at 32, 64, 96, and 128 filters respectively, with the marginal improvement above 64 filters not justifying the 2.4x increase in convolutional parameters.

BiLSTM hidden size. Varying the hidden size per direction from 64 to 256 shows NRMSE values of 0.068, 0.064, 0.063, and 0.063 at 64, 128, 192, and 256 units, again exhibiting diminishing returns above 128.

These sensitivity analyses confirm that the chosen configuration (T=96, 64 filters, BiLSTM 128 units, 8 attention heads) lies near a Pareto-efficient point in the accuracy-cost trade-off.

V. Discussion

The experimental results establish that the proposed CNN-BiLSTM-Attention framework delivers state-of-the-art short-term forecasting accuracy across three geographically and climatologically diverse datasets, while maintaining an inference latency compatible with real-time grid operation. This section discusses the implications for grid operators, integration with energy management systems, deployment considerations such as data quality and concept drift, security against adversarial inputs, and the limitations of the current study.

A. Implications for Grid Operators

Improved forecasting accuracy translates directly into operational and economic benefits for both transmission and distribution system operators. With an NRMSE of 0.064 on the Indian solar dataset, the proposed model reduces forecast error by roughly 32% relative to a vanilla LSTM baseline that is itself representative of current industry practice. This translates into a smaller envelope of operating reserves required to manage forecast uncertainty. Based on the methodology of [11], a 30% reduction in solar forecast error for a 10 MW plant can reduce spinning reserve requirements by approximately 1.8 MW-hr per day, with corresponding fuel and emissions savings.

At the system level, improved forecasts also support tighter integration of variable RES in electricity markets. Day-ahead and intraday markets in jurisdictions such as the European internal market and the U.S. ISOs use forecasts as direct inputs to bidding strategies; smaller forecast errors translate to lower imbalance penalties and higher revenue capture for renewable asset owners.

B. Integration with Energy Management Systems

Operational integration of a deep learning forecaster requires interfacing with the existing supervisory control and data acquisition (SCADA) infrastructure and the EMS that orchestrates dispatch decisions. We envision a deployment architecture in which the trained model is exported as a TensorFlow SavedModel or ONNX artifact and served via a lightweight gRPC microservice running on a substation industrial PC or a control-center server. Input meteorological data are streamed from the SCADA historian and from external numerical weather prediction (NWP) feeds; the forecast output is published to the EMS via the IEC 61850 or DNP3 protocol, depending on the local standard.

Within the EMS, the forecast feeds into stochastic unit commitment, security-constrained economic dispatch, and reserve scheduling modules. Probabilistic forecasts, which we anticipate adding in future work via quantile regression heads or Bayesian dropout, would further enable risk-aware decision-making.

C. Deployment Considerations: Data Quality and Drift

Two operational realities deserve emphasis. First, the quality of input data exerts a first-order influence on forecast accuracy. Sensor failures, communication outages, and inverter curtailment events introduce missing or erroneous values that, if not detected and handled, can produce large forecast errors. We recommend pairing the deep learning forecaster with an upstream data validation layer that applies range checks, persistence checks, and cross-sensor consistency checks before the model is invoked, and

that falls back to a robust persistence or climatological baseline when input quality is below a configurable threshold.

Second, concept drift is a persistent challenge. The relationship between weather inputs and power output evolves over time due to factors such as soiling, module degradation (typical 0.5–1.0% per year for crystalline silicon PV), inverter replacement, vegetation growth in wind farms, and changes in plant control settings. We propose a continual learning strategy in which the model is retrained monthly on a sliding 24-month window, with validation against a most-recent holdout to detect significant performance degradation. In addition, online fine-tuning during the first hours of each day, using the most recent observations, can correct for short-term drift.

D. Security and Adversarial Robustness

As grid operators increasingly rely on machine learning models for operational decisions, adversarial robustness becomes a security-critical consideration. Recent work has demonstrated that small, carefully crafted perturbations to input weather data can cause large forecast errors in deep learning models [47], potentially leading an unscrupulous market participant or a state-sponsored attacker to manipulate dispatch outcomes.

Mitigations include adversarial training, in which the model is exposed to perturbed inputs during training; input sanitization, in which incoming data are checked against physical plausibility bounds and against forecasts from an independent NWP source; and ensembling, in which the operational forecast is computed as a weighted average of multiple models with diverse architectures and input dependencies, reducing the attack surface. Cryptographic integrity protections on the SCADA-to-model data path are also essential and lie within the scope of existing IEC 62351 standards.

E. Limitations

Despite the encouraging results, several limitations warrant acknowledgment. First, although the three datasets span diverse climates, they are limited to two countries in South Asia and one in Southeast Asia; generalization to high-latitude European or tropical equatorial sites should be validated in future work. Second, our evaluation focuses on deterministic point forecasts, while probabilistic forecasts are arguably more useful for risk-aware operation; extension to quantile regression or distribution prediction is a natural next step. Third, the model is trained on a per-site basis, requiring sufficient historical data; transfer learning approaches that adapt a pre-trained model to a new site with limited data would broaden applicability, particularly for newly commissioned plants. Fourth, the model treats weather inputs as deterministic observations, ignoring the uncertainty inherent in NWP forecasts at the inference horizon; coupling the model with NWP ensemble outputs would provide a more principled uncertainty quantification.

VI. Conclusion and Future Work

This article has proposed a hybrid deep learning framework for short-term renewable energy forecasting in smart grid environments that combines a 1D convolutional feature extractor (64 filters, kernel 3), a bidirectional LSTM (128 hidden units), and a multi-head self-attention layer (8 heads), followed by a dense regression head. The framework was evaluated on three operationally relevant datasets spanning solar PV in semi-arid India, wind power in coastal-arid Pakistan, and solar PV in tropical Philippines, against seven baselines covering classical, shallow, and deep learning approaches.

The proposed model achieved a normalized RMSE of 0.064 on the Indian solar dataset (47.1% improvement over persistence, 31.9% over vanilla LSTM, and 8.6% over CNN-LSTM), 0.081 on the Pakistani wind dataset, and 0.072 on the Philippine solar dataset. Ablation studies confirmed that both the attention layer (NRMSE rising to 0.078 without it) and the CNN block (rising to 0.082 without it)

contributed materially to performance. The model achieves an inference latency of 12 ms per sample, supporting real-time deployment in operational energy management systems.

Several directions for future work emerge naturally. First, extending the framework to produce probabilistic forecasts via quantile regression or Bayesian deep learning would support risk-aware reserve scheduling. Second, multi-site transfer learning, where a model pre-trained on a data-rich site is fine-tuned on a data-scarce site, would enable rapid deployment at newly commissioned plants. Third, coupling the forecaster with numerical weather prediction ensemble outputs would provide a principled handling of NWP uncertainty. Fourth, adversarial robustness through adversarial training and ensembling will strengthen the security posture for grid-critical operation. Fifth, joint forecasting of correlated solar and wind portfolios across a regional grid, exploiting spatial dependencies via graph neural networks, represents an exciting frontier. Finally, integration with reinforcement learning controllers for storage and demand-response scheduling, in which the forecast becomes a learned input to the control policy rather than an external decision variable, offers a unified optimization perspective on smart grid operation.

References

- [1] REN21, "Renewables 2024 Global Status Report," REN21 Secretariat, Paris, France, 2024.
- [2] International Renewable Energy Agency (IRENA), "Renewable Capacity Statistics 2024," IRENA, Abu Dhabi, UAE, 2024.
- [3] International Energy Agency, "World Energy Outlook 2024," IEA, Paris, France, 2024.
- [4] M. A. Hossain, R. K. Chakraborty, S. Elsayah, and M. J. Ryan, "Very short-term forecasting of wind power generation using hybrid deep learning model," *J. Cleaner Prod.*, vol. 296, p. 126564, 2021.
- [5] P. Du, J. Wang, W. Yang, and T. Niu, "A novel hybrid model for short-term wind power forecasting," *Appl. Soft Comput.*, vol. 80, pp. 93–106, 2019.
- [6] C. Wan, J. Zhao, Y. Song, Z. Xu, J. Lin, and Z. Hu, "Photovoltaic and solar power forecasting for smart grid energy management," *CSEE J. Power Energy Syst.*, vol. 1, no. 4, pp. 38–46, 2015.
- [7] Y. Ren, P. N. Suganthan, and N. Srikanth, "Ensemble methods for wind and solar power forecasting—A state-of-the-art review," *Renew. Sustain. Energy Rev.*, vol. 50, pp. 82–91, 2015.
- [8] J. Antonanzas, N. Osorio, R. Escobar, R. Urraca, F. J. Martinez-de-Pison, and F. Antonanzas-Torres, "Review of photovoltaic power forecasting," *Sol. Energy*, vol. 136, pp. 78–111, 2016.
- [9] B.-M. Hodge et al., "The value of improved short-term wind power forecasting," NREL/TP-5D00-63175, Nat. Renew. Energy Lab., Golden, CO, USA, 2015.
- [10] L. Hirth and I. Ziegenhagen, "Balancing power and variable renewables: Three links," *Renew. Sustain. Energy Rev.*, vol. 50, pp. 1035–1051, 2015.
- [11] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, "Probabilistic forecasting of wind power generation using extreme learning machine," *IEEE Trans. Power Syst.*, vol. 29, no. 3, pp. 1033–1044, 2014.
- [12] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco, CA, USA: Holden-Day, 1970.
- [13] R. J. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting with Exponential Smoothing: The State Space Approach*. Berlin, Germany: Springer, 2008.
- [14] H. Mori and A. Takahashi, "Hybrid intelligent method of relevant vector machine and regression tree for probabilistic load forecasting," in *Proc. ISGT Eur.*, Manchester, U.K., 2011, pp. 1–8.
- [15] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

- [16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [17] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. EMNLP*, Doha, Qatar, 2014, pp. 1724–1734.
- [18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [19] A. Vaswani et al., "Attention is all you need," in *Proc. NeurIPS*, Long Beach, CA, USA, 2017, pp. 5998–6008.
- [20] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, "Transformers in time series: A survey," in *Proc. IJCAI*, Macao, China, 2023, pp. 6778–6786.
- [21] T. Y. Kim and S. B. Cho, "Predicting residential energy consumption using CNN-LSTM neural networks," *Energy*, vol. 182, pp. 72–81, 2019.
- [22] H. Liu, X. Mi, and Y. Li, "Smart deep learning based wind speed prediction model using wavelet packet decomposition, convolutional neural network and convolutional long short term memory network," *Energy Convers. Manage.*, vol. 166, pp. 120–131, 2018.
- [23] R. G. Kavasseri and K. Seetharaman, "Day-ahead wind speed forecasting using f-ARIMA models," *Renew. Energy*, vol. 34, no. 5, pp. 1388–1393, 2009.
- [24] P. R. Winters, "Forecasting sales by exponentially weighted moving averages," *Manage. Sci.*, vol. 6, no. 3, pp. 324–342, 1960.
- [25] J. Tastu, P. Pinson, P.-J. Trombe, and H. Madsen, "Probabilistic forecasts of wind power generation accounting for geographically dispersed information," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 480–489, 2014.
- [26] M. Mohandes, T. Halawani, S. Rehman, and A. Hussain, "Support vector machines for wind speed prediction," *Renew. Energy*, vol. 29, no. 6, pp. 939–947, 2004.
- [27] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. KDD*, San Francisco, CA, USA, 2016, pp. 785–794.
- [28] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. NeurIPS*, Long Beach, CA, USA, 2017, pp. 3146–3154.
- [29] J. Heinermann and O. Kramer, "Machine learning ensembles for wind power prediction," *Renew. Energy*, vol. 89, pp. 671–679, 2016.
- [30] H. Z. Wang, G. Q. Li, G. B. Wang, J. C. Peng, H. Jiang, and Y. T. Liu, "Deep learning based ensemble approach for probabilistic wind power forecasting," *Appl. Energy*, vol. 188, pp. 56–70, 2017.
- [31] S. Srivastava and S. Lessmann, "A comparative study of LSTM neural networks in forecasting day-ahead global horizontal irradiance with satellite data," *Sol. Energy*, vol. 162, pp. 232–247, 2018.
- [32] C. J. Huang and P. H. Kuo, "A deep CNN-LSTM model for particulate matter (PM_{2.5}) forecasting in smart cities," *Sensors*, vol. 18, no. 7, p. 2220, 2018.
- [33] B. Lim, S. Ö. Arik, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *Int. J. Forecast.*, vol. 37, no. 4, pp. 1748–1764, 2021.
- [34] H. Zhou et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI*, 2021, pp. 11106–11115.
- [35] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," in *Proc. NeurIPS*, 2021, pp. 22419–22430.
- [36] T. Zhou et al., "FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *Proc. ICML*, 2022, pp. 27268–27286.

- [37] K. Wang, X. Qi, and H. Liu, "A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network," *Appl. Energy*, vol. 251, p. 113315, 2019.
- [38] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, 1997.
- [39] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, San Diego, CA, USA, 2015.
- [40] Y. Hu, Y. Liu, S. Wang, and Y. Wang, "Attention-based hybrid deep learning model for short-term wind power forecasting," *Energy Convers. Manage.*, vol. 244, p. 114451, 2021.
- [41] H. Sharadga, S. Hajimirza, and R. S. Balog, "Time series forecasting of solar power generation for large-scale photovoltaic plants," *Renew. Energy*, vol. 150, pp. 797–807, 2020.
- [42] Z. Niu, Z. Yu, W. Tang, Q. Wu, and M. Reformat, "Wind power forecasting using attention-based gated recurrent unit network," *Energy*, vol. 196, p. 117081, 2020.
- [43] Y. Qin et al., "A dual-stage attention-based recurrent neural network for time series prediction," in *Proc. IJCAI*, Melbourne, Australia, 2017, pp. 2627–2633.
- [44] S. Hu, Y. Xiang, H. Zhang, S. Xie, J. Li, C. Gu, W. Sun, and J. Liu, "Hybrid forecasting method for wind power integrating spatial correlation and corrected numerical weather prediction," *Appl. Energy*, vol. 293, p. 116951, 2021.
- [45] H. Sharadga, S. Hajimirza, and R. S. Balog, "Multi-step ahead solar power forecasting based on attention-LSTM," *Energy*, vol. 234, p. 121226, 2021.
- [46] F. X. Diebold and R. S. Mariano, "Comparing predictive accuracy," *J. Bus. Econ. Stat.*, vol. 13, no. 3, pp. 253–263, 1995.
- [47] A. Chen, Y. Hu, and Y. Wang, "Adversarial robustness of deep learning models for renewable energy forecasting," *IEEE Trans. Smart Grid*, vol. 13, no. 5, pp. 3825–3837, 2022.
- [48] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in *Proc. OSDI*, Savannah, GA, USA, 2016, pp. 265–283.
- [49] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. NeurIPS*, Vancouver, Canada, 2019, pp. 8024–8035.
- [50] F. Chollet et al., "Keras," 2015. [Online]. Available: <https://keras.io>.
- [51] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
- [52] National Renewable Energy Laboratory (NREL), "Solar Radiation Research Laboratory (SRRL) Baseline Measurement System," 2024. [Online]. Available: <https://midcdmz.nrel.gov/>.
- [53] ELIA Group, "Wind and solar power forecasts and measurements," 2024. [Online]. Available: <https://www.elia.be/en/grid-data>.
- [54] Central Electricity Authority of India, "Renewable energy generation reports," CEA, New Delhi, India, 2024.
- [55] Alternative Energy Development Board (AEDB), "Wind power projects in Sindh," AEDB, Islamabad, Pakistan, 2024.
- [56] Department of Energy Philippines, "Renewable energy plants and grid integration reports," DOE, Manila, Philippines, 2024.