Using Stacked Long Short Term Memory with Principal Component Analysis for Short Term Prediction of Solar Irradiance based on Weather Patterns

Engr. Justin D. de Guia
Electronics and Communications Engineering Department
De La Salle University
Manila, Philippines
justin_deguia@dlsu.edu.ph

Engr. Ronnie S. Concepcion II
Electronics and Communications Engineering Department
De La Salle University
Manila, Philippines
ronnie_concepcionii@dlsu.edu.ph

Engr. Hilario A. Calinao Jr.
Electronics and Communications Engineering Department
De La Salle University
Manila, Philippines
hilario_calinao@dlsu.edu.ph

Engr. Jonnel Alejandrino
Electronics and Communications Engineering Department
De La Salle University
Manila, Philippines
jonnel_alejandrino@dlsu.edu.ph

Dr. Elmer P. Dadios
Manufacturing Engineering and Management Department
De La Salle University
Manila, Philippines
elmer.dadios@dlsu.edu.ph

Dr. Edwin Sybingco
Electronics and Communications Engineering Department
De La Salle University
Manila, Philippines
edwin.sybingco@dlsu.edu.ph

***Abstract*— Energy production of photovoltaic (PV) system is heavily influenced by solar irradiance. Accurate prediction of solar irradiance leads to optimal dispatching of available energy resources and anticipating end-user demand. However, it is difficult to do due to fluctuating nature of weather patterns. In the study, neural network models were defined to predict solar irradiance values based on weather patterns. Models included in the study are artificial neural network, convolutional neural network, bidirectional long-short term memory (LSTM) and stacked LSTM. Preprocessing methods such as data normalization and principal component analysis were applied before model training. Regression metrics such as mean squared error (MSE), maximum residual error (max error), mean absolute error (MAE), explained variance score (EVS), and regression score function (R^2 score), were used to evaluate the performance of model prediction. Plots such as prediction curves, learning curves, and histogram of error distribution were also considered as well for further analysis of model performance. All models showed that it is capable of learning unforeseen values, however, stacked LSTM has the best results with the max error, R^2, MAE, MSE, and EVS values of 651.536, 0.953, 41.738, 5124.686, and 0.946, respectively.**

***Keywords*—Bayesian ridge regression, convolutional neural network, energy management system, long-short term memory, solar irradiance prediction**

I. Introduction

Energy production of PV system is heavily influenced by solar irradiance and weather patterns The nature of solar irradiance and weather conditions results to fluctuating characteristics of solar power; hence, accurate prediction of solar energy is difficult. Factors such as cloud presence, aerosols, and dust affect the accuracy of predictions made [1].

Accurate solar irradiance forecasting is recommended for ensuring economic integration of PV system in the smart grid. Prediction of energy production and end-user demand is crucial for achieving optimal dispatching of available energy resources and anticipating end-user demand [2]. Forecasting methods allows the management of the uncertain nature of solar irradiance. It enables end users to be flexible in making decisions in operations such as load scheduling [3]. Users can also enjoy the benefits of profitable bilateral contracts because of its profit made in selling of excess energy produced to electric distribution company.

There has been an intensive research in developing solar irradiance models. It is reported to have four approaches; statistical approach, physical approach, and artificial intelligence (AI) approach. Statistical approaches depends on time series data of measured parameters to learn its present trends. Physical approach depends on satellite images and numerical weather predictions (NWP). However, these methods require extensive labor and financial resources. AI approach is an economic and reliable way for prediction. It uses techniques such as artificial neural network for forecasting by using exhaustive search of complex patterns to forecast output values [4][5].

This study used AI approach to forecast solar irradiance values for energy management system. Four neural network models were considered in defining the forecast models; artificial neural network, convolutional neural network, bidirectional long-short term memory, and stacked long-short term memory. Preprocessing of dataset was done first to lessen the computational time requirement in model training. Features used for solar irradiance are weather patterns due to its strong correlations. The performance of model prediction is evaluated using plots and various regression metrics.

II. Dataset Exploration

A. Data Gathering

The weather station deployed in Morong, Rizal consists of microcontrollers with sensors. It records the measurements of the considered weather parameters. Data gathering was conducted between September 2019 up to February 2020. Recorded measurements are stored in file as comma separated values (.csv) format. Table 1 shows the sensors used in gathering weather parameters and Table 2 shows the statistics of the parameters chosen as features during modeling.

TABLE I. SENSORS USED IN WEATHER STATION

Sensors used and Parameters measured		
Name of Sensor or Equipment	***Parameter Measured***	***Unit***
PDB C-139 Photodiode based Pyranometer	Solar Irradiance	W/m^2
DHT22	Humidity and Station Temperature	Humidity is expressed in percentage, temperature is in °C
DS18B20	Ambient Temperature	°C
BMP 180	Station Altitude, Sea Level Pressure, Absolute Pressure	Station Altitude: m Sea Level and Absolute Pressure: hPa
Anemometer	Windspeed	km/sec
BH1750	Illuminance	Lux

The output variable in the study is the solar irradiance. Solar irradiance is the amount of intensity from the sun, usually in the form of electromagnetic radiation, hitting power per unit each second. It is measured in terms of watt per square meter (W/m^2). Pyranometer is used for measuring solar irradiance on plane surface. The pyranometer used in the study is a photodiode calibrated based on its spectral property. The photodiode is enclosed with Teflon diffuser for proper reading operations.

Features are windspeed, ambient light, humidity, ambient and station temperature, station altitude, and absolute and sea level pressure. To measure ambient temperature, a digital thermometer DS18B20 was used. It is a waterproof sensor used for capturing temperature even in wet conditions. It can sense data up to 125 degree Celsius. A piezoresistive sensor called BMP180 was used in capturing data about station altitude, sea level pressure and absolute pressure. DHT22 was used for capturing humidity and the weather station temperature. Illuminance is measured using digital ambient light sensor found in mobile phones called BH1750. Lastly, anemometer was used in measuring windspeed. The equipment outputs a voltage signal ranging from 0 to 5 V and transforms it into analog wind speed measurement. In the study, it can measure up to 30 m/s.

TABLE II. STATISTICS OF FEATURE PARAMETERS

Feature	***Average***	***Standard Deviation***	***Minimum Value***	***Maximum Value***
Windspeed	15.9235	2.6095	0.5065	26.6640
Illuminance	1131.1892	710.1353	0.0245	4541.8745
Humidity	74.1667	19.1923	31.8	100
Ambient Temperature	31.8207	12.1710	16.8200	63.94
Station Temperature	32.6449	4.9210	23.98	45.8
Sea Level Pressure	1020.3179	2.3139	1012.4520	1027.7080
Station Altitude	69.77	19.1071	8.8	135
Absolute Pressure	1004.8927	2.2789	997.1420	1012.1680

B. Pair Plots

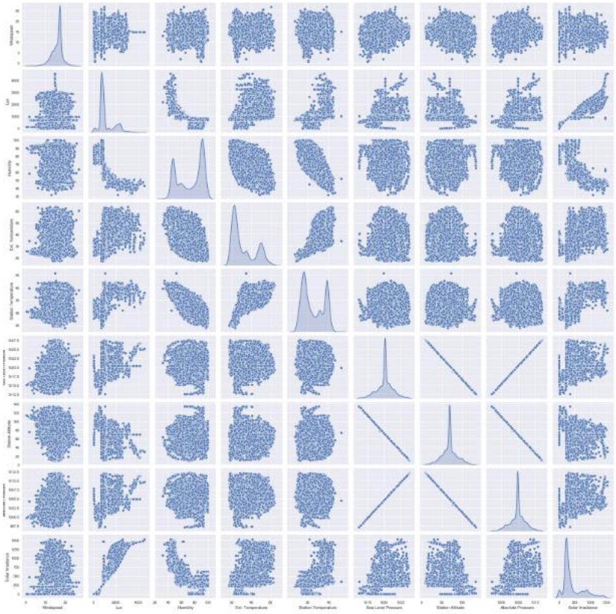


Fig. 1. Pair Plot with Kernel Density Estimators found in Diagonal

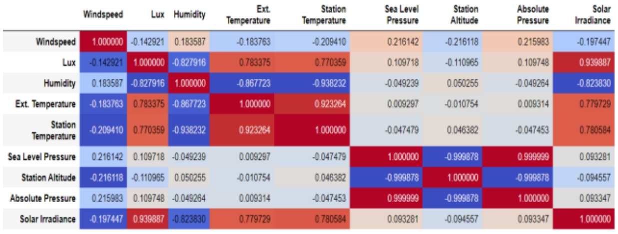
Figure 1 shows the visualization of the relationship between features and its spread using pair plots. The diagonal part of the visualization is the kernel density estimators of each features. Kernel density estimators is a non-parametric mathematical process of finding the estimates of probability density function of continuous random variable. It is an attempt to draw an inference about a certain feature based on finite samples. It also allows to see the spread of each parameters.

Some of the parameters don't have normal distribution like appearance such as humidity and temperature. Features such as wind speed, solar irradiance and illuminance are either skewed to the left or right. Pressure and station altitude are the only features that has almost a normal distribution appearance. Linear regression models are not enough to fit the data and requires more powerful processes such as neural networks. One of the assumptions for fitting regression models is the normality of all variables. Neural networks have no assumption on data, errors, or target. It has the capability to assume any function, even for complex patterns. Neural networks have activation function which is responsible for handling non-linearity of sequences.

Observing the scatter plot between two parameters, it shows an upward trend between solar irradiance and illuminance. This implies that as sunlight intensifies, so is the illuminance. Humidity shows downward trend to solar irradiance and illuminance. Humidity strongly correlates with presence of rain. During rainfall, humidity increases due to evaporation. Rainfall implies less sunlight intensity and lower ambient temperature. It is also noted that station altitude and pressure possess downward linear relationship.

Lastly, absolute pressure and sea level pressure shows a perfect upward linear relationship. This can imply the redundancy of using both features during model fitting.

C. Correlation Map



	Windspeed	Lux	Humidity	Ext. Temperature	Station Temperature	Sea Level Pressure	Station Altitude	Absolute Pressure	Solar Irradiance
Windspeed	1.000000	-0.142921	0.183587	-0.183763	-0.209410	0.216142	-0.216118	0.215983	-0.197447
Lux	-0.142921	1.000000	-0.827916	0.783375	0.770359	0.109718	-0.110965	0.109748	0.939887
Humidity	0.183587	-0.827916	1.000000	-0.867723	-0.938232	-0.049239	0.050255	-0.049264	-0.823830
Ext. Temperature	-0.183763	0.783375	-0.867723	1.000000	0.923264	0.009297	-0.010754	0.009314	0.779729
Station Temperature	-0.209410	0.770359	-0.938232	0.923264	1.000000	-0.047479	0.046382	-0.047453	0.780584
Sea Level Pressure	0.216142	0.109718	-0.049239	0.009297	-0.047479	1.000000	-0.999878	0.999999	0.093281
Station Altitude	-0.216118	-0.110965	0.050255	-0.010754	0.046382	-0.999878	1.000000	-0.999878	-0.094557
Absolute Pressure	0.215983	0.109748	-0.049264	0.009314	-0.047453	0.999999	-0.999878	1.000000	0.093347
Solar Irradiance	-0.197447	0.939887	-0.823830	0.779729	0.780584	0.093281	-0.094557	0.093347	1.000000

Fig. 2. Correlation Map of the Features

Figure 2 shows the correlation map between parameters. Since the scatter plot of solar irradiance and illuminance trends upward, its regression coefficient has a value of 0.94. Since humidity shows downward trend with solar irradiance and illuminance, its regression coefficient are negative values. Ambient temperature has regression coefficient value of -0.8279 and 0.8238 to illuminance and solar irradiance, respectively. Humidity has inverse relationship with ambient temperature and station temperature with regression coefficient values -0.8667 and -0.9382, respectively. Windspeed has weak direct relationship with humidity, sea level pressure and absolute pressure having regression coefficient values of 0.1836, 0.2161, and 0.2159, respectively.

III. Methodology

A. Flowchart

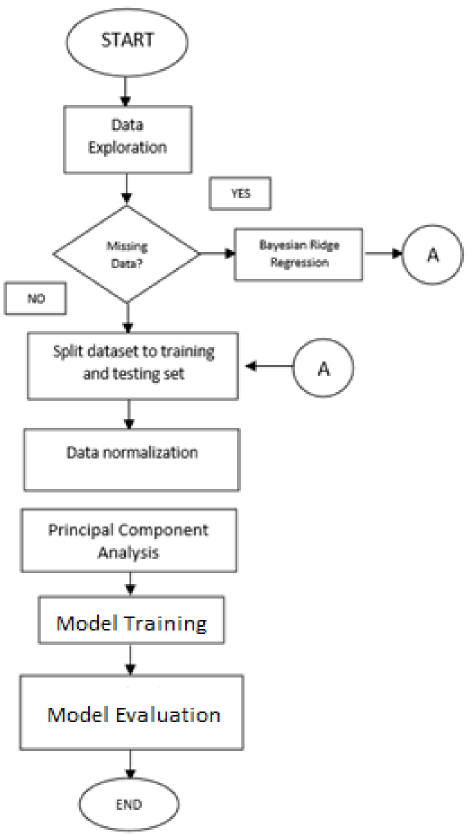


Fig. 3. Flowchart of the Methodology

Figure 3 shows the flowchart of the study's methodology. It is necessary to explore the dataset first by checking the relationship between parameters and the nature of its distribution. Check if there are missing values in the dataset. To handle missing data, apply data imputation by using Bayesian Ridge Regression. After splitting the dataset into training and testing set, normalize the dataset to obtain faster convergence during model training. Use principal component analysis to eliminate redundant features. Afterwards, define and train the models. Callback methods such as early stopping is applied to avoid model overfitting. The training will stop if learning curve does not improve. It is important to reshape the dimension of the input sequences from one-dimensional to three-dimensional arrays since it is required input dimension of the models. Lastly, evaluate the performance of model's prediction by observing the plot of training history's learning curves, histogram of error distributions, and plot of predicted values versus the true values found in validation set. To check the performance of the model quantitively, evaluate the made predictions and validation set using various regression metrics.

B. Data Preprocessing

1) Bayesian Ridge Regression

Bayesian Ridge Regression was implemented to estimate the value of missing rows based on the correlation found on rows. Instead of interpolation, it estimates model for regression problems using probability distribution [6][7]. The output value assumes that it is generated from Gaussian distribution. The model for Bayesian Linear Regression with the output drawn from normal distribution is expressed as

$$y \sim N(\beta^T X, \sigma^2 I) \quad (1)$$

The output is generated from Gaussian distribution in terms of mean and variance. The mean for linear regression is the product of the transpose of weight matrix and predictor matrix. Variance is expressed in terms of the product of square of standard deviation and identity matrix. Bayesian ridge regression works by finding the posterior distribution of model parameters since it is also assumed that it is generated from Gaussian distribution as well. The posterior probability of the model is a conditional probability, in terms of training and output sequences or

$$P(\beta|y, X) = \frac{P(y|\beta, X) * P(\beta|X)}{P(y|X)} \quad (2)$$

For simplicity sake, posterior probability is the ratio of the product of likelihood of data and prior probability of parameters to normalization. Bayesian ridge regression has its advantage: adaptability to the given dataset and use of regularization parameters during estimation. However, model inferences can cost a lot of computational time.

2) Data Normalization

Data normalization is also referred as data rescaling because datasets are rescaled by the minimum and range values of array without modifying the differences in each element. Elements now have common scale, ranging from 0 to 1. Normalizing data helps improves training the model, leading to faster convergence. Other preprocessing methods such as principal component analysis, is quite sensitive regarding to initial variances and requires prior normalization [8][9][10]. Data normalization is expressed mathematically as,

$$\frac{X - X_{min}}{X_{max} - X_{min}} \quad (3)$$

Where X is any point in column, X_{min} is the minimum value found in column, and X_{max} is the maximum value found in column,

3) *Principal Component Analysis*

Principal component analysis (PCA) is a statistical technique used for transforming higher dimensional dataset into lower dimension while retaining most of the information from the previous dataset. It is done by creating new and smaller uncorrelated variables called principal components from larger, correlated variables. Principal components maximize the variability of the data, leading to solving an eigenvector problem. PCA assumes that the sample size should be greater than 150 and there should be a correlation between features [11][12][13].

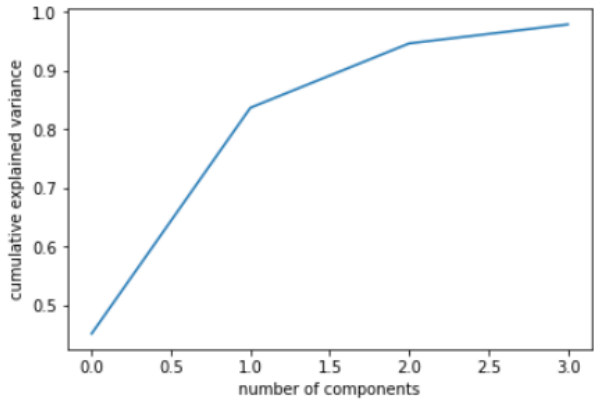


Fig. 4. Cumulative Explained Variance

Figure 4 shows the cumulative explained variance curve. This determines what number of principal components should be created during dimension reduction. From the plot, it is shown that if the component exceeds to 3, the variance will be close to 100%. Using the source code, PCA projected the original dataset into 4 principal components. To see the quantity of variance attributed to principal components, the explained variance ratio was calculated. This is done by getting the ratio of one principal component to the total variance.

TABLE III. EXPLAINED VARIANCE RATIO

Principal Component	*Explained Variance Ratio*
1	0.4518
2	0.3851
3	0.1096
4	0.0324

Table 3 shows the explained variance ratio of each principal component. 97% of the variance was retained after principal component analysis. Principal component 1 holds 45.18% of the variance, principal component 2 holds 38.51% of the variance, and principal components 3 and 4 holds the rest of the variance with the percentage of 10.96% and 3.24%, respectively.

C. *Neural Network Modeling*

1) *Artificial Neural Network (ANN)*

Table 4 shows the summary of the artificial neural network model used in the study. It consists of 2 hidden layers, with 64 nodes. Both hidden layers use rectified linear function as an activation function. Output layer has only one node. Rmsprop optimizer was used as a loss function which is responsible for searching the best weights during model training. Rmsprop is based on the moving average of square gradients and then dividing the gradient by the root average.

TABLE IV. MODEL SUMMARY OF ARTIFICIAL NEURAL NETWORK

Layer	*Output Shape*	*Number of Parameters*
Dense	(None, 64)	320
Dense	(None, 64)	4160
Dense	(None,1)	65

2) *Convolutional Neural Network (CNN)*

Table 5 shows the summary of the convolutional neural network model used in the study. The first layer of CNN is the Conv1D layer. This layer is used for sequences with fixed length segment such as time series data. It has 64 output filters used for convolution operation with kernel size of 2 units. It uses rectified linear unit as activation function. Next layer is the max pooling which is responsible for preventing overfit of learned features. It moves its windows steps 2 units, defined by its parameter “pool size”. The input sequence should be reshaped into 3 dimensions since it requires 3 parameters namely, batch size, steps, and features. It has a hidden layer with 50 nodes and output layer with one node. Adam algorithm was used as a loss function.

TABLE V. MODEL SUMMARY OF CONVOLUTIONAL NEURAL NETWORK

Layer	*Output Shape*	*Number of Parameters*
Conv1D	(None,3,64)	192
Max Pooling 1D	(None,1, 64)	0
Dense	(None,1,50)	3250
Dense	(None,1,1)	51

3) *Bidirectional Long-Short Term Memory (bidirectional LSTM)*

Table 6 shows the summary of the bidirectional LSTM used in the study. It is a variant of LSTM meant to improve the performance of traditional LSTM during sequence classification problems. It can preserve information both past and future timesteps since it trains 2 LSTMs. In the study, bidirectional LSTM uses 128 memory units and uses rectified linear unit as activation function.

TABLE VI. MODEL SUMMARY OF BIDIRECTIONAL LSTM

Layer	*Output Shape*	*Number of Parameters*
LSTM	(None, 4, 4)	33792
Dense	(None,1)	65

4) *Stacked Long Short Term Memory (Stacked LSTM)*

Table 7 shows the summary of the stacked LSTM used in the study. Stacked LSTM possess multiple hidden LSTM layers and contains multiple memory units. In the study, it has 2 LSTM layers with 64 neurons as memory units. It uses rectified linear unit as activation function.

TABLE VII. MODEL SUMMARY OF STACKED LSTM

Layer	*Output Shape*	*Number of Parameters*
LSTM	(None,4, 64)	16896
LSTM	(None, 64)	33024
Dense	(None,1)	65

D. *Metrics for Assessing Prediction Performance*

1) Explained Variance Score

Explained variance score (EVS) explains the dispersion of a given dataset, quantified in terms of variance. The explained variance is mathematically expressed as

$$explained\ variance(y, \hat{y}) = 1 - \frac{Var(y-\hat{y})}{Var(y)} \quad (4)$$

Where $\hat{y}$ is the predicted value, y is the true value and *Var()* is the variance. The best possible value for explained variance is 1.

2) Maximum Residual Error

Max error computes the maximum residual error, which captures the deviation between predicted values and true values. A perfect fitted model has a max error of 0 although it is unlikely to have that score. The metric provides the extent of error during model fitting. Maximum residual error is mathematically expressed as

$$Max\ Error\ (y, \hat{y}) = \max\left(|y_i - \hat{y}_i|\right) \quad (5)$$

Where $\hat{y}_i$ is the predicted value of the i-th sample and y_i is its corresponding true value.

3) Mean Absolute Error

Mean absolute error (MAE) refers to the absolute error. Given that $\hat{y}_i$ is the predicted value of the i-th sample and y_i is its corresponding true value, it is mathematically expressed as

$$MAE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} |y_i - \hat{y}_i| \quad (6)$$

4) Mean Squared Error

Mean squared error (MSE) refers to the quadratic error. Given that $\hat{y}_i$ is the predicted value of the i-th sample and y_i is its corresponding true value, it is mathematically expressed as

$$MSE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} (y_i - \hat{y}_i)^2 \quad (7)$$

5) Regression Score Function

Regression score function R^2 computes the coefficient of determination. Coefficient of determination refers to the ratio of variance between two variables. It gives information about the goodness of fit and performance of model for predicting samples not included in the training set. Best possible score for this metric is 1.0 and can output a negative value. Given that $\hat{y}_i$ is the predicted value of the i-th sample and y_i is its corresponding true value over n samples, it is mathematically defined as

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2} \quad (8)$$

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \quad (9)$$

IV. Neural Network Modeling

This section presents the result of fitting and training of the models used in the study. For guidance, the result of the model is arranged clockwise, namely, artificial neural network, convolutional neural network, bidirectional long-short term memory, and stacked long short-term memory.

A. *Learning Curves*

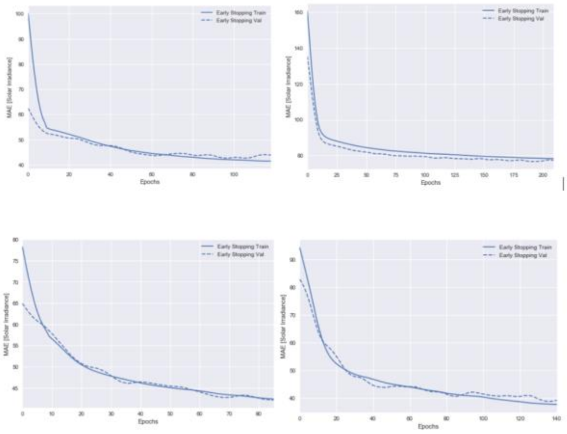


Fig. 5. Plot of Training History with Mean Absolute Error As Metric

Figure 5 shows the learning curve of the models used with mean absolute error on the Y-axis. In all models, error decreases as the training progresses. The validation and training error are close to one another, which indicates that model is capable of learning values not present in the training set. For ANN, the validation and test error are close to one another which is an indication of a good fit and the model can learn more. It takes about 100 epochs to be stable in learning the input sequence. Its validation error is slightly higher than the training error and takes an estimated average error of 45 units. For CNN, however, the validation error is slightly lower than the training error. The model might overlook random samples during model prediction. It took 200 epochs for the CNN to be stable in learning. It is noted that its mean average error is estimated at around 80 units. Both Bi-LSTM and stacked LSTM has its validation error slightly higher than the training error. Also, both models have an estimated mean average error of 35 units. Bi-LSTM is the fastest model to get stable in learning, recorded at around 80 epochs.

B. *Model Prediction*

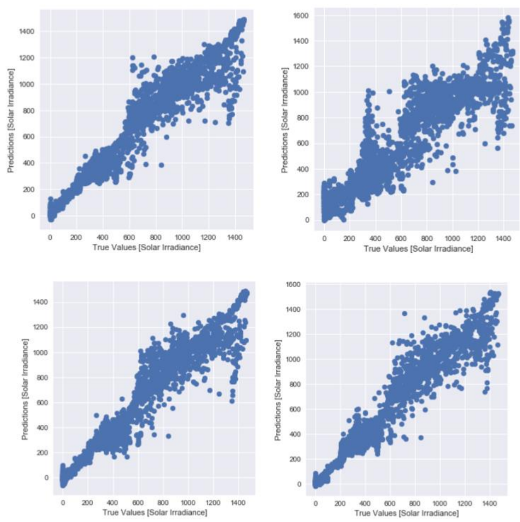


Fig. 6. Plot of Model's Predicted Values vs True Values

Figure 6 shows how the model generalizes using the testing set by plotting the predicted values versus true values. A diagonal like appearance indicates good performance of model in prediction. Out of 4 models, only PCA-CNN has many outliers and doesn't have a good diagonal appearance. This is because in observance of the plot of training history, the model might overfit and overlook future values.

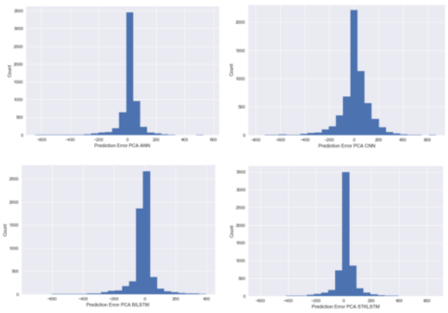


Fig. 7. Error Distribution of Model's Predicted Values

Figure 7 shows the histogram of the error count of each model. All models produce a Gaussian like distribution appearance. Compared to other models, CNN has more spikes in error counts.

V. Metrics Evaluation

TABLE VIII. Result of Metric Evaluation In Model Performance

Model	*Max Error*	*R^2Score*	*MAE*	*MSE*	*EVS*
PCA-ANN	644.131	0.945	44.302	6064.393	0.946
PCA-CNN	786.688	0.867	77.849	14623.66	0.867
PCA-BILSTM	737.558	0.945	41.901	6063.944	0.946
PCAST KLSTM	651.536	0.953	41.738	5124.686	0.946

Table 8 shows the result of metric evaluation for the performance of model in predicting solar irradiance. It is noted that CNN performs poorly compared to other models because of having highest values in error. This is because CNN is applicable to classification problems [14]. Stacked LSTM has the best metrics evaluation result. It has the lowest error produced during model training and the highest score for regression score function. Therefore, it is the model that can predict the closest values found in the testing set.

VI. Conclusion

The researchers conclude that stacked LSTM is the best model for predicting solar irradiance as supported by plots presented and metric evaluations. Stacked LSTM has the least amount of errors produced on a certain period of training epoch. It can also predict the closest value of the testing set by noting its regression score. This is because stacked LSTM is the appropriate model for time-series data [15]. It is recommended to implement ensemble learning with stacked LSTM for improvement in model prediction.

Acknowledgment

The researchers express their gratitude to the Department of Science Technology - Engineering Research and Development Technology (DOST-ERDT) and the De La Salle University Intelligent Systems Laboratory (DLSU-ISL) under the project of "Smart Farm" for their support.

References

[1] A. Kumar, M. Rizwan, and U. Nangia, "Artificial neural network based model for short term solar radiation forecasting considering aerosol index," *2018 2nd IEEE Int. Conf. Power Electron. Intell. Control Energy Syst. ICPEICES 2018*, pp. 212–217, 2018, doi: 10.1109/ICPEICES.2018.8897290.

[2] Y. Iwafune, Y. Yagita, T. Ikegami, and K. Ogimoto, "Short-term Forecasting of Residential Building Load for Distributed Energy Management," *2014 IEEE Int. Energy Conf.*, pp. 1197–1204, 2014, doi: 10.1109/ENERGYCON.2014.6850575.

[3] K. Aurangzeb, "Short Term Power Load Forecasting using Machine Learning Models for energy management in a smart community," *2019 Int. Conf. Comput. Inf. Sci.*, pp. 1–6, 2019.

[4] F. Serttas, F. O. Hocaoglu, and E. Akarslan, "Short term solar power generation forecasting: A novel approach," *PVCon 2018 - Int. Conf. Photovolt. Sci. Technol.*, pp. 2018–2021, 2018, doi: 10.1109/PVCon.2018.8523919.

[5] C. M. T. Huang, Y. C. Huang, and K. Y. Huang, "A hybrid method for one-day ahead hourly forecasting of PV power output," *Proc. 2014 9th IEEE Conf. Ind. Electron. Appl. ICIEA 2014*, vol. 5, no. 3, pp. 526–531, 2014, doi: 10.1109/ICIEA.2014.6931220.

[6] Y. Yang and Y. Yang, "Hybrid prediction method for wind speed combining ensemble empirical mode decomposition and bayesian ridge regression," *IEEE Access*, vol. 8, pp. 71206–71218, 2020, doi: 10.1109/ACCESS.2020.2984020.

[7] P. Krzysztof, H. Wiktor, N. Henryk, and K. Wojciech, "Energy performance of photovoltaic panels due to phenomena of urban air pollution disposal," *2016 17th Int. Carpathian Control Conf.*, pp. 407–412, 2016, doi: 10.1109/CarpathianCC.2016.7501132.

[8] M. A. Salama, A. E. Hassanien, and A. A. Fahmy, "Reducing the influence of normalization on data classification," *2010 Int. Conf. Comput. Inf. Syst. Ind. Manag. Appl. CISIM 2010*, pp. 609–613, 2010, doi: 10.1109/CISIM.2010.5643523.

[9] M. S. Azmi, N. A. Arbain, A. K. Muda, Z. A. Abas, and Z. Muslim, "Data normalization for triangle features by adapting triangle nature for better classification," *2015 IEEE Jordan Conf. Appl. Electr. Eng. Comput. Technol. AEECT 2015*, 2015, doi: 10.1109/AEECT.2015.7360572.

[10] V. Sathya Durga and T. Jeyaprakash, "An Effective Data Normalization Strategy for Academic Datasets using Log Values," *Proc. 4th Int. Conf. Commun. Electron. Syst. ICCES 2019*, no. Icces, pp. 610–612, 2019, doi: 10.1109/ICCES45898.2019.9002089.

[11] R. Gordillo-orquera, L. M. Lopez-ramos, S. Muñoz-romero, P. Iglesias-casarrubios, D. A. Id, and A. G. M. Id, "Analyzing and Forecasting Electrical Load Consumption in Healthcare Buildings," pp. 1–18, doi: 10.3390/en11030493.

[12] L. Chen, Y. Liu, and C. Zhu, "ROBUST TENSOR PRINCIPAL COMPONENT ANALYSIS IN ALL MODES Longxi Chen Ce Zhu School of Information and Communication Engineering / Center for Robotics University of Electronic Science and Technology of China (UESTC), Chengdu , 611731 , China," *2018 IEEE Int. Conf. Multimed. Expo*, pp. 0–5, 2018.

[13] P. Mair, "Principal Component Analysis and Extensions," no. Icecs, pp. 179–210, 2018, doi: 10.1007/978-3-319-93177-7_6.

[14] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," *Proc. 2017 Int. Conf. Eng. Technol. ICET 2017*, vol. 2018-Janua, no. August, pp. 1–6, 2018, doi: 10.1109/ICEngTechnol.2017.8308186.

[15] S. Siami-Namini, N. Tavakoli, and A. Siami Namin, "A Comparison of ARIMA and LSTM in Forecasting Time Series," *Proc. - 17th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2018*, no. April 2019, pp. 1394–1401, 2019, doi: 10.1109/ICMLA.2018.00227.