COMPARATIVE EVALUATION OF MACHINE AND DEEP LEARNING ALGORITHMS FOR SOLAR RADIATION PREDICTION

Mohammed El-Amine Azz[1] **Raza Imam**[2] **Mohammed Talha Alam**[3] **Mohammad Al Zarooni**[4]
[1,2,3,4]MBZUAI, Abu Dhabi, United Arab Emirates
{[1]mohammed.azz, [2]raza.imam, [3]mohammed.alam, [4]mohammad.alzarooni}@mbzuai.ac.ae

Abstract

Given the concerns over environment degradation and the depletion of fossil fuel supplies, the best long-term strategy is to transition to solar energy. So, for solar-powered devices, precise solar radiation forecasting is crucial. To produce predictions about the typical daily sun radiation, we investigate different machine and deep learning models (ARIMA, Exponential Smoothing, LSTM, and Neural Prophet). We experiment with predicting one day at a time using ground truths as well as predicting multiple days ahead, although former is a good approach with good predictive power, the later is very difficult due to the unpredictable nature of solar radiation due to the different climatic factors. The performance of each model is evaluated using metrics such as MAPE and RMSE in this univariate time series analysis using the radiation parameter. For the purpose of prediction without ground truth for the future, among all models Neural Prophet seems to provide the best results overall in terms of both RMSE and MAPE. Finally, evaluated assessment of this work provides a basis for accurate estimation of solar irradiance, which can promote further development and administration of solar power systems.

1 Introduction

Solar energy has a wide range of possible applications and is a major, clean, and sustainable source of energy [1]. The development and evaluation of solar-powered systems, environmental research, the treatment of water resources, the estimation of crop yields, and so on. all depend on solar irradiation. As a result, accurate methods may be built to calculate solar radiation using more data [2]. The consistency of solar irradiation and its use is constrained by elements like seasons, geography, cloud density, and other environmental conditions. Volatility and inconsistency are inherent properties of solar irradiation. Because of this, resource planners must be flexible in order to account for these inconsistencies while undertaking planning, which is crucial for the development and administration of solar power systems. Also, the optimum prospective is to switch to solar energy in the long run given the concern for environmental damage and the depletion of fossil fuel reserves. Therefore, accurate solar radiation forecasting is essential for solar-powered systems that are built to generate huge amounts of energy to regulate the supply and demand of energy. The need for solar irradiance estimation and its sensitivity has spurred numerous scientists to develop practical solutions [3], [4].

1.1 Background and Motivation

Solar irradiance is generally measured using SR measurement instruments. These methods, nevertheless, have large implementation expenses and calibration demands. As a result, they are not obtainable at the majority of stations globally. That is why it is extremely important to estimate solar radiation data using accurately manageable meteorological variables like moisture, heat, air velocity, weather patterns, and many others. Numerous studies in this field have been inspired to find efficient strategies due to the importance of solar irradiance forecasting as well as its sophistication.

Code available at [Github]

Several models have been developed to forecast solar radiation observations from this perspective. Many among them are empirical models because they are built on mathematical concepts. Empirical models are simple to compute and largely acknowledged as a method for predicting solar radiation data [5]. Despite the fact that empirical methods have been extensively used to forecast the mean global solar radiation, these approaches are not capable of forecasting short-term solar irradiance with any degree of accuracy because of the rapid shifts in meteorological conditions.

Along with empirical studies, other AI techniques including support vector machines (SVM), deep learning (DL), K nearest neighbors (KNN), artificial neural networks (ANN), etc., emerged to be widely used methods in the forecasting of solar irradiance. According to prior studies, AI algorithms have produced outcomes that are more precise than those produced by empirical approaches when it comes to forecasting solar radiation [6], [7]. Despite the fact that there are numerous meteorological sites across the globe, the great majority of those lack the means to monitor solar radiation data since the equipment, administration, and calibration of such instruments are expensive [8]. As a result, it is important to forecast solar radiation data using specific variables that are simpler and more efficient to monitor in a specific area.

2 Literature Review

There are a number of studies that employ a unique model to determine and forecast solar radiation in the literature. Such models are based on machine learning, statistics, or empirical research. The brief literature review focuses on statistical and machine learning forecasting approaches as there is an increasing demand for research on machine learning techniques in several scientific domains.

Brahim Belmahdi et al. (2022) suggested an algorithm method in order to select the best machine learning approaches and time series models that reduce predicting errors. They discovered that ARIMA and Feedforward neural networks (FFNN) score higher and provide accurate approximations when used with the matching GSR output [9]. Recent research (2022) also proposed a method employing neural networks to forecast solar radiation by collecting meteorological data from five different Bangladeshi cities. Predictions were done separately for each of the five cities using Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) networks. The GRU model was reported to have produced the best outcome of the three models [10].

In another study [11] (2021), the authors suggested using deep learning clustering to learn sun irradiance characteristics. For each cluster, a Feature Attention Deep Forecasting (FADF) deep neural network was designed to produce forecasts for global horizontal irradiance (GHI). The proposed strategy outperformed smart persistence and the latest models in simulations, according to the results, in terms of solar predicting accuracy. Similarly, a multi-modal fusion was developed by Ajith et al. [12] network to analyze solar irradiance micro predictions utilizing both infrared scans and prior sun irradiance data. The suggested approaches' solar forecasts were compared to benchmark models using the Mean Absolute Percentage Error (MAPE) as well as other qualitative metrics. The experimental findings showed that multi-modal fusion networks perform better than conventional techniques at forecasting sun irradiance for both overcast and mixed days.

Besharat et al. (2013) analyzed several global solar radiation models that are currently available and divided them into four categories: models that are based on sunshine, clouds, temperature, and other meteorological parameters. After doing their research, they came to the conclusion that sunlight models provide the best accuracy when compared to other investigated methods [13]. The work that has just been discussed makes it abundantly explicit that numerous probabilistic and AI models are feasible and can be successfully integrated.

3 Methodology

3.1 Dataset

The dataset used in this work has been compiled by NASA from the HI-SEAS weather station. It was a part of the Space Apps Moscow tournament. The dataset is available on multiple sources and can be found on Kaggle at [14]. The initial features of the dataset are UNIX time, date in yyyy-mm-dd format, solar radiation in watts per meter squared, Temperature in degrees Fahrenheit, percentage humidity, Barometric pressure in Hg, wind direction in degrees, wind speed in miles per hour, and lastly sunrise and sunset in Hawaii time with 32686 observations from 1 September 2016 to 30 December 2016, with an average observation interval of 5 minutes. As there were multiple readings throughout the day, we decided to take the daily averages for the purposes of testing. Also, we had data for the following dates, which are: 30 September 2016, 30 November 2016, 6 December 2016, and 7 December 2016. We decided to interpolate these values for radiation to do our testing.

Date	Radiation	Temperature	Pressure	Humidity	WindDirection(Degrees)	Speed
2016-09-01	298.451600	54.556000	30.436640	78.968000	112.564000	6.396560
2016-09-02	333.471219	56.379928	30.446487	72.501792	115.392079	5.804086
2016-09-03	123.974574	55.507092	30.438546	94.599291	192.678546	4.960248
2016-09-04	153.584321	53.589286	30.454107	93.878571	126.336714	5.184571
2016-09-05	306.065125	53.199288	30.419680	79.943060	209.169253	5.830676

Figure 1: Sample head of our experimented dataset

The organized data contains 6 climate features (Radiation, Temperature, Pressure, Humidity, Wind Direction, and Wind Speed), with 121 observations from 1 September 2016 to 30 December 2016. Figure 1 also shows head samples from the organized dataset, whereas Figure 2 shows the relative trends of the radiation feature for different time intervals. We used "seasonal.decompose" to visualize the trends and we concluded that the radiation data has a weekly seasonality.

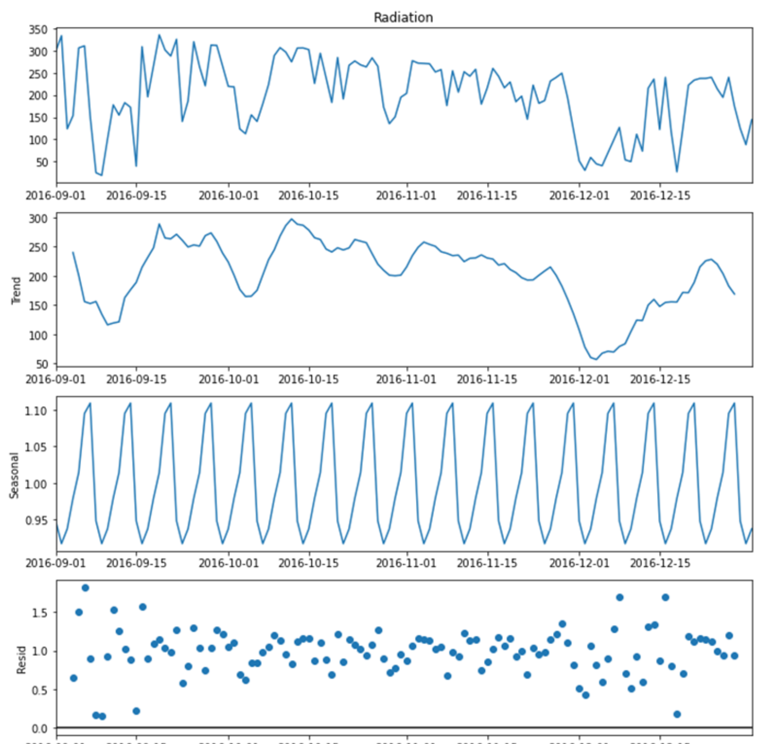


Figure 2: The first plot showcases the daily radiation. The next 2 plots showcase the general trends and seasonal trends. From the plots, we see that the lowest points occurred during the first half of September and December. The residual plot also shows more variability during those time frames. From the seasonality plot, we see a repeating pattern in regular intervals, which could be due to weekly seasonal factors.

3.2 Methods

3.2.1 ARIMA

ARIMA models (Auto Regressive Integrated Moving Average), is a class of models that interprets given time series based on the past values. It is one of the most widely used time series forecasting approaches alongside Exponential Smoothing. This model deals well with non-seasonal time series that exhibit patterns [8]. Generally, the ARIMA model is characterized by 3 parameters: p, d, q. "p" is the order of the 'Auto Regressive' term, which is the number of lags (previous Ys) that are used as predictors. 'q' is the order of the 'Moving Average' term, which refers to the number of lagged forecast errors that should go in the model. 'd' represents differencing, where its purpose is to make the time series stationary. As such, we can represent the ARIMA equation as follows:

$$Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + .. + \beta_p Y_{t-p} \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + .. + \phi_q \epsilon_{t-q}$$

3.2.2 Exponential Smoothing

Exponential Smoothing is a forecasting approach used for univariate time series data. It produces forecasts using weight averages of past observations. These weighted averages, unlike ARIMA, exponentially decrease for older observations. There are forms of these models that extend to analyzing data with trends and seasonal components. We can adjust the model's parameter values which help us change the rate of how quickly older observations lose their weight in the calculations. It also allows tweaking the relative importance of new observations to the older ones for any area's requirements. Different types of Exponential smoothing include SES, DES, and Holt-Winters method [15].

3.2.3 LSTM

LSTM (Long Short-Term Memory) Network is an RNN-trained model using Backpropagation through time to tackle the vanishing gradient problem [16]. It uses memory blocks connected through layers instead of neurons. A block contains gates that manage the state and output and operates on an input sequence. Each gate is triggered using the sigmoid activation units which make the flow of information conditional. The 3 types of gates within a unit are the Forget gate, which conditionally throws parts of the information away from the block, the Input gate which conditionally decides input value is updated, and the Output gate which decides the output based upon the input and memory of the block. The gates of the units have weights that are learned during training. For the input in our simple model, we will use one lookback input which mimics using the information of only one day before. LSTM is sensitive to scaling so before training the model, values are scaled, and then unscaled after forecasting.

3.2.4 Prophet and Neural Prophet

Prophet is a Facebook open-source software used in forecasting time series data [17]. It is based on additive models in which non-linear trends are fit with yearly, weekly, daily, and holiday effects. It performs very well with time series that have seasonal effects, such as temperatures. Unlike some models, Prophet can handle missing data, outliers, and shifts in trends well. The models are fit in Stan, so the forecasting is very fast to get. In addition, this procedure allows for tweaking and adjusting the forecasts. For example, we can use human-interpretable parameters, improving the forecast with additional information. The model in its core comprises of the sum of 3 functions and an error term [18]. The 3 functions are growth, seasonality, and holidays. Adding the error term, the model equation is represented as:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

Neural Prophet is the new version of the Prophet model by Facebook [19]. The main differences and improvements are the usage of gradient descent for optimization via PyTorch, using AR-NET to model autocorrelation of time series, using FFNNs (Feed-Forward Neural Networks) to model lagged regressors and configure non-linear deep layers, being tuneable to specific forecast horizons and lastly custom metrics and losses For the testing part of this paper, we will be using the new version Neural Prophet.

4 Results and Discussions

For the time series forecasting, we split the data into 2. The training set has daily observations from the 1st of September to the 20th of November, while the test set contains from the 11th to the 31st of December. We fit 4 models (ARIMA, Exponential Smoothing: Holt-Winter, LSTM, and Neural Prophet). For evaluations, we resorted to the testing MAPE since it is scaled invariant along with RMSE. We used 3 different evaluation phases which are, Training, where we evaluate the model prediction against the training data, Testing, where we evaluate the model predictions using ground truths as inputs against the testing set, and lastly, the 7days testing where we used allowed models to predict the next 7 days independent of the inputs from the testing data. Firstly, we fit the ARIMA model with auto_arima in Python. This allowed the model to test different parameters for p,q,d to improve the results which resulted in (1,0,1) parameters.

4.0.1 Training

We trained the 4 models on the first 2 thirds of the data. After fitting the models, we applied them to the training set. In ARIMA, the model followed the trends with a shrink. Computing the MAPE and RMSE, we obtained 40.50 and 62.49 respectively. Exponential Smoothing seemed to lag behind the real values however the values were not shrunk like ARIMA. The training MAPE and RMSE for this model are 41.58 and 67.96 respectively. Training Wise, LSTM performed the same as ARIMA in terms of the predicted values, however, due to the look back used in LSTM, the training set used for evaluation is missing the first and last days, leading to a training MAPE of 40.39 and RMSE of 62.35. As for Neural Prophet, we achieved the best MAPE and RMSE of 37.38 and 55.73 respectively. The prediction plots are shown in Figure 3.

Figure 3: Training set model predictions.

4.0.2 Testing With Ground Truth

We evaluated 3 of our 4 models (excluding Neural Prophet) on the testing set, always using the actual history as the input including from the testing set itself. In ARIMA we achieved a testing MAPE of 67.53 and a testing RMSE of 62.09. Exponential Smoothing performed better than ARIMA in this case with a testing MAPE and RMSE of 52.43 and 59.82 respectively. Lastly, LSTM performed the worst on the testing set with a testing MAPE of 80.81 and a testing RMSE of 69.37. The prediction plots are shown in Figure 4.

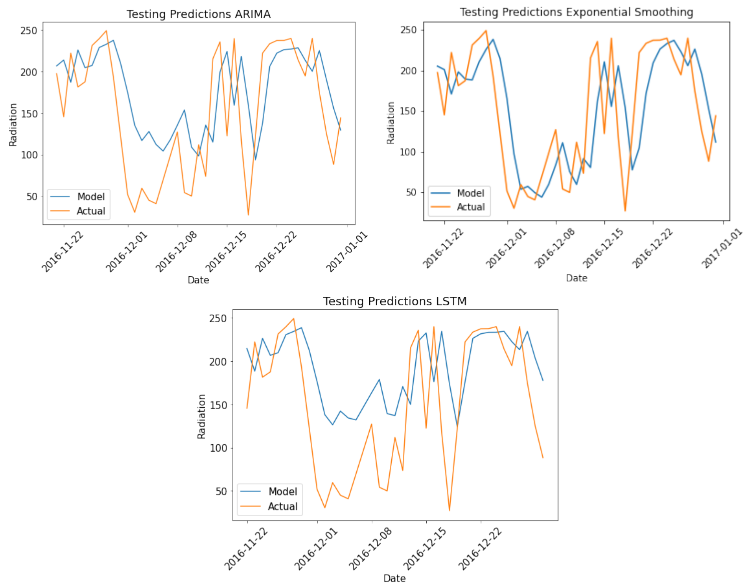


Figure 4: Testing set model predictions with ground truths.

4.0.3 Out of Sample Testing (7 Days)

In this test, we predict the next 7 days after the training set without the usage of ground truths from the testing sets in our predictions. In ARIMA, we obtained a MAPE of 15.44 and an RMSE of 36.19. For Exponential Smoothing, we obtained a horizontal line with a MAPE of 14.44 and an RMSE of 30.89. Similar in the trend to ARIMA, the LSTM model achieved a MAPE and an RMSE of 17.17 and 35.77 respectively. Lastly, the Neural Prophet model was the only model that did not converge into a flat line, yielding a MAPE of 13.52 and an RMSE of 34.89. The prediction plots are shown in Figure 5.

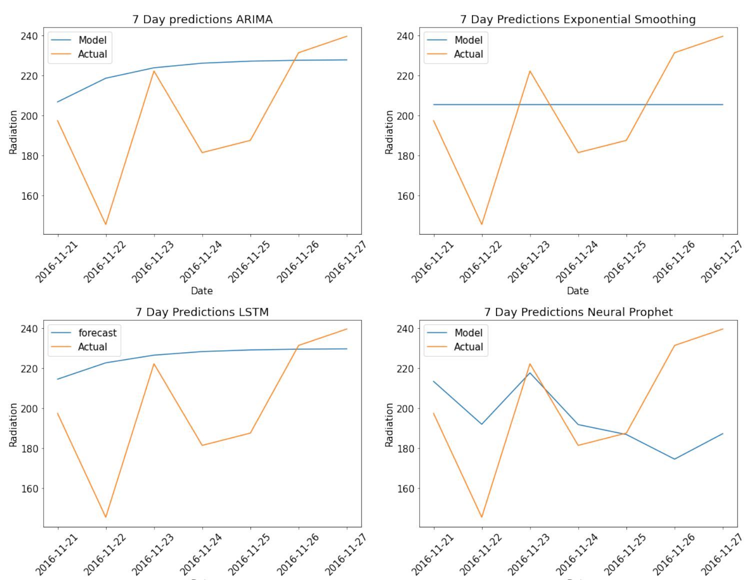
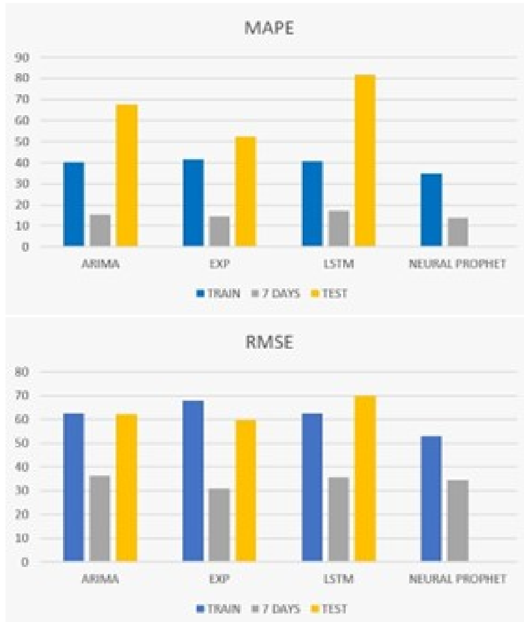


Figure 5: 7 days testing predictions with no test set ground truths.

4.0.4 Comparisons and Discussions

The Results obtained from the models were summarised in Figure 6.



Metric\Model	ARIMA	Exponential Smoothing	LSTM	Neural Prophet
Training MAPE (%)	40.50	41.58	40.39	**37.38**
Testing MAPE (%)	67.53	**52.43**	80.81	-
7 Days MAPE (%)	15.44	14.44	17.17	**13.52**
Training RMSE	62.49	67.96	62.35	**55.73**
Testing RMSE	62.09	**59.82**	69.37	-
7 Days RMSE	36.19	**30.89**	35.77	34.89

Figure 6: The first 2 plots represent the bar graphs for both MAPE and RMSE for each model for each type of test. The table summarises the numerical results with the best results being in bold.

Training-wise, the Neural Prophet model performed the best out of all models, however, we could not use it for testing using ground truth since the model does not need ground truth values for testing after the model is trained. For the testing with ground truths, the Exponential Smoothing Model performed the best out of the 3 models, while LSTM performed very poorly in comparison. The LSTM can be significantly improved by adding more inputs for the look-back, but that will shrink the training and testing sets even further. As for the 7 Days testing without the usage of ground truths, Neural Prophet achieved the best MAPE, but Exponential Smoothing had the lowest RMSE. Given that Exponential Smoothing results in a straight line, its value will diminish quickly with longer intervals as it does not add a lot of information as Neural Prophet can capture the decreasing trend if we increase the interval to be the full testing set.

We limited the testing without ground truths to 7 days, because ARIMA, Exponential Smoothing as well as LSTM quickly converge to a flat line which is not helpful in the longer term. The only model which can give a decent general trend for future prediction on large intervals would be Neural Prophet as it incorporates the weekly trends in the model. If we decide to increase the period to be the whole testing set, we can see the differences more clearly between Neural Prophet and Exponential Smoothing in Figure 7.

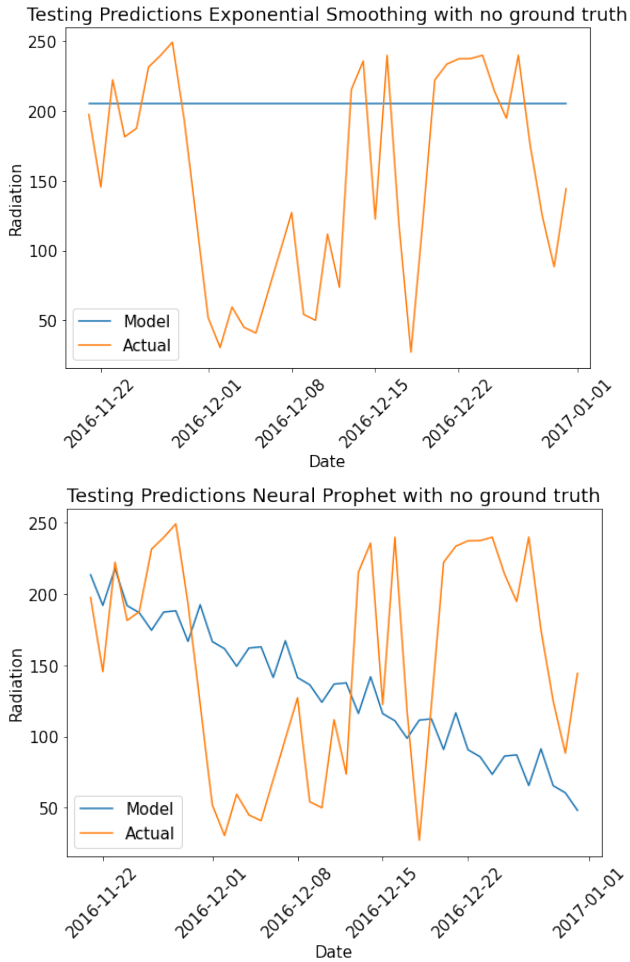


Figure 7: The 2 plots correspond to the model prediction without ground truth for the testing set for both Exponential Smoothing and Neural Prophet.

Although the RMSE of the Exponential Smoothing model was lower in the span of 7 days, it is outperformed by Neural Prophet in the long run (90.62 for Neural Prophet vs 90.97 for Exponential Smoothing). The predictive capability of the Exponential Smoothing model is not well to forecast a long period in the future without the use of ground truths from the test data. On the other hand, Neural Prophet performs better in the long run both in MAPE and RMSE while trying to follow the general trend of the data along with the weekly trends.

5 Conclusion

Predicting solar radiation is a very challenging problem using univariate time series analysis. While predicting one day at a time using ground truths is a good approach with good predictive power, predicting multiple days ahead is very difficult due to the unpredictable nature of solar radiation due to the different climate factors. In practice, predicting one day at a time would be feasible and better in areas where solar panels are used. However, predicting multiple days can be cost-efficient and does not require obtaining more readings of climate factors. For the purpose of prediction without ground truth for the future, Neural Prophet seems to provide the best results overall. For future work, we would like to explore multivariate time series where we could use multiple factors in parallels, such as wind and temperature, to give more predictability power to our models. We would also like to experiment with different time series with and without seasonal trends. We would also like to improve the LSTM model by increasing the inputs to achieve better results.

References

[1] Gang Liu. Sustainable feasibility of solar photovoltaic powered street lighting systems. *International Journal of Electrical Power & Energy Systems*, 56:168–174, 2014.

[2] Adel Mellit and Alessandro Massi Pavan. A 24-h forecast of solar irradiance using artificial neural network: Application for performance prediction of a grid-connected pv plant at trieste, italy. *Solar energy*, 84(5):807–821, 2010.

[3] Vahid Nourani, Gozen Elkiran, Jazuli Abdullahi, and Ala Tahsin. Multi-region modeling of daily global solar radiation with artificial intelligence ensemble. *Natural Resources Research*, 28(4):1217–1238, 2019.

[4] Na Dong, Jian-Fang Chang, Ai-Guo Wu, and Zhong-Ke Gao. A novel convolutional neural network framework based solar irradiance prediction method. *International Journal of Electrical Power & Energy Systems*, 114:105411, 2020.

[5] Ali Etem Gürel, Ümit Ağbulut, and Yunus Biçen. Assessment of machine learning, time series, response surface methodology and empirical models in prediction of global solar radiation. *Journal of Cleaner Production*, 277:122353, 2020.

[6] Yanfeng Liu, Yong Zhou, Yaowen Chen, Dengjia Wang, Yingying Wang, and Ying Zhu. Comparison of support vector machine and copula-based nonlinear quantile regression for estimating the daily diffuse solar radiation: A case study in china. *Renewable Energy*, 146:1101–1112, 2020.

[7] Hilmi Cenk Bayrakçı, Cihan Demircan, and Ali Keçebaş. The development of empirical models for estimating global solar radiation on horizontal surface: A case study. *Renewable and Sustainable Energy Reviews*, 81:2771–2782, 2018.

[8] Selva Prabhakaran. Arima model – complete guide to time series forecasting in python, 2022. Available at `https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/`.

[9] Brahim Belmahdi, Mohamed Louzazni, and Abdelmajid El Bouardi. Comparative optimization of global solar radiation forecasting using machine learning and time series models. *Environmental Science and Pollution Research*, 29(10):14871–14888, 2022.

[10] ANM Fahim Faisal, Afikur Rahman, Mohammad Tanvir Mahmud Habib, Abdul Hasib Siddique, Mehedi Hasan, and Mohammad Monirujjaman Khan. Neural networks based multivariate time series forecasting of solar radiation using meteorological data of different cities of bangladesh. *Results in Engineering*, 13:100365, 2022.

[11] Chun Sing Lai, Cankun Zhong, Keda Pan, Wing WY Ng, and Loi Lei Lai. A deep learning based hybrid method for hourly solar radiation forecasting. *Expert Systems with Applications*, 177:114941, 2021.

[12] Meenu Ajith and Manel Martínez-Ramón. Deep learning based solar radiation micro forecast by fusion of infrared cloud images and radiation data. *Applied Energy*, 294:117014, 2021.

[13] Fariba Besharat, Ali A Dehghan, and Ahmad R Faghih. Empirical models for estimating global solar radiation: A review and case study. *Renewable and Sustainable Energy Reviews*, 21:798–821, 2013.

[14] NASA. Solar radiation dataset, 2022. Available at `https://www.kaggle.com/datasets/dronio/SolarEnergy`.

[15] Jim Frost. Exponential smoothing for time series forecasting, 2021. Available at `https://statisticsbyjim.com/time-series/exponential-smoothing-time-series-forecasting/`.

[16] Jason Brownlee. Time series prediction with lstm recurrent neural networks in python with keras, 2022. Available at https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/.

[17] Benjamin Letham Sean J. Taylor. Forecasting at scale, peerj preprints 5:e3190v2, 2017. Available at https://doi.org/10.7287/peerj.preprints.3190v2.

[18] Mitchell Krieger. Time series analysis with facebook prophet: How it works and how to use it, 2021. Available at https://towardsdatascience.com/time-series-analysis-with-facebook-prophet-how-it-works-and-how-to-use-it-f15ecf2c0e3a.

[19] Serafeim Loukas. Neuralprophet for time-series forecasting: Predicting stock prices using facebook's new model, 2021. Available at https://pub.towardsai.net/neuralprophet-for-time-series-forecasting-predicting-stock-prices-using-facebooks-new-model-