Probabilistic Models for Photovoltaic Production Forecasting: A Case Study at the Politecnico di Torino Campus

Probabilistic Models for Photovoltaic Production Forecasting: A Case Study at the Politecnico di Torino Campus

Antonio Costantino Marceddu*, Jacopo Sini*, Alessandro Ciocia†, Gianluca Cardinale*,
Filippo Spertino†, Gianfranco Chicco†, Bartolomeo Montrucchio*

*Department of Control and Computer Engineering
†Department of Energy
Politecnico di Torino
Turin, Italy
Email: {antonio.marceddu, jacopo.sini, alessandro.ciocia, filippo.spertino, gianfranco.chicco, bartolomeo.montrucchio}@polito.it, gianluca.cardinale@studenti.polito.it

***Abstract*—Forecasting energy production from renewable sources is essential for the effective management of electricity transmission grids. However, the intrinsic dependence of these sources on meteorological conditions introduces significant uncertainty. If the characteristics of a photovoltaic (PV) plant are known, deterministic models allow production to be estimated with a reasonable degree of accuracy. When these characteristics are unknown, these techniques cannot be used.**

Assuming the availability of historical data relating to generated power and meteorological data measured in the vicinity of the plant (such as irradiance and temperature), energy production can be estimated using probabilistic models. This paper, therefore, presents an analysis of this specific case at the Politecnico di Torino Campus. Established and recently developed Recurrent Neural Networks (RNNs) were trained on consumptive data to evaluate their strengths and limitations. The results were then analyzed to determine future actions with forecasting data.

***Index Terms*—Bidirectional Long Short Term Memory, Energy, Long Short Term Memory, Neural Networks, Photovoltaic Systems, Recurrent Neural Networks, Renewable Energy Sources, Solar Energy, Solar Power Generation.**

I. Introduction

RENEWABLE ENERGY PRODUCTION is a key point for making the modern society development compatible with the environment. Among the main technologies are photovoltaic (PV) panels, which convert solar energy into usable electricity, and wind turbines, which harness the energy in the convective motion of air (wind). The challenge of these technologies lies in their discontinuity, as they both depend on atmospheric conditions. The daily production of a PV plant is dependent on its own characteristics, such as the panel technology itself, the mounting angle and orientation of the panel, geographic position, and, of course, the day of the year, on which the Sun's trajectory depends. If these data are known, deterministic models can be created to estimate production. Unfortunately, these are not sufficient, since it also depends on the weather conditions, due to the capability of clouds to reflect into the space a part of the solar power, reducing the irradiance on the panels, and to a lesser extent, the temperature, whose increase reduces the conversion efficiency of the panels.

Deterministic models for PV generation define a straightforward relationship between the electrical output and weather variables, i.e., solar irradiance and module temperature. Unlike diode-based equivalent circuit approaches, which reproduce the entire current–voltage characteristics, these models avoid the need for detailed parameter identification and complex computations [1] [2] [3]. Although circuital representations may enhance accuracy at the device level, their higher complexity does not necessarily yield substantial benefits for large-scale energy forecasting. For this reason, simplified proportional models relying on irradiance and temperature are generally regarded as the most effective trade-off between accuracy and computational efficiency [4].

Since deterministic models require manual setup and detailed knowledge of the characteristics of the plant—which are not always available—this study adopts probabilistic models based on machine learning. It is considered a scenario in which such characteristics are entirely unknown, and only a small quantity of measured data is available. The goal is to understand the behavior of the Politecnico di Torino campus plant by relating Global Tilted Irradiance (GTI), G, and ambient temperature, T_a, to the produced power, P_{AC}. These insights will then be helpful in further developing such systems.

Typically, research papers present the methodology before the case study. In this work, however, the opposite approach was adopted: the plant characteristics and the available data are described first, followed by the methodology, including the data preprocessing steps. This order is designed to present a workflow tailored to the type of data available in the case study. By first outlining the context, as well as the nature, quality, and length of the dataset, the reader gains a more intuitive understanding of the data-processing pipeline, which ultimately prepares the data for use as input to probabilistic models. The whole paper is then organized as follows. Section II discusses some of the more recent papers on the topic. Follows Section III, which reports the case study, and Section IV, which describes the proposed methodology. Subsequently, Section V presents the experimental results, while Section VI draws the conclusions.

II. Probabilistic Models for Forecasting the Photovoltaic Production Profile

In [5], two different Machine Learning (ML) approaches, Gaussian Process Regression (GPR) and Support Vector Machine (SVM), were tested to forecast the PV power using data coming from [6]. The time of day, solar PV panel temperature, solar flux, relative humidity, and ambient temperature were used as input variables. Matern 5/2 GPR obtained the best results, with a Root Mean Square Error (RMSE) of 7.967 kW, a Mean Average Error (MAE) of 5.3025 kW, and a Coefficient of Determination (R^2) of 0.98.

Another research on PV production is presented in [7]. Two PV systems, one located inside a research laboratory and the other on the roof of Qatar University, were used to acquire the needed measures for the task. Two ML algorithms, M5P regression tree [8] and Simple Linear Regression [9], were evaluated using RMSE, MAE, Root Mean Absolute Error (RMAE), Root Relative Squared Error (RRSE), and Relative Absolute Error (RAE). In the end, the first one turned out to be the most performing, at the expense of greater complexity.

In [10], Regression Tree (RT), Support Vector Regression (SVR), and Artificial Neural Network (ANN) ML methods were tested for day-ahead PV power predictions. Meteorological and PV operational datasets were acquired over a year at the University of Cyprus (UCY) external testing facility. RMSE, MAE, Mean Absolute Percentage Error (MAPE), and Skill Score (SS) were used to assess the obtained performance. The ANN achieved the best performance, obtaining a MAPE of 0.61%, an RMSE of 10.37W, an nRMSE of 0.76%, and a SS of 92.22%.

Two other popular ML algorithms, Decision Trees (DT) and Random Forest (RF), were tested in [11], always for PV power prediction, using a small dataset of which nine features were considered, including temperature, irradiance, Direct Current (DC) power, and DC voltage. In general, RF was capable of achieving the best results.

A more comprehensive work is [12], where the authors designed a trained Dynamic Bayesian Network (DBN) for forecasting the power production of a 40 MW PV plant composed of 182,880 solar panels. The training dataset contains records from the 10^{th} May 2020 to 9^{th} May 2021 (1 year) meteorological data, operational indicators, sensor data such as inverters temperatures, number of faulty inverter modules, inverter on/off status, curtailment (impossibility to deliver the produced power due to grid saturation) status and historical power data. Considering a one-day-ahead forecasting with a 1-month rolling window, a 69.0397 kW MAE, 137.4292 kW RMSE, 0.0603 Normalized Root Mean Square Error (NRMSE), and 0.94 R^2 were achieved.

III. Case Study

A. Photovoltaic System Description

This photovoltaic installation represents the most significant plant on the Politecnico di Torino campus, with a total installed capacity across multiple rooftops of approximately 1 MW. This system alone contributes 604 kW. The system was commissioned in 2015 and consists of 1,849 modules with a nominal efficiency of 21.1%. The overall installed capacity is 604 kW, with the modules mounted on a shed-type roof at a tilt angle of 30°, oriented southwest with an azimuth of 30° west of south, as shown in Fig. 1. The modules exhibit a thermal power loss coefficient of 0.38%/°C and a Nominal Operating Cell Temperature (NOCT) of 45 °C, in accordance with the specifications of the manufacturer. The system employs a distributed inverter architecture, featuring multiple DC/AC converters rated at 25 kW each. The plant is equipped with a dedicated monitoring infrastructure, including calibrated reference solar cells and three-phase energy meters for the collection of the plant's total production.

Fig. 1. View of the photovoltaic system installed on the rooftop of the Politecnico di Torino canteen.

B. Weather data measurements

The meteorological data are collected from a weather station located on the rooftop of Politecnico di Torino. The system measures global, diffuse, and direct irradiance on the horizontal plane. Global irradiance is monitored with a Class A pyranometer (ISO 9060 standard) operating in the spectral range 300–3000 nm, with a directional error below ±20 W/m^2 at 1000 W/m^2 and an overall daily uncertainty within ±5% at 95% confidence. Diffuse irradiance is acquired through a second pyranometer equipped with an occultation device to remove the beam component, while direct irradiance is measured by a pyrheliometer aligned with the Sun by an automated tracker. The entire setup is periodically calibrated (every 2–3 years), and the tracking error on the incidence angle remains typically below ±5%. Measurements are sampled every minute, averaged over 15-minute intervals, and stored in a local SQL database for further processing.

IV. Methodology

A. Problem Analysis

The problem discussed in this paper is in the form of an n-step prediction model expressed in the form:

$$\mathbf{y}_{t:t+n} = f(\mathbf{x}_{t:t+n}) \tag{1}$$

where:

- $f(\cdot)$ is the prediction function learned by the probabilistic model;
- n is the number of time steps to be predicted;
- $\mathbf{x}_{t:t+n} = [x_t, x_{t+1}, \ldots, x_{t+n}]$ are observations of the targets inputs (GTI G and ambient temperature T_a) over a look-back window k;
- $\mathbf{y}_{t:t+n} = [y_t, y_{t+1}, \ldots, y_{t+n}]$ is the model forecast power production P_{AC} from the plant. Given the relations, the model description can be simplified as follows:

$$\mathbf{P}_{AC} = f(\mathbf{G}, \mathbf{T}_a) \tag{2}$$

The model is trained on a dataset composed of past historical data of length $h > n$.

B. Preprocessing

First, the data was processed to ensure its full usability with probabilistic models. The first step involved a preliminary data correction. Then, to best balance the datasets and conduct reliable tests for different periods of the year, production variations were taken into account on a monthly basis. Subsequently, the data was organized into sequences suitable for the Recurrent Neural Network (RNN) models considered, eventually augmented, and finally normalized.

1) Preliminary data analysis and correction: The dataset time window considered for the task is from October 2022 to September 2025. An initial check was performed to ensure that the GTI was zero in the absence of natural light. For all available days, during the hours without natural light, which in Italy can reasonably be between 11:00 PM and 4:00 AM UTC (Coordinated Universal Time), the GTI values were set to exactly zero. The 15-minute data was then averaged over an hour to provide hourly granularity.

2) Dataset Balance and Monthly Variability: The available dataset was first grouped by the number of full days, which were then split into training, validation, and testing sets with respective percentages of 80%-10%-10%. This split was not randomized across the entire dataset; instead, a split approach based on monthly data was chosen to perform balanced tests and account for variability in seasonal and monthly energy production. Due to connection issues, blackouts, and other reasons, the dataset used for this work has some "holes": that is, it does not contain all the data for all the days present in the interval considered. For this reason, only 602 24-hour sequences are available, which is less than two years. In detail, there are 179 days available in the spring season, 87 days in summer, 111 days in fall, and 225 days in winter. It is worth noting that the winter season, which is typically the period of lowest production in Italy, presents data almost three times higher than the summer season, in which production tends to be higher. The dataset is therefore strongly unbalanced. Subsequently, the days were split between the train, validation, and test datasets based on the aforementioned split percentages.

3) Data Augmentation: After splitting the dataset into training, validation, and testing sets, the training dataset was either used as is or subjected to Data Augmentation (DA) techniques to overcome data limitations partially. Simple stochastic augmentation techniques were employed, including the injection of Gaussian noise and the application of multiplicative scaling factors to the irradiance–power relationship. These perturbations aimed to increase the variability of the training set, thereby facilitating more stable convergence of the probabilistic models.

4) Normalization: The variables of interest were normalized to fall within the range of 0 to 1. The normalization is defined as follows:

$$x_{i\_norm} = \frac{x_i - min(x)}{max(x) - min(x)} \tag{3}$$

where x_{i_norm} is the obtained i^{th} normalized value, x_i is the original i^{th} value, $min(x)$ is the minimum value in the dataset, and $max(x)$ is the maximum value in the dataset.

C. Probabilistic Models

Seven different architectures were considered: Vanilla RNN, LSTM [13], GRU [14], BiLSTM [15], BiGRU, minLSTM [16], and minGRU. All of them are very suitable for the task at hand because they are architectures belonging to the family of RNNs, which, unlike normal Feedforward Neural Networks (FNN) or other similar networks, have the characteristic of being able to recur on the same neuron and, therefore, to memorize past inputs in order to make better predictions. While LSTM is a well-established architecture, GRU is a more recent and simplified version, which is, for this reason, usually better suited for analyzing shorter time series. Newer mathematical formulations for LSTM and GRU were presented in [16]. Such versions, called minLSTM and minGRU, have the following key changes concerning their counterpart:

- Hidden state dependencies, i.e., dependencies on previous time steps, have been removed, transforming the process from sequential to parallel.
- Functions for limiting the range of output values in traditional models, such as tanh and sigmoid, have been replaced by linear transformations, making calculations faster and easier.
- Training has been made more stable and efficient on long sequences by not scaling the output based on the sequence length.

These changes made minGRU and minLSTM significantly faster to train and less memory-intensive, while maintaining results comparable to or better than their classical counterparts and in line with more demanding alternatives, such as Transformers [17] and Mamba [18]. BiLSTM and BiGRU are the bidirectional versions of the LSTM and GRU architectures, which means that they process the input sequence not only from the bottom to the top of the network, but also in the opposite direction.

Since minLSTM and minGRU are novel approaches, it was also decided to use the classical LSTM and GRU models to compare the results obtained by the reduced and classical versions in this particular task.

The seven different models were tested, considering $n = 24$, which corresponds to a full day. The following structure has been adopted for all models:

- Stacked m-layer RNN each having a hidden layer of k units and a dropout of $d\%$, where the number of stacked layers m was set from 1 to 4, the size k of the hidden layers were powers of two from $2^4 = 16$ to $2^9 = 512$, and the dropouts d were from 0% to 30% in 5% increments.
- 1 Fully Connected (FC) layer having the same size as the hidden layers neurons, doubled in the case of BiLSTM and BiGRU.

D. Hyperparameters

The following hyperparameters were tested to train the models:

- Learning Rate (LR): 0.5, 0.25, 0.1, 0.075, 0.05, 0.025, 0.01, 0.0075, 0.005, 0.0025, 0.001, 0.0005, 0.0001
- Batch Size (BS): 256, 128, 64, 32, 16, 8.
- Number of epochs: 2000.
- Optimizer: Nadam [19].
- Loss Function: Huber.

After a preliminary analysis of the LR, it was noted that the learning curve was better, and training progressed better, with a learning rate between 0.1 and 0.01. For this reason, several experiments were performed by increasing the LR values around these values. The Huber loss function combines the characteristics of RMSE and MAE, thus penalizing large errors like the former, while being robust to outliers like the latter. Its mathematical formulation is as follows:

$$L_\delta(r) = \begin{cases} \frac{1}{2}(y_i - \hat{y}_i)^2 & \text{if } |y_i - \hat{y}_i| \leq \delta, \\ \delta(|y_i - \hat{y}_i| - \frac{1}{2}\delta) & \text{otherwise.} \end{cases}$$

where y_i is the true observed value at time i, $\hat{y}_i$ is the predicted value at time i, and δ is a positive threshold parameter, controlling the loss transitions from quadratic to linear.

Having a large number of epochs significantly increases training time and, if the network stops learning, may even be unnecessary. To overcome these problems, it was decided to implement an early stopping strategy with the following parameters:

- Metric monitored: Validation Loss.
- Patience (number of epochs without improvement before the training is stopped): 15.
- Delta (minimum performance increase considered meaningful from the baseline): 0.0001.

An LR reduction strategy has also been implemented to try to continue training in case the Validation Loss is not improving:

- Metric monitored: Validation Loss.
- Mode: minimum (the LR is reduced once the monitored metric stops decreasing).
- Factor (proportional reduction factor applied to the LR): 0.5.
- Patience (number of epochs without improvement before the LR is decreased): 4.
- Cooldown (minimum number of epochs to wait before executing another LR reduction): 2.
- Threshold (sensitivity threshold for identifying meaningful updates to the optimal value): 0.0001.

To ensure maximum reproducibility between the different tests performed, a fixed seed was set for the main libraries used: numpy [20], pandas [21], [22], and pytorch [23].

E. Indicators

Several Key Performance Indicators (KPIs) were used to evaluate the performance of the probabilistic models:

- RMSE, which can be obtained by taking the square root of the Mean Squared Error (MSE). It penalizes larger errors more, preventing them from dominating. Its mathematical formulation is as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

  where n is the number of samples (time steps).
- MAE, which measures the average magnitude of the error in a set of predictions. It provides a simple and intuitive measure of how far predictions tend to be from the true values. The lower it is, the better the model performs. It is defined as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- MBE, which indicates whether the model systematically underestimates (positive MBE) or overestimates (negative MBE) the observations. It is essential to note that an MBE of zero does not guarantee accurate model performance, as it merely indicates that the errors tend to cancel out on average. It is defined as follows:

$$\text{MBE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)$$

- R^2, which indicates how well the model fits the data. It ranges from 0 to 1, where zero indicates that the model performs no better than the average prediction, while one indicates that it makes perfect predictions. It is defined as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

  where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ is the mean of the observed values.

V. Experimental Results

The results obtained are presented in Tables I and II, which displays direct and clipped outputs, i.e., those whose value is constrained between the minimum and maximum energy production values observed during training. The minimum value is set to zero, as energy production is assumed to be always positive. For the clipped values, the upper and lower values returned by the networks are therefore reported within

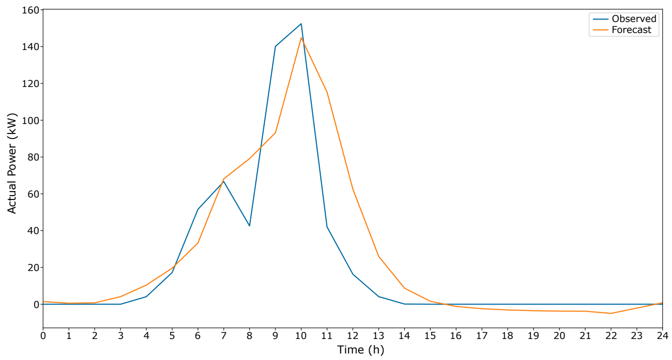

Fig. 2. Predicted LSTM measurements compared to those observed on a cloudy day, without clipping the night hours.

TABLE II
COMPARISON OF THE BEST KPI OBTAINED WITH THE PROBABILISTIC MODELS CONSIDERED USING CLIPPED OUTPUTS

	Clipped Outputs			
Model	**RMSE (kW)**	**MAE (kW)**	**MBE**	**R^2**
Vanilla RNN	52.663	28.224	4.748	0.747
LSTM	**38.534**	**19.32**	**0.095**	**0.875**
GRU	42.852	21.349	0.429	0.853
BiLSTM	43.856	21.408	2.641	0.824
BiGRU	41.281	21.862	2.387	0.845
minLSTM	46.826	23.565	4.066	0.8
minGRU	49.761	28.396	-0.247	0.77

this range. What can be observed is that this effect generally tends to have positive effects on some metrics, but at the same time, it tends to degrade others, particularly the MBE. Therefore, for completeness of information, Tables I and II report both cases. Examining the results, it is evident that LSTM outperformed the other models for all KPIs of interest, followed by GRU and BiGRU.

The hyperparameters used to obtain the best KPI for each probabilistic model are reported in Table III. It can be seen that training benefited mainly from a low BS, especially for a value equal to 16, except in two cases. Furthermore, only a small number of cases have benefited from DA, so this aspect should be further investigated.

An example of a cloudy day forecast performed with the best trained probabilistic model, the LSTM, is shown in Figure 2. It can be seen that the model provides a reliable representation of daily production, even in the harsh case of a cloudy day. However, it lacks accuracy in tracking the actual trend of daily production.

VI. CONCLUSIONS

This research work was useful for assessing the behavior of the probabilistic models, particularly RNNs, to describe the PV power production when trained using a small amount of data. The results obtained are good, but they do not seem sufficient to describe physical behavior as accurately as possible. To address this issue, statistical DA techniques have been attempted, but the results do not demonstrate significant benefits. In the future, the use of these DA techniques could be further explored by attempting to use existing data to generate more plausible and physically based perturbations, such as algorithms to correlate power and irradiance better, estimate cloud passage, estimate the temperature of the plant site based on time and year, and more. This solution comes at the cost of losing the ability to forecast power data without knowing the physical characteristics of the plant, such as its geographic location and panel inclination. Additional input variables could also provide more insight into the time of year, passing clouds, and other factors, allowing networks to better adjust for seasonal variations. Future work will continue to explore this research theme in greater detail and will benefit from the findings presented here.

REFERENCES

[1] S. Prakash *et al.*, "Modeling and performance analysis of simplified three-diode photovoltaic module," *Journal of Electrical Engineering*, vol. 1, pp. 55–64, 2022.

[2] V. R. Lo Verso, S. Invernizzi, A. Carlin, and A. Polato, "Towards the factory of the future: A new concept based on optimized daylighting for comfort and energy saving," in *2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC)*, 2015, pp. 701–706.

[3] A. Carlin and V. R. M. Lo Verso, "Is lighting based on photopic requirements in assembly workstations also optimal for non-visual performance in a human-centric perspective? a case study in the automotive sector," in

TABLE I
COMPARISON OF THE BEST KPI OBTAINED WITH THE PROBABILISTIC MODELS CONSIDERED

	Normal Outputs			
Model	**RMSE (kW)**	**MAE (kW)**	**MBE**	**R^2**
Vanilla RNN	52.779	28.898	4.074	0.746
LSTM	**38.602**	**19.50**	**0.084**	**0.864**
GRU	42.879	21.92	-0.142	0.852
BiLSTM	43.856	21.414	2.635	0.824
BiGRU	41.467	23.33	0.919	0.843
minLSTM	46.831	23.69	3.94	0.8
minGRU	49.763	28.429	-0.279	0.774

TABLE III
COMPARISON OF THE HYPERPARAMETERS USED TO OBTAIN THE BEST PROBABILISTIC MODELS

Model	Train Loss	Val Loss	LR	BS	SL #	HS	Dropout	DA
Vanilla RNN	0.31527	0.2428	0.075	16	1	128	0	N
LSTM	**0.06458**	**0.06835**	0.0025	16	2	128	0.2	N
GRU	0.07182	0.07483	0.1	32	2	256	0	N
BiLSTM	0.13462	0.1528	0.05	16	2	128	0.15	Y
BiGRU	0.14299	0.17153	0.075	16	1	128	0	N
minLSTM	0.20751	0.22478	0.075	16	2	256	0.15	Y
minGRU	0.31527	0.2428	0.075	16	2	256	0	N

International Journal of Mechanics and Control, vol. 26, no. 01, 2025, pp. 185–212.
[4] A. Ciocia, G. Chicco, and F. Spertino, "An improved model for ac power from grid connected photovoltaic systems and comparison with large-scale hourly measured data," *IEEE Transactions on Industry Applications*, vol. 59, no. 4, pp. 4427–4437, 2023.
[5] B. Zazoum, "Solar photovoltaic power prediction using different machine learning methods," *Energy Reports*, vol. 8, pp. 19–25, 2022, 2021 The 8th International Conference on Power and Energy Systems Engineering.
[6] V. Omubo-Pepple, C. Israel-Cookey, and G. Alaminiokuma, "Effects of temperature, solar flux and relative humidity on the efficient conversion of solar energy to electricity," *European Journal of Scientific Research*, vol. 35, pp. 173–180, 10 2009.
[7] K. Benhmed, F. Touati, M. Al-Hitmi, N. A. Chowdhury, A. S. Gonzales, Y. Qiblawey, and M. Benammar, "Pv power prediction in qatar based on machine learning approach," in *2018 6th International Renewable and Sustainable Energy Conference (IRSEC)*, 2018, pp. 1–4.
[8] J. R. Quinlan, "Learning with continuous classes," in *5th Australian Joint Conference on Artificial intelligence*, vol. 92, 11 1992, pp. 343–348. [Online]. Available: https://api.semanticscholar.org/CorpusID:1056674
[9] X. Yan and X. G. Su, *Linear Regression Analysis: Theory and Computing*. USA: World Scientific Publishing Co., Inc., 2009.
[10] S. Theocharides, G. Makrides, G. E. Georghiou, and A. Kyprianou, "Machine learning algorithms for photovoltaic system power output prediction," in *2018 IEEE International Energy Conference (ENERGYCON)*, 2018, pp. 1–6.
[11] R. P. Ponraj, V. Ravindran, K. Chittibabu, A. T, C. A, and E. Louis, "Machine learning based solar pv power prediction," in *2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)*, 2023, pp. 1646–1652.
[12] Q. Zhang, H. Yan, and Y. Liu, "Power generation forecasting for solar plants based on dynamic bayesian networks by fusing multi-source information," *Renewable and Sustainable Energy Reviews*, vol. 202, p. 114691, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032124004179
[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014. [Online]. Available: https://arxiv.org/abs/1412.3555
[15] M. Schuster and K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
[16] L. Feng, F. Tung, M. O. Ahmed, Y. Bengio, and H. Hajimirsadeghi, "Were rnns all we needed?" 2024. [Online]. Available: https://arxiv.org/abs/2410.01201
[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
[18] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," 2024. [Online]. Available: https://arxiv.org/abs/2312.00752
[19] T. Dozat, "Incorporating Nesterov Momentum into Adam," in *Proceedings of the 4th International Conference on Learning Representations*, 2016, pp. 1–4. [Online]. Available: https://openreview.net/pdf/OM0jvwB8jIp57ZJjtNEZ.pdf
[20] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. [Online]. Available: https://doi.org/10.1038/s41586-020-2649-2
[21] The pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134
[22] Wes McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference*, Stéfan van der Walt and Jarrod Millman, Eds., 2010, pp. 56 – 61.
[23] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, *PyTorch: an imperative style, high-performance deep learning library*. Red Hook, NY, USA: Curran Associates Inc., 2019.