

# A Simplified LSTM Neural Networks for One Day-Ahead Solar Power Forecasting

CHUN-HUNG LIU<sup>1</sup>, JYH-CHERNG GU<sup>1</sup>, (Member, IEEE),  
AND MING-TA YANG<sup>2</sup>, (Member, IEEE)

<sup>1</sup>Department of Electrical Engineering, National Taiwan University Science and Technology, Taipei 106335, Taiwan

<sup>2</sup>Department of Electrical Engineering, National Penghu University of Science and Technology, Magong 880011, Taiwan

Corresponding author: Ming-Ta Yang (mtyang@gms.npu.edu.tw)

**ABSTRACT** In recent years, exploration and exploitation of renewable energies are turning a new chapter toward the development of energy policy, technology and business ecosystem in all the countries. Distributed energy resources (DERs) are being largely interconnected to electrical power grids. This dispersed and intermittent generational mixes bring technical and economic challenges to the power systems in terms of operations, stability, reliability, interoperability and the policy making. In addition, DERs cause the significant impacts to the operation of traditional centralized generation power plants and the dispatch control centers. Under such circumstances, the accuracy of DERs power forecasting is one of the critical problems for TSO and DSO such as unit commitment, smooth fluctuations, peak load shifting, demand response, etc. In this paper, a simplified LSTM algorithm built over the architecture of Machine Learning methodology to forecast one day-ahead solar power generation is introduced. Through the machine learning processes of data processing, model fitting, cross validation, metrics evaluation and hyperparameters tuning, the result shows that the proposed simplified LSTM model outperform the MLP model. Moreover, the forecast of LSTM model can successfully capture intra-hour ramping on different weather scenarios. The average RMSE is 0.512 which is quite promising to inspire that the proposed methodology and architecture can best fit the short-term solar power forecasting applications.


**INDEX TERMS** Artificial neural networks, DER, LSTM, machine learning, solar power forecasting.

## I. INTRODUCTION

The growing penetration of DERs implementation brings a significant impact on grid stability and operation. The accurate forecasting of DERs generation can help TSO and DSO to optimize the unit commitment and regulate the power quality as well as the activity of demand response. Today, the study of PV forecasting becomes one of the mainstreams in terms of the prediction research territory. Many algorithms are used in Solar Power Forecasting such as Numerical Weather Prediction (NWP), Cloud Imagery and Satellite-Based Models, Statistical Time Series Models, and Artificial Neural Networks (ANN), which are one of the most popular algorithms for such study. In this work, the author will focus on the discussion of the different ANNs models.

ANNs have become popular among researchers since 1980. They have been applied in various prediction

applications as the most effective methods and more specifically to the forecasting of PV power generation where deliver a higher level of accuracy. ANNs are widely introduced in forecasting the PV power generation in most researches due to its non-linearity in meteorological data. It is more suitable compared with the statistical methods when a non-linear and complicated bonding exists between the data without any prior assumption [1]. In most studies, a single-layer ANN is used to solve the problem. However, a significant number of complex problems are caused by the unsolvable data pattern using a single layer ANN. Complex input and output relationships exist among different variables. To overcome these problems, ANNs have been modified into several types that follow different architectures and input-output mapping procedures. Among these research works, the most commonly used ANNs are Multilayer Perceptron (MLP), Feed Forward Neural Network (FFNN), Radial Basis Neural Network (RBNN), Recurrent Neural Network (RNN), Back Propagation Neural Network (BPNN), General Regression

The associate editor coordinating the review of this manuscript and approving it for publication was Juan A. Lara .

Neural Network (GRNN), and Adaptive Neuro-Fuzzy Interface Systems (ANFIS).

MLP is a supervised feed-forward ANN [2], [3] that consists of one or more layers between the input and output layers of the network. These layers, called hidden layers can be adjusted to suit the complexity of a problem. A previous study demonstrated that more than two hidden layers are rarely required when performing complex mapping [4]. FFNN is a relatively less complex ANN architecture. In FFNN, the information only moves in a forward direction from the input layer to the output layer. To process the information doesn't require a feedback loop or cycle. These neural networks have been adopted in several forecasting and pattern-recognition applications [5]–[8]. RBNN is considered a two-layer ANN. The learning process can be separated into two different stages based on synaptic weights [9]. *Mandal et al.* reported that RBNN achieves good performances and involves less computation time for learning [10]. When applied to PV power forecasting, RBNN is preferred for its universal approximation property and structural simplicity [11]. RBNNs are used for the daily global solar radiation prediction, using meteorological data such as air temperature, humidity, and sunshine duration [12]. RNN performs well in learning different and computational structures and complex relationships. Therefore, it is considered a good method for time-series data forecasting. *Yona et al.* reported that compared with FFNN, errors in forecasting results are significantly minimized with RNN [13]. BPNN is the most powerful supervised learning algorithm [14], [15]. However, this algorithm tends to have a slow convergence rate, a training prone to oscillations, and easily falls into the local minimum [16]. To address these limitations, the researchers introduced a modified BP network [17]. Among other ANN-related methods, BPNN has been widely used because of its excellent non-linear mapping function, which is suitable for solving complex regression problems [18]–[22]. The first study of LSTM in day-ahead solar energy forecasting, the empirical results show that LSTM outperform a large number of alternative methods [23]. A correction approach using discrete grey model (GDM) with LSTM for non-ideal weather profile is discussed [24]. Two LSTM neural networks are employed working on temperature and power outputs forecasting and show excellent performances compared with ARIMAX and MLP [25]. An LSTM model with a clearness-index approach and classify the type of weather by k-means is introduced to improve the prediction accuracy on cloudy days [26]. Moreover, recently, a partial daily pattern prediction (PDPP) framework is proposed to improve the TCM (time correction modification) on LSTM-RNN model [27]. However, the main drawbacks of ANNs are overfitting and the large number of data required during the training process [28], [29], which increases the complexity of the implementation and the cost in the real practice

In this study, a simplified approach of the time series forecasting using LSTM model based on machine learning architecture for one day-ahead solar power forecasting is

proposed, using the limited train data to obtain the good forecasting result without the sacrifice of accuracy. The train data matrix with the moving window technique, also introduced in [30], is well presented and the methodologies of machine learning processes are detailed discussed. The architecture of this paper is arranged as follows: Section I introduces the mainstream research of ANNs. Section II discusses the methodologies of Machine learning, LSTM block, data preparation, and preprocessing. Section III introduces the experiment architecture and test results. Section IV for discussion and Section V for the conclusion and future work.

## II. METHODOLOGY

### A. INTRODUCTION TO LSTM NEURAL NETWORK

Like other recurrent neural networks, LSTM is a neural network of units that are interconnected as shown in Fig. 1. Each unit contains information about the previous state.

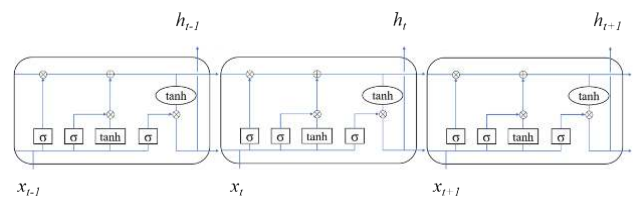


FIGURE 1. Overview of LSTM interactive layers.

LSTM networks are comprised of various gates that contain information about the previous state. This information is either written, stored or read from a cell that is acting like a memory. The cell decides on whether to store the incoming information, when it reads, writes and erase via the gates opening and closure. They act based on the signals received and block or pass on information based on its strength and import by filtering with their own sets of weights. These weights are similar to those that modulate input and hidden states by adjusting through the network's training process. The model proposed in this paper is a Long Short-Term Memory (LSTM) network that consists of an input layer, a hidden layer, and an output layer. Fig. 2 shows the LSTM model featuring of input gate,  $i_t$ , output gate,  $o_t$ , forget gate,  $f_t$  and cell state,  $\tilde{C}_t$ .

The forget gate  $f_t$  shown in Fig. 3 must decide what information to be kept and what information to be discarded from the cell state  $\tilde{C}_t$ . The decision is made by the logistic function which outputs a value between 0 to 1. A value means to keep the totality of the information and 1 means to forget its totality. Function is described in (1).

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

where

$\sigma$ : Activation function

$W_f$ : Weight of forget gate

$b_f$ : Bias of forget gate

$x_t$ : Input at time  $t$

$h_{t-1}$ : Hidden layer output at time  $t-1$

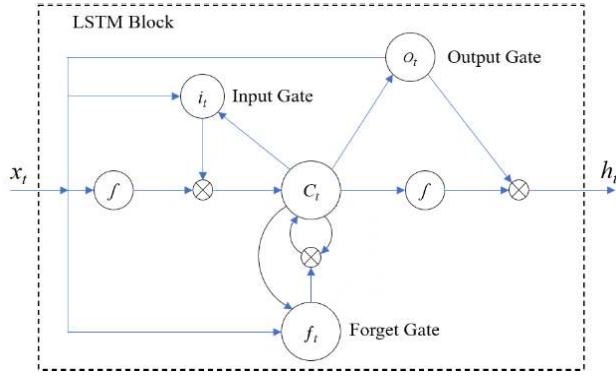


FIGURE 2. LSTM unit.

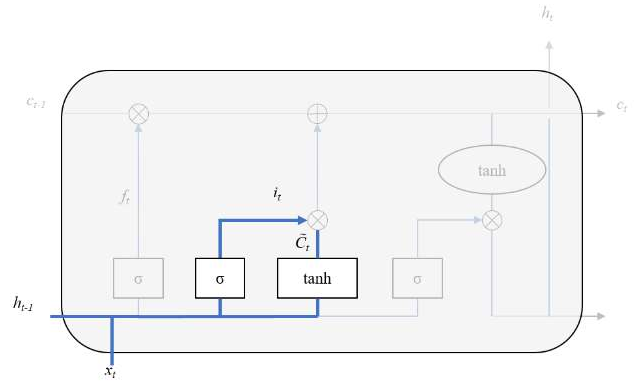


FIGURE 4. Flow of the Input gate  $i_t$ .

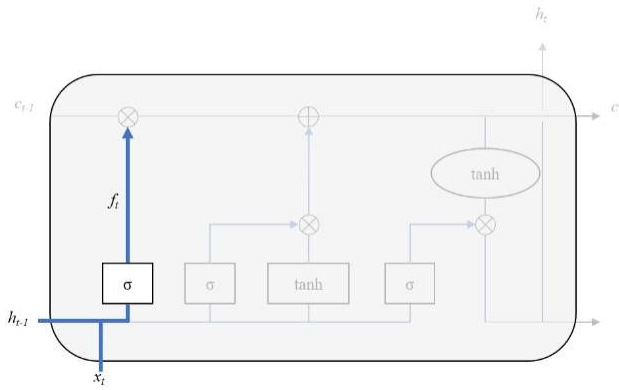


FIGURE 3. Flow of the Forget gate  $f_t$ .

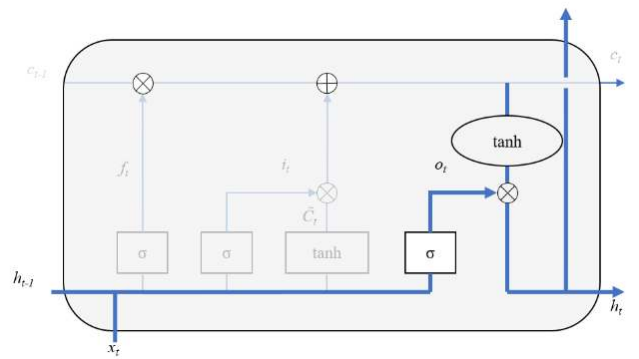


FIGURE 5. Flow of the output gate  $o_t$ .

The input gate  $i_t$  shown in Fig. 4 decides the input values to be updated by the LSTM blocks. The calculation for  $i_t$  is determined by (2) and the calculation of  $\tilde{C}_t$  is determined by (3).

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (3)$$

where:

$\sigma$  : Activation function

$W_f$ : Weight of forget gate

$b_i$ : Bias of input gate

$x_t$ : Input at time  $t$

$h_{t-1}$ : Hidden layer output at time  $t-1$

$W_c$ : Weight of cell

$b_c$ : Bias of cell

The final state would be the output state shown in Fig. 5. A tanh is included in the output gate  $o_t$  to decide which part of the cell state is chosen to output. For the hidden gate  $h_t$ , the output gate  $o_t$  is multiplied with another logistic function to scale values between 0 and 1. The calculation for  $o_t$  is determined by (4) and the calculation of  $h_t$  is determined by (5).

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (4)$$

$$h_t = o_t + \tanh(C_t) \quad (5)$$

where:

$\sigma$  : Activation function

$W_o$ : Weight of output gate

$B_o$ : Bias of output gate

### 1) DATA MATRIX INPUTS

The input data matrix to feed LSTM model is three-dimensional array as shown in Fig. 6.

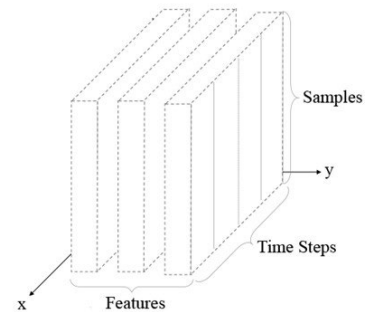


FIGURE 6. LSTM input three-dimensional array.

The first dimension represents the number of samples trained in the network. The second dimension represents the time-steps, and the third dimension represents the number of features in one input sequence.

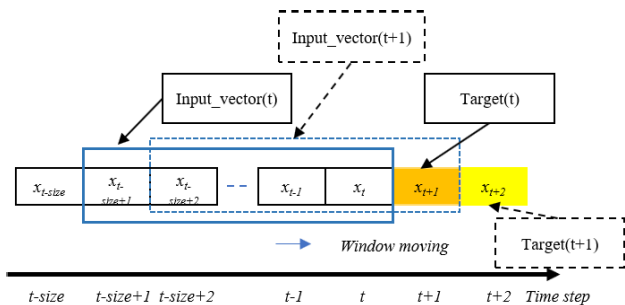
LSTM input layer is defined by the input shape argument of the first layer. The input shape argument takes a tuple of two values that define the number of time steps and features. The number of samples is assumed to be one in our case but can be more in case of multiple inputs. The three inputs components are given with a 3-dimensional array input.

Considering our test where we have one sequence of multiple time steps which and one feature, we have a following time series of one single data signal. This one-dimensional array will be reshaped into a three-dimensional array with one sample, the number of timesteps to predict one day, and one feature at each time step. The reshape process must evenly reorganize the data in the array. Based on past observations (from  $x$  to  $x_{t-n}$ ), a new value of Power is predicted (Next Step) at each step. The input data matrix is illustrated in Table 1.

**TABLE 1. Input data matrix.**

Step	Next Step				
#1	...	$x_{t-3}$	$x_{t-2}$	$x_{t-1}$	$x_t$
#2		$x_{t-4}$	$x_{t-3}$	$x_{t-2}$	$x_{t-1}$
#3		$x_{t-5}$	$x_{t-4}$	$x_{t-3}$	$x_{t-2}$
#4		$x_{t-6}$	$x_{t-5}$	$x_{t-4}$	$x_{t-3}$
#5		$x_{t-7}$	$x_{t-6}$	$x_{t-5}$	$x_{t-4}$
#6		$x_{t-8}$	$x_{t-7}$	$x_{t-6}$	$x_{t-5}$
#7		$x_{t-9}$	$x_{t-8}$	$x_{t-7}$	$x_{t-6}$
...					...

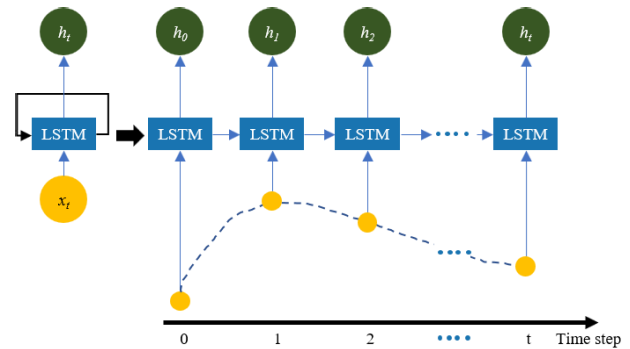
The behavior of the LSTM network is shown in Fig. 7. The LSTM network will read the past observations from  $t-1$  to  $t-n$  (the number of past observations defined by the Input Vector( $t$ )) to feed the input vector at the first timestep. As a result, a value target ( $t$ ) is found at timestep ( $t+1$ ). Then at the second timestep, the Input Vector increases of one timestep ( $t+1$ ) and the same process is repeated to find the value at target ( $t+1$ ) which is at the timestep ( $t+2$ ).



**FIGURE 7. Input matrix mechanism.**

In the Fig. 8 one can see the two outputs  $h_t$  and  $o_t$  of the LSTM Neural Network at each step. An input value  $x_t$  feeds the LSTM Neural Network that produces a hidden input  $h_t$  and an output value  $o_t$  to feed the LSTM at each new step.

The architecture of Machine Learning can be shown in Fig. 9. Raw data are collected from both weather data acquisition and photovoltaic energy production equipment



**FIGURE 8. LSTM input and outputs.**

such as PV inverter, energy meter and data logger and pre-processed. Pre-processing requires different steps that will be described later in chapter II.C. After data are pre-processed, they are sampled and split into two datasets for training and validation. These datasets will need to be preprocessed in order to select and scale their features before to reduce their dimensions to fit LSTM input requirements.

LSTM is trained with the training dataset and during the process, hyperparameters and optimization will be applied to the learning algorithm training. After training feature scaling and dimensionality reduction are inverted to revert changes to the initial dataset and the model is saved to generate predictions that compared with actual data for evaluation.

The first step of data preprocessing is the feature selection. In one dataset that features several signals, the feature selection is the process to select the best signal to feed the LSTM Neural Network. The method to select the best candidate signal is to check the correlation between signals.

The second step is the feature scaling. It involves two different data preprocessing which are normalization and standardization.

## 2) DATA NORMALIZATION

Normalization is the process of rescaling data from the original range so that all values are within the range of 0 and 1. Normalization requires to know or to accurately estimate the minimum and maximum observable values, which can be estimated from available recorded data. In case of solar activity, minimum and maximum values can be found easily since the solar pulse repeats daily. Data normalization is based on (6).

$$x_{norm}^{(i)} = \frac{x^{(i)}}{x_{max} - x_{min}} \tag{6}$$

where:

- $x^{(i)}$ : Particular sample
- $x_{norm}^{(i)}$ : New value of  $x^{(i)}$
- $x_{max}$ : Largest value in a feature column
- $x_{min}$ : Smallest value in a feature column

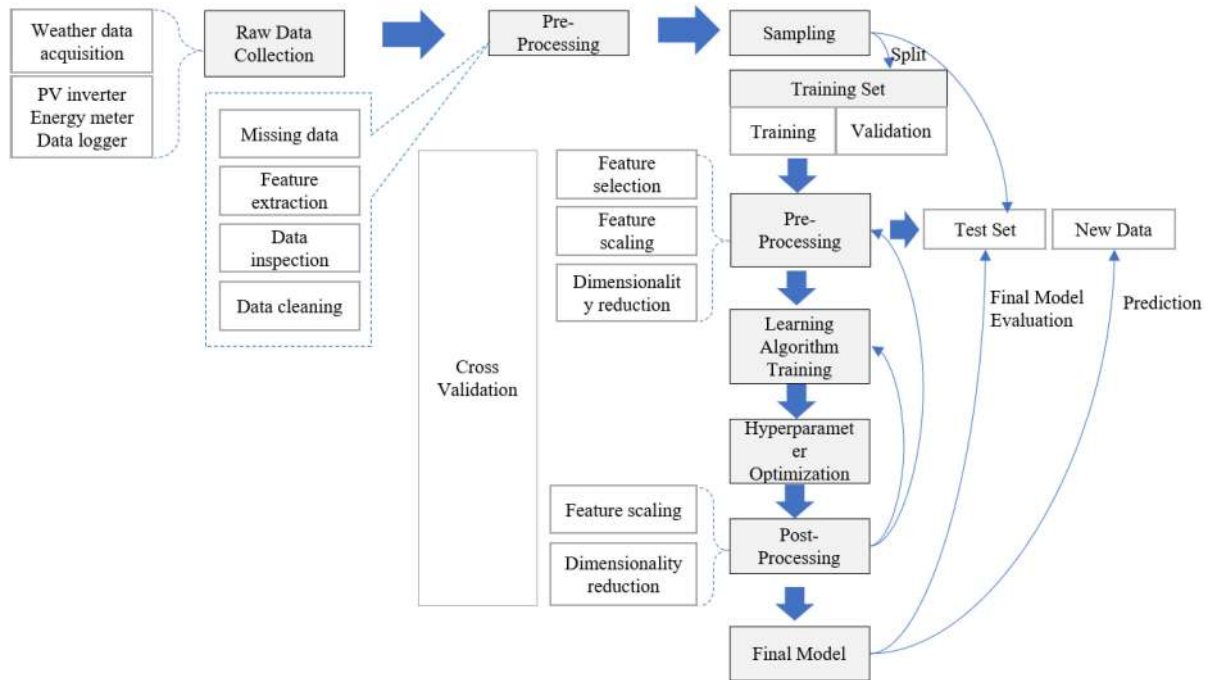


FIGURE 9. Architecture of machine learning.

As a result, each signal is normalized by a value between 0 and 1, which correspond with the minimum and maximum values they reached among the entire dataset.

3) DATA STANDARDIZATION

The principle of standardizing a dataset involves rescaling the distribution of values so that the mean of observed values is 0 and the standard deviation is 1. It can be obtained by subtracting the mean value or centering the data with (7).

$$x_{std}^{(i)} = \frac{x^{(i)} - u_x}{\sigma_x} \tag{7}$$

where:

- $x^{(i)}$ : Particular sample
- $x_{std}^{(i)}$ : New value of  $x^{(i)}$
- $u_x$ : Sample mean of particular feature column
- $\sigma_x$ : Corresponding standard deviation

Standardization can be useful and can be required in some machine learning algorithms when data have input values with differing scales. However, standardization assumes that observations fit a Gaussian distribution (bell curve) with a well-behaved mean and a standard deviation. Moreover, standardization requires to know or accurately estimate the mean and standard deviation of observable values. In case of solar power data, this estimation is possible from the training dataset.

4) NDIMENSIONALITY REDUCTION

Since LSTM only accept a three-dimensional array, training dataset needs to be reshaped accordingly to the input

data matrix described previously. Time series index is dropped, and time series is reshaped to host past observation data.

5) TRAIN, VALIDATION, TEST DATASETS

The training dataset is a dataset of examples used for learning the neural network dataset. A validation dataset is a dataset of examples used to tune the hyperparameters. A test dataset is a dataset that is independent of the training dataset, but that follows the same probability distribution as the training and validation datasets. For the purpose of the test, training dataset will represent 80% of the entire available data, while the validation dataset will occupy the last 20%. The test validation is one day of recorded data and not included in the training dataset or the test dataset. One can see an example of dataset division in Fig. 10 for KHH. The latest day of the original dataset is set as the test dataset while the rest of the dataset is split in 2 datasets: in blue the training dataset and in orange the validation dataset.

6) HYPERPARAMETERS

Hyperparameters allow to tailor the behavior of the LSTM training algorithm and for specific datasets. They are different from parameters, which are the internal coefficients or weights for a model found by the learning algorithm. Since it is challenging to tune LSTM model with the best hyperparameters, values have been set first to conduct the first series of tests. The number of neurons is set to 100 neurons and the activation function is set to “Relu.”

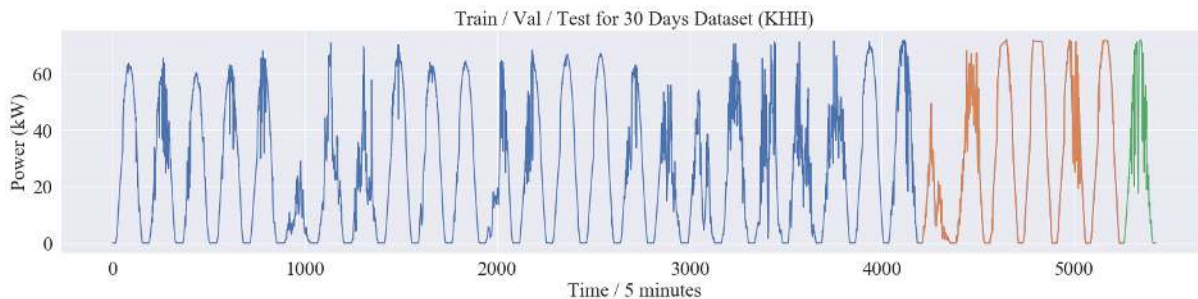


FIGURE 10. Dataset dividing.

**B. DATA ACQUISITION**

Raw data are collected from PV systems via photovoltaic equipment such as inverters, power meters, and weather station sensors and recorded. Data are centralized to an edge computing gateway where they were modeled and forwarded to a server for data analysis. The integration of data at the server is combined with data from weather forecasting third-party server and retried by API. The data collection diagram is shown in Fig. 11.

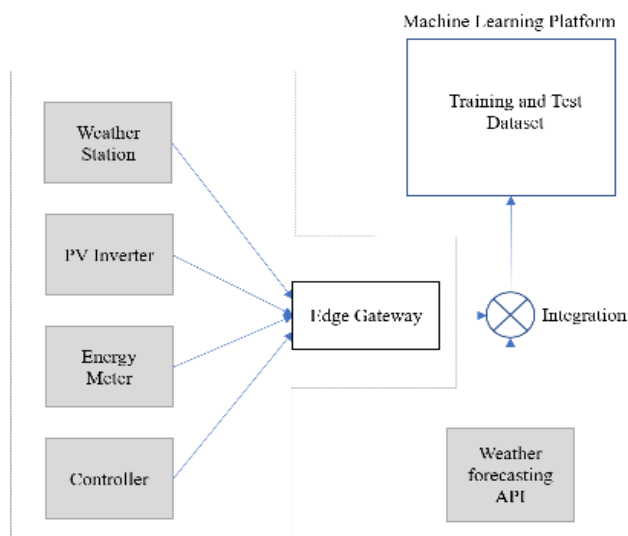


FIGURE 11. Data collection.

**C. DATA PREPROCESSING**

**1) MISSING DATA**

LSTM models can possibly handle missing values, but the quality in predicting data would still be affected if the datasets that provide data to train the LSTM network shows poor quality.

PV data are recorded periodically over the time by data loggers and series can be resampled to save storage or transport. Resampling data can prevent missing values when missing values don't exceed the new resampling time resolution. Also resampling data can solve the problem of variable time step

in a dataset, for example when data logger has been reconfigured or reset unintentionally.

However, due to other technical reasons such as system failures, large chunk of data can be corrupted or missing in the time series and cannot be recovered by data resampling. In case of PV data where solar activity is a cycle of 24 hour, it is possible to improve the quality of the dataset by removing entire days without affecting the integrity of the dataset.

**2) FEATURE EXTRACTION**

In order to extract the best features from the original dataset, the heating map will be used to reveal correlation between signals. The heating map can confirm the high correlation between the signal to forecast with other significant signals from the system. Fig. 12 is an illustration of a heating map generated from a dataset.

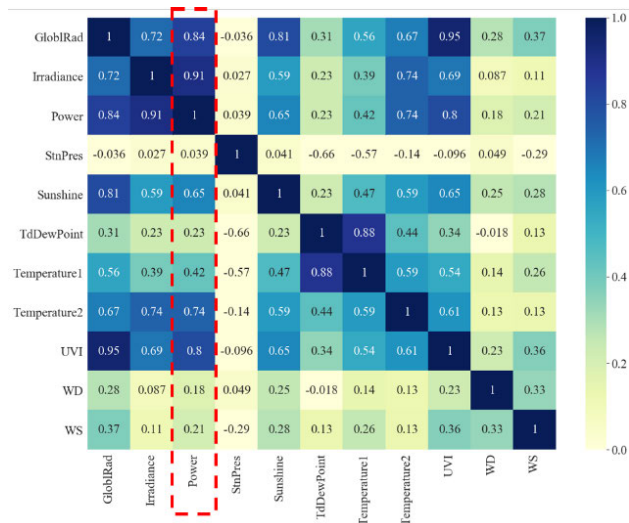


FIGURE 12. Heating map.

**3) DATA CLEANING**

PV data observation is usually a daily basis pulse of photovoltaic power, starting from zero watts just before sunrise and returning to zero watts just after sunset. A peak is observed somewhere around the middle of day. The pulse

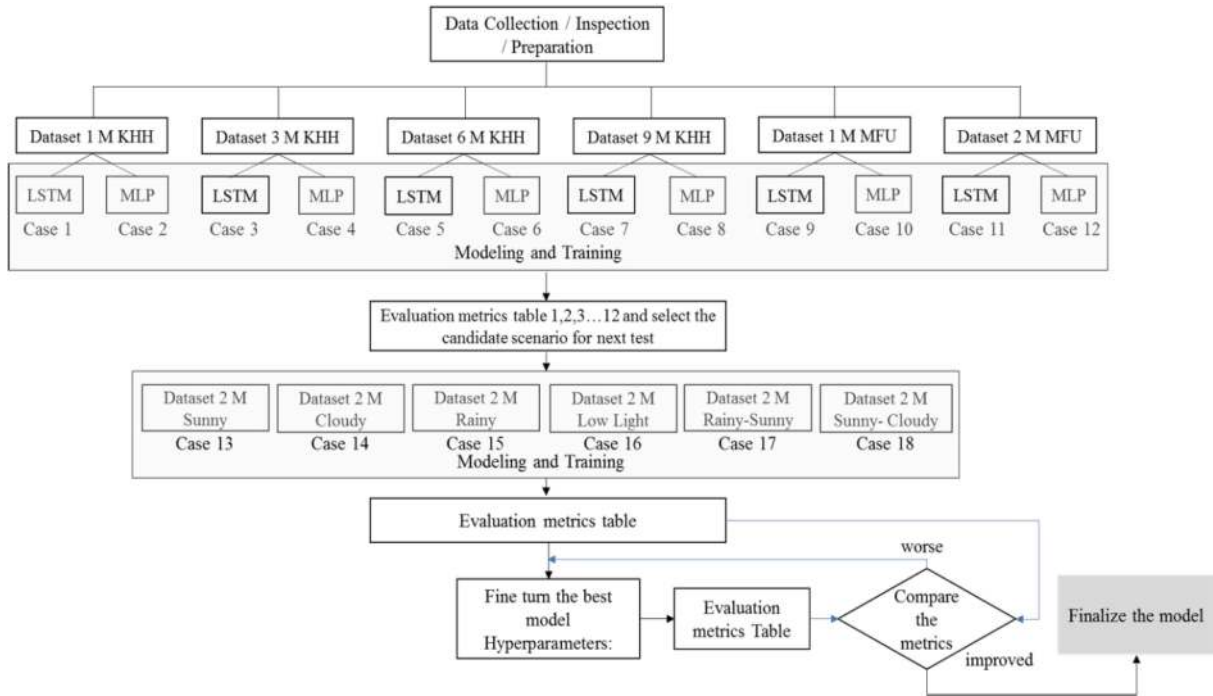


FIGURE 13. Architecture of experiment platform.

is usually corrupted by different factors, such as weather perturbation (clouds, rain, snow) and data is measured at regular intervals (such as every minute or every five minutes). Recorded data should be an accurate reflection of what the system records from the environment factors and recording conditions. However, real datasets often deviate from these expectations. Several disturbance factors affect the quality of the datasets: missing data (due to hardware or software resilience and performance, changing or irregular measurement intervals (expected or unexpected), or changes in local clock time (time shifts).

The reasons can be diverse: Data acquisition system may go offline for a period of time or fail to acquire data while the system is producing power. Also, it is common for some data acquisition systems to simply stop collecting data when the sun goes down.

Likewise, several causes can cause irregular measurement intervals. For example, data loggers can be commissioned to record data at 5 minutes interval and later changed to record at 1-minute interval. Also, timestamps can be skipped or delayed (skipped scans).

Recorded time can appear to shift suddenly relative to the underlying physical processes for some reasons. A common reason is the mishandling of daylight savings time shifts that occur in some countries. It is also common to observe devices not synchronized or having its internal clock changed for some reasons (power cut, etc.).

Abnormal data coming from a missing sensor reading or incorrect readings from logging devices can also result in

negative data and radiation data that are bigger than the theoretical data limits.

Fundamentally, if a data set is too corrupted by the types of errors described above, it will not be possible to extract useful information from that data set. But for moderately corrupted data sets, we want to clean up these types of errors so that we can then analyze the data, such as to estimate overall system degradation or look for loss factors. It is particularly interesting in methods that automate this process.

If a dataset is corrupted by several reasons described earlier, it will not be possible to extract useful information from it. However, for dataset corrupted moderately, it is possible to clean up data so data can be later extracted for analysis.

A first step would be to estimate the overall degradation of data by these factors. Methods have been developed to clean data in an automated process.

### III. EXPERIMENT AND TEST RESULT

This section will present experiment and test results of test that comply with the methodology presented in the second chapter.

#### A. ARCHITECTURE OF EXPERIMENT PLATFORM

The architecture of the experiment platform is shown in Fig. 13. The first test phase will compare LSTM with MLP neural networks with different sources and different lengths of datasets. After Evaluating metrics, best model and best dataset source and length will be selected and tuned in the second phase of the test. The number of neurons and

the activation function will be determined to find the best hyperparameters values and hence finalize the model.

1) MFU AND KHH PV SITES

Raw data inputs selected were the time series of two geographically separated PV systems from two different countries (Chiang Rai, Thailand, and Kaohsiung, Taiwan), which have the same tropical climate. Fig. 14. Illustrates the geographic location of the two PV systems.



FIGURE 14. Location of PV systems.

The meteorological data from Chiang Rai and Kaohsiung were collected respectively from the Mae Fah Luang University and Kaohsiung City Government.

2) DATA COLLECTION

The two PV systems will be designated as MFU (Chiang Rai) and KHH (Kaohsiung) in order to differentiate them. Table 2 shows the site information of these PV systems.

TABLE 2. PV systems characteristics.

PV Sites	MFU	KHH
Country	Thailand	Taiwan
City	Chiang Rai	Kaohsiung
Coordinates	19°54'34"N 99°49'39"E	22°38'N 120°16'E
Altitude	390 M	9 M
Average Temperature	24.2°C	24.4°C
Average RH	75%	76%
Daily Sunshine Hours	6.13	7

Raw data collected from the two PV systems have been collected from photovoltaic equipment such as inverters, power meters, and weather station sensors and recorded every 5 minutes and during different time frame. Fig. 15 illustrates a photovoltaic site system architecture with its equipment. Data are collected at MFU are recorded daily from 00:00 to 23:55 while data collected at KHH are recorded from 05:00 to 19:30. Also, both time series have missing and/or outliers' observations. By calculating minimum and maximum for each signal in the time series, we can notice consistent range values, except some outliers' values in the inverter temperature range.

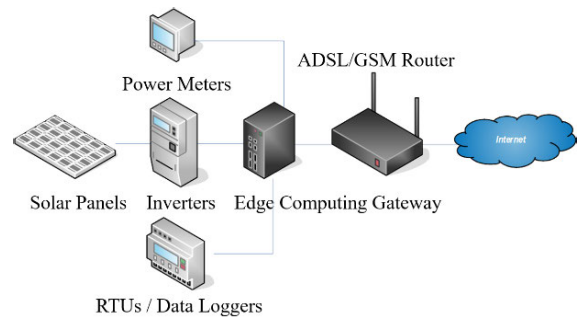


FIGURE 15. PV site system architecture.

The dataset collected from KHH features 20 data signals while 9 were collected at MFU. Table 3 and Table 4 show the data collected at KHH and MFU, respectively. Times series contains all signals recorded from photovoltaic equipment.

TABLE 3. KHH signal characteristics.

Data	Range
Generation (kW)	151989.0 - 262240.5
Global Radiance (W/m <sup>2</sup> )	0.0 - 3.6
Irradiance (W/m <sup>2</sup> )	-1 - 1283
Observed Time (H)	5h-19h
Power (kW)	0.0 - 101.757
Precipitation / Hour (mm)	0.0 - 1.0
Precipitation (mm)	0.0 - 43.5
PV Power (kW)	0.0 - 105.1
Relative Humidity (%)	26 - 100
Sea Pressure (Pa)	980.2 - 1025.7
Standard Pressure (Pa)	979.9 - 1025.3
Sunshine	0.0 - 1.0
Dew Point	4.6 - 28.2
Temperature 1 (°C)	14.0 - 34.4
Temperature 2 (°C)	-1 - 63
UV Index	0 - 12
Wind Direction (°)	0 - 360
Wind Direction Gust	10 - 350
Wind Speed (m/s)	0.0 - 15.9
Wind Speed Gust	1.3 - 31.0

3) DATA INSPECTION

When observing data over one month, we can distinctly see signal variations over each day. Fig. 16 and Fig. 17 show the variations of recorded signals for a period of one month for KHH and MFU respectively.

Variations are cyclic with one day period. When observing signal variations over one day, we can notice the correlation with the sun activity in Fig. 18.

4) HEATMAP

Correlation between signals can be overserved with a heatmap when applied to data in Fig. 19 and Fig. 20 for KHH and MFU, respectively.



TABLE 4. MFU signal characteristics.

Data	Range
Ambient Temperature (°C)	13.5875 - 52.1529
Humidity (%)	14.7895 - 93.5157
Inverter Temperature (°C)	0 - 165.8
Irradiance (W/m <sup>2</sup> )	-0.244922 - 1370.88
Module Temperature (°C)	24.0599 - 83.6777
Power (kW)	0 - 18.7847
PV Power (kW)	0 - 19.3937
Wind Direction (°)	-0.0439453 - 26.4091
Wind Speed (m/s)	5.75095 - 7.90951

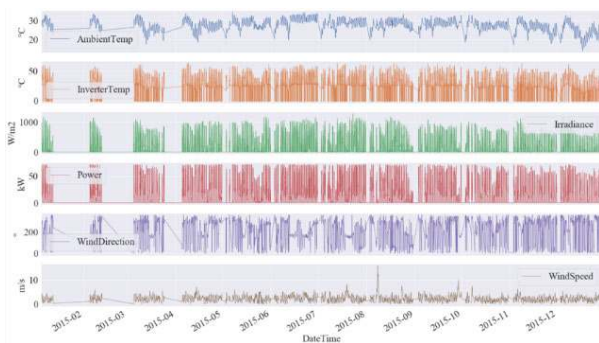


FIGURE 16. KHH 30 Days dataset.

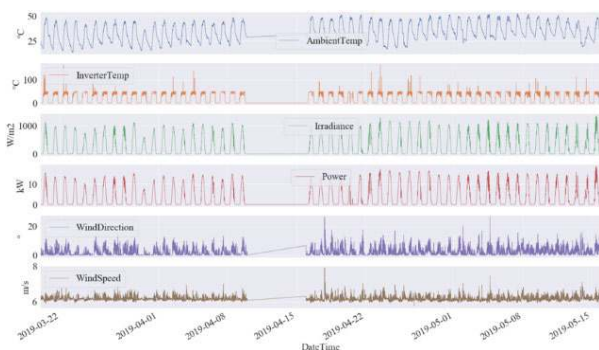


FIGURE 17. MFU 30 Days dataset.

High correlation can be observed between signals, in the KHH dataset for example, Power is high correlated with Irradiance, Global Irradiance and UV Index as Photovoltaic power is directly linked by these factors. However, other signals have low correlation with power, such as Standard Pressure or Wind.

A second data inspection has been carried to show correlation between signals and assess the quality of output active power data that is our first concern.

Fig. 21 shows correlation between power and other signals in KHH Dataset. Unsurprisingly, it can be observed that Power is highly correlated with UV Index, Irradiance and Global Irradiance it can be also noticed that power is likely to be high as the temperature increases. However, wind speed and wind gust have low correlation with Power.

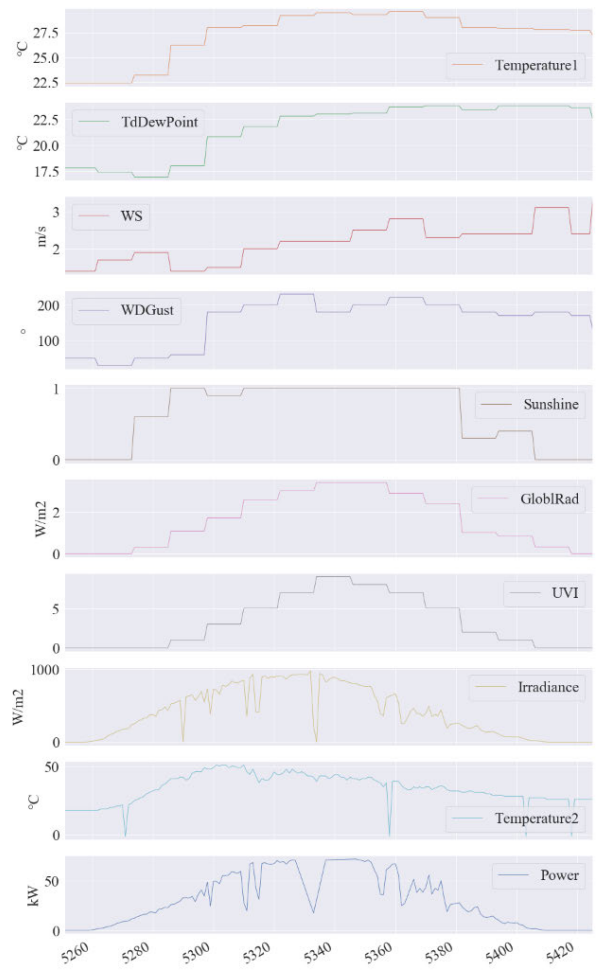


FIGURE 18. Signal variations over one day (KHH).

Fig. 22 shows the correlation between Power and other signals in MFU dataset. It can be observed that the temperature variables are moderately positively correlated with active power, while the humidity variable is highly negatively correlated with active power. The correlation results show that the active power is likely to be high as the increased temperature and the decreased humidity. Wind speed have low correlation values while irradiance and photovoltaic power have very high correlation values with active power.

Box plot of power daily variations per hour in Fig. 23 for KHH and Fig. 24 for MFU show maximum values at noon and best performances between 9:00 And 14:00 as sun rises higher above the horizon.

5) POWER MATRIX

We can visualize the dataset by extracting its data to a power matrix that shows the measured power for each recorded day. The power matrix can reveal information such as night shifts, missing data or inconsistencies in timestamps.

In Fig. 25, a function applied to the dataset reveal night shifts and missing data in MFU dataset.

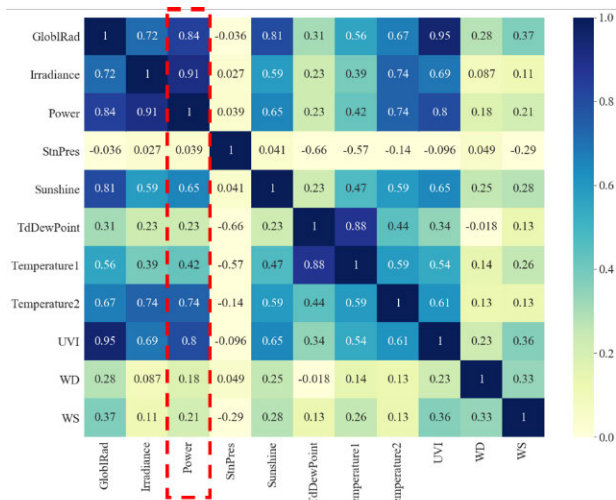


FIGURE 19. Heatmap (KHH).

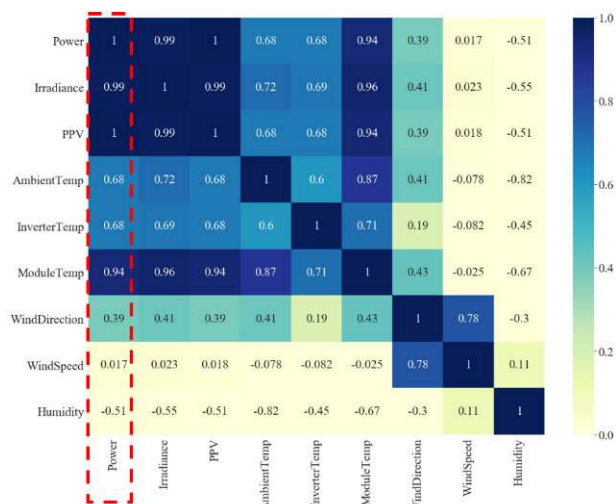


FIGURE 20. Heatmap (MFU).

Missing data can be filled, and night shifts can be ignored to reduce training time. Fig. 26 shows missing data filled in MFU dataset.

6) DATA PREPARATION

Clean datasets are first normalized and subdivided in 3 distinct datasets without overlapped data between them. For example, MFU 30 days dataset is split in 24 days for training data, 5 days for validation test and 1 day for test dataset. The three datasets are visualized in Fig. 27.

KHH time series is the longest with 278 recorded day while MFU only has 51 recorded days. For the purpose of our tests, we will use different lengths of dataset for KHH and MFU. Datasets lengths are defined in Table 5.

B. SCENARIOS OF INITIAL TESTS

The purpose of our first test is to evaluate the performance of different neural networks and to demonstrate the

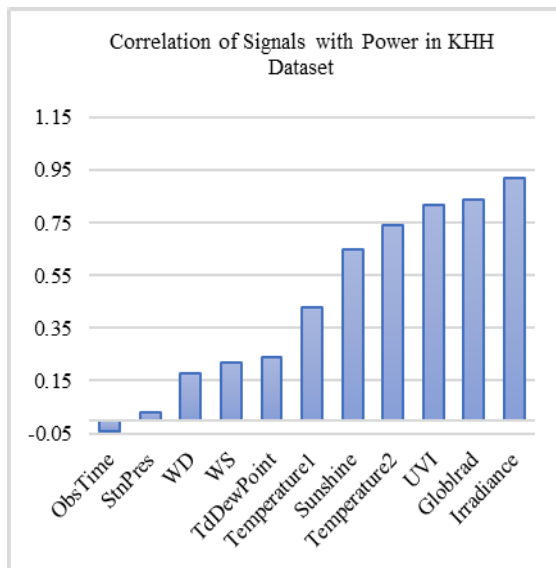


FIGURE 21. Correlation between Power and other signals in KHH dataset.

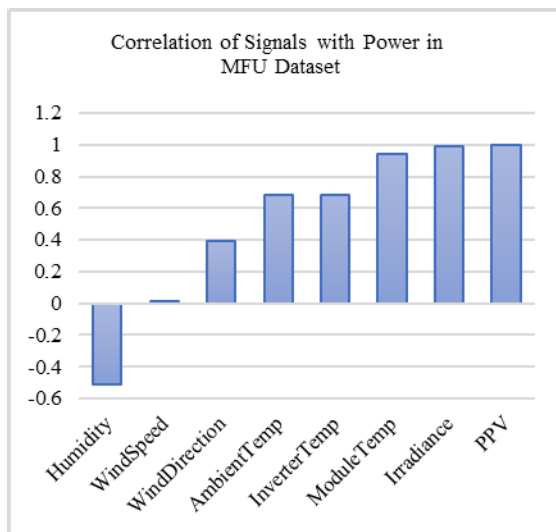


FIGURE 22. Correlation between Power and other signals in MFU dataset.

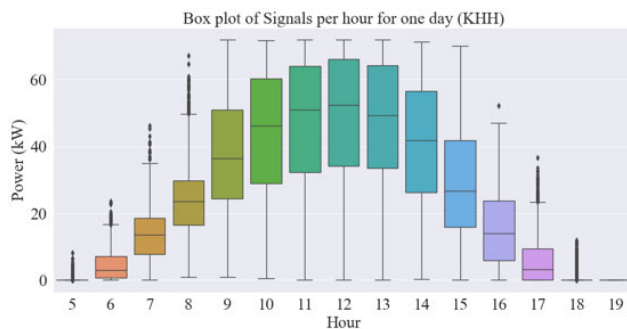


FIGURE 23. Box plot of Power signal per Hour for one Day in KHH dataset.

benefits of the Long Short-Term Memory recurrent neural network (LSTM) over Multilayer Perceptron (MLP).

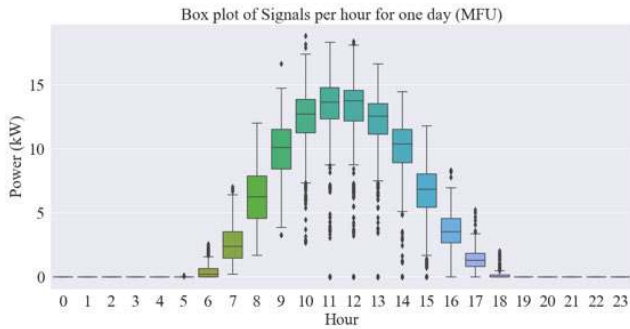


FIGURE 24. Box plot of power signal per Hour for one Day in KHH Dataset.

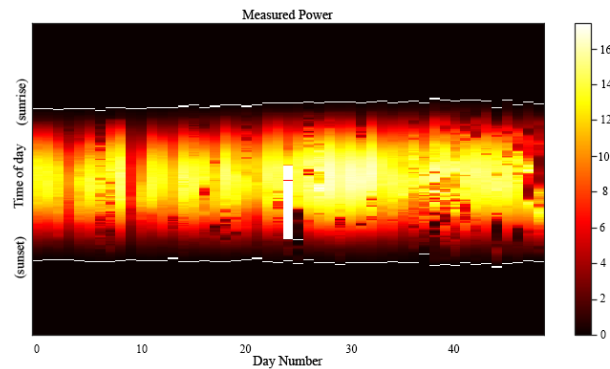


FIGURE 25. Night shifts and missing spotted data in MFU Dataset.

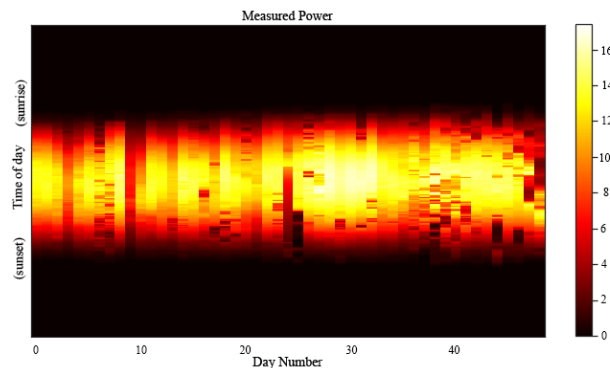


FIGURE 26. Dataset fixed in MFU Dataset.

Different parameters have been set for the two different models as they feature their proper characteristics. Table 6 summarizes the parameters used for the two models.

The multilayer perceptron is created with a first layer of 100 neurons to receive the input dimensions corresponding to number of points we defined for the past observations. An output layer is added to return the prediction. The non-linear function ‘Relu’ is set for activation. The model is compiled with ‘Adam’ optimizer and root mean squared error (RMSE) for loss calculation.

The long-short-term memory model is first created by calling a sequential constructor, then the LSTM layer is added

TABLE 5. Lengths of training validation and test datasets used during tests.

Dataset	MFU	KHH
Length	51 days	278 days
Training	30,50 days	1,3,6,9 Months
Validation	20% of Training Datasets	
Test	1 Day	

TABLE 6. Hyperparameters applied for each model MLP and LSTM.

Hyperparameter	MLP	LSTM
Model	Sequential	Sequential
Activation	Relu	Relu
Input	Input Array	Input Array
First Layer	100 neurons	100 neurons
Hidden Layer	-	- (Dropout)
Output	Output Array	Output Array
Loss Calculation	RMSE	RMSE
Optimizer	Adam	Adam

with a number of neurons equal to 100. A dropout layer is then added to prevent overfitting (regularization) and finally an output layer is created to receive outputs. The model is then compiled with RMSE as loss calculation method and ‘adam’ is chosen as optimizer.

The evaluation of these 2 models intends to compare and understand the level of accuracy of forecasts according to the indicators that have been presented in the “Hyperparameters: Inputs, Layers, Hidden Layers” section. Thus, one can compare different architectures and adjust the parameters in order to obtain the best possible prediction.

The two models MLP and LSTM will be evaluated by comparing the two different datasets KHH and MFU with different lengths (1 month, 3 months, 6 months, 9 months). Table 7 lists the 12 cases conducted during the tests.

### 1) RESULTS

The two learning machines have been trained, evaluated and then test consecutively with these multiple dataset lengths to predict one day forecast.

Fig. 28 to Fig. 39 show one day forecast compared with actual data predicted by the two models MLP and LSTM when trained with different dataset lengths.

### 2) EVALUATION METRICS

Loss, Validation Loss and RMSE are the most important evaluators for the assessment of the quality of the network. Therefore, though these evaluators are not comparable between them, they might exhibit the same behavior when comparing different simulations and datasets. Table 8 shows the results for the 12 first cases.

LSTM shows better accuracy compared with MLP when the model is trained with KHH dataset as illustrated in Fig 40.



FIGURE 27. Training Dataset (Blue), Validation (Orange) and Test Dataset (Green) in MFU Dataset.

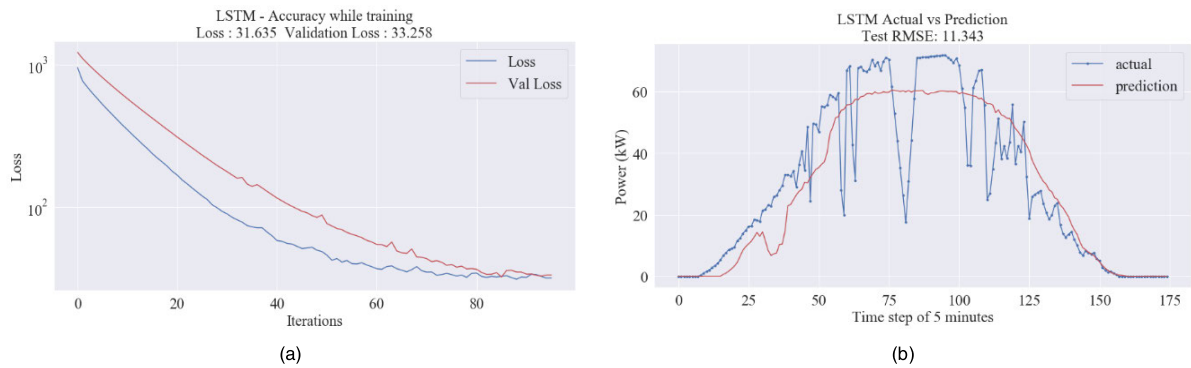


FIGURE 28. Test Case #1 LSTM with 1 Month Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

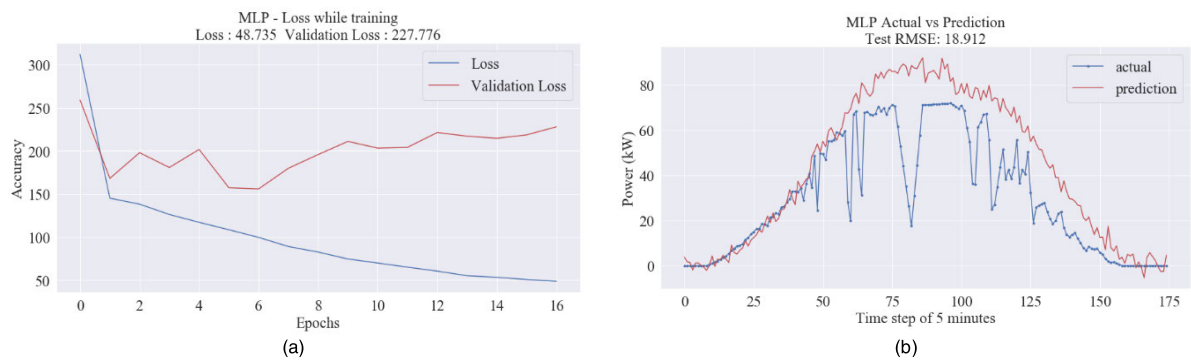


FIGURE 29. Test Case #2 MLP with 1 Month Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

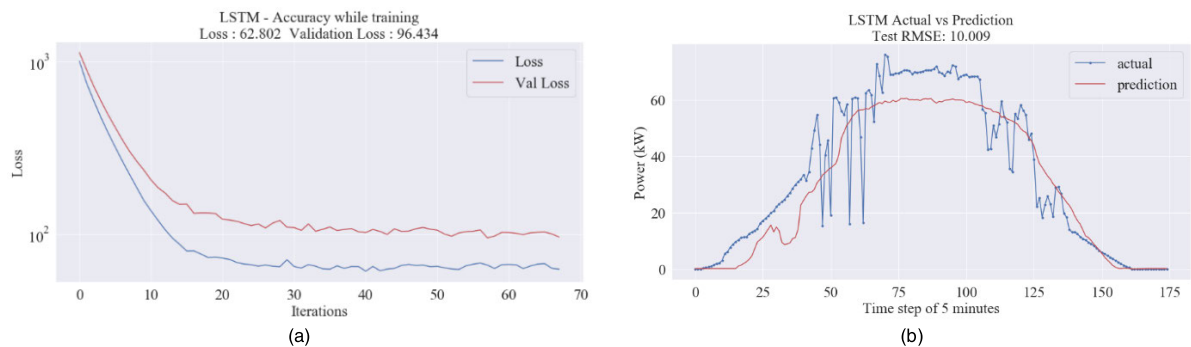


FIGURE 30. Test Case #3 LSTM with 3 Months Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

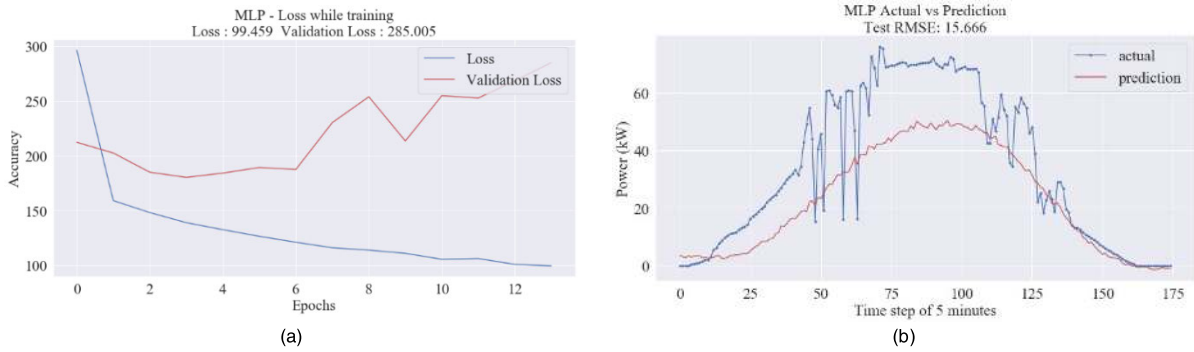


FIGURE 31. Test Case #4 MLP with 3 Months Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

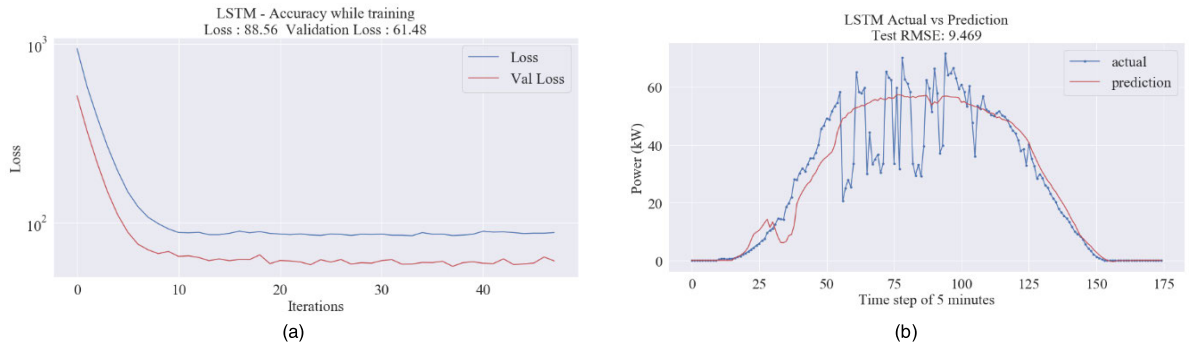


FIGURE 32. Test Case #5 LSTM with 6 Months Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

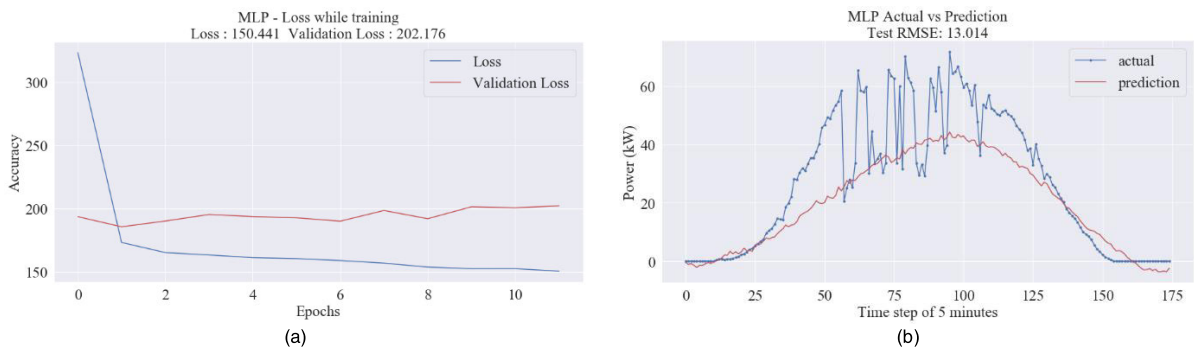


FIGURE 33. Test Case #6 MLP with 6 Months Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

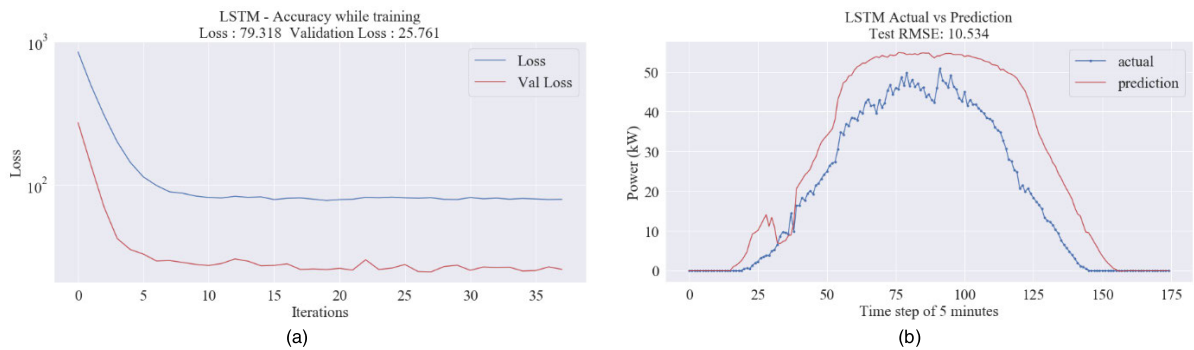


FIGURE 34. Test Case #7 LSTM with 9 Months Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

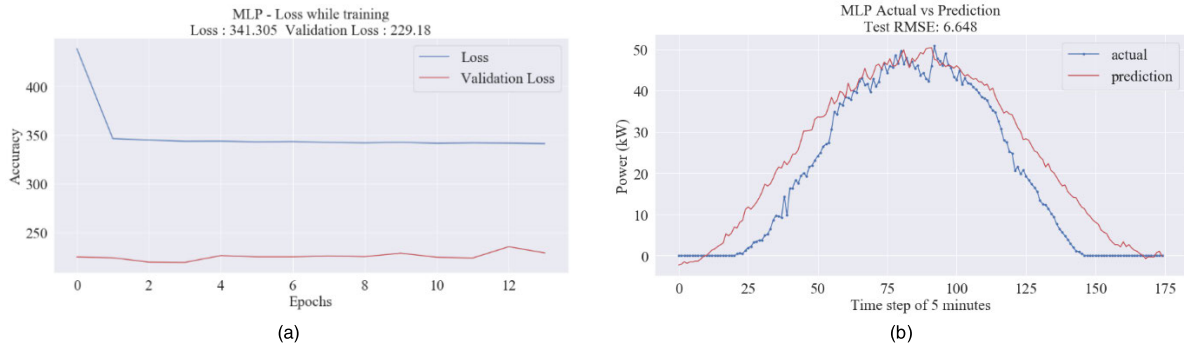


FIGURE 35. Test Case #8 MLP with 9 Months Dataset (KHH) (a) Accuracy (b) Actual vs Prediction.

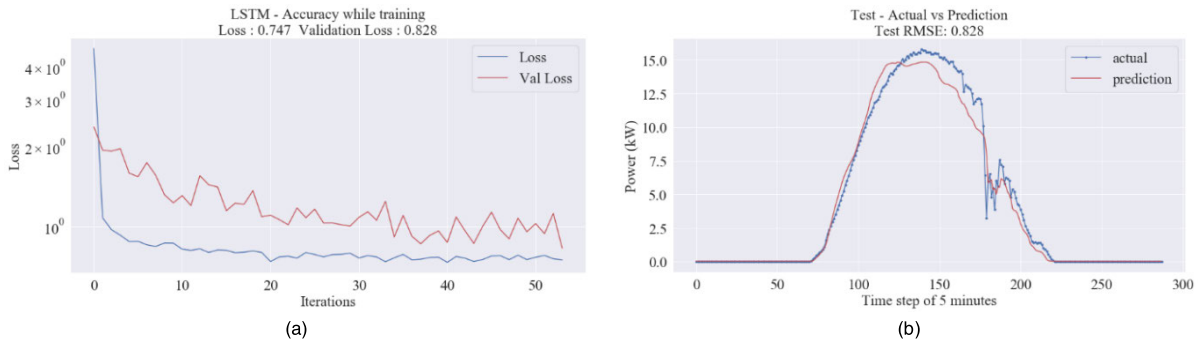


FIGURE 36. Test Case #9 LSTM with 30 Days Dataset (MFU) (a) Accuracy (b) Actual vs Prediction.

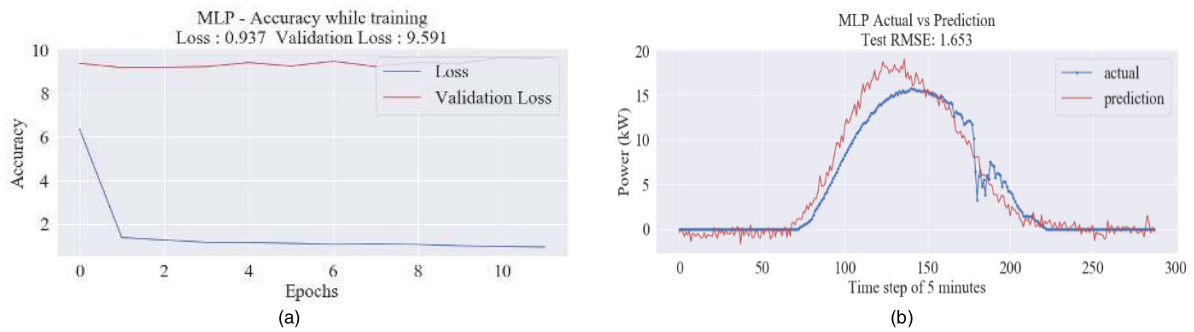


FIGURE 37. Test Case #10 MLP with 30 Days Dataset (MFU) (a) Accuracy (b) Actual vs Prediction.

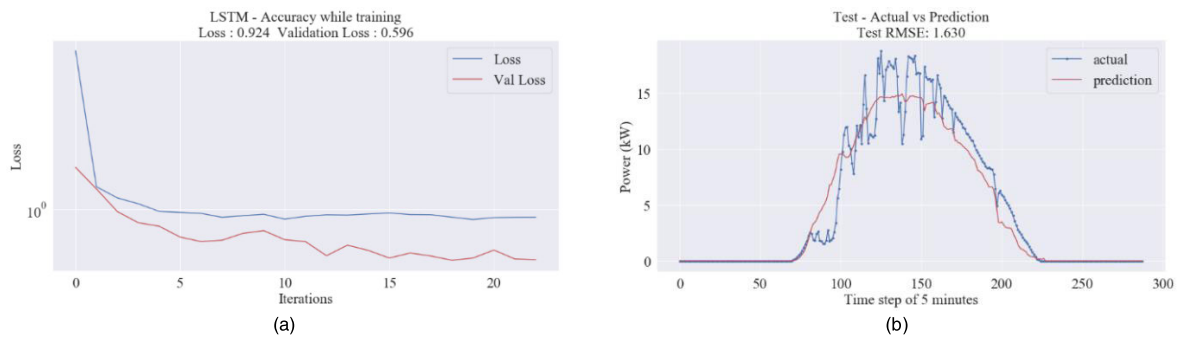


FIGURE 38. Test Case #11 LSTM with 50 Days Dataset (MFU) (a) Accuracy (b) Actual vs Prediction.

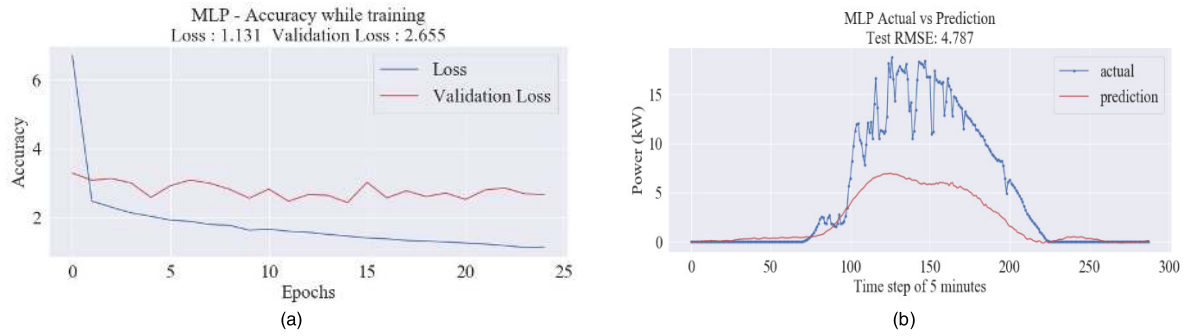


FIGURE 39. Test Case #12 MLP with 50 Days Dataset (MFU) (a) Accuracy (b) Actual vs Prediction.

TABLE 7. Dataset lengths for test cases #1 to #12.

Case	Model	Dataset	Length
#1	LSTM	KHH	1 Month
#2	MLP	KHH	1 Month
#3	LSTM	KHH	3 Months
#4	MLP	KHH	3 Months
#5	LSTM	KHH	6 Months
#6	MLP	KHH	6 Months
#7	LSTM	KHH	9 Months
#8	MLP	KHH	9 Months
#9	LSTM	MFU	30 Days
#10	MLP	MFU	30 Days
#11	LSTM	MFU	50 Days
#12	MLP	MFU	50 Days

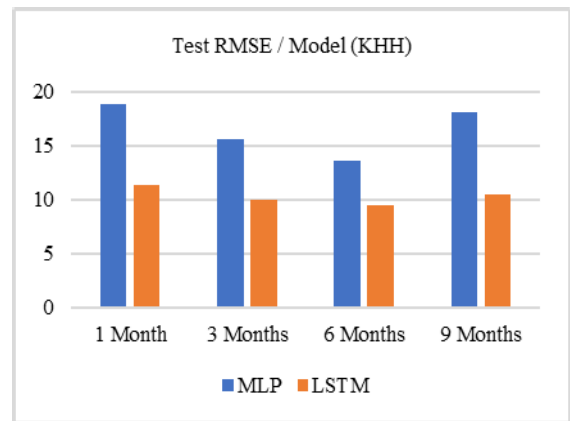


FIGURE 40. Accuracy between LSTM and MLP for KHH.

TABLE 8. Results of tests case #1 to #12.

#	Model	Dataset	Day	Loss	Validation Loss	Test RMSE
#1	LSTM	KHH	30	31.635	33.258	11.343
#2	MLP	KHH	30	48.735	227.77	18.912
#3	LSTM	KHH	90	62.802	96.434	10.009
#4	MLP	KHH	90	99.459	285.00	15.666
#5	LSTM	KHH	180	88.560	61.480	9.469
#6	MLP	KHH	180	150.44	202.17	13.014
#7	LSTM	KHH	270	79.318	25.761	10.534
#8	MLP	KHH	270	341.30	229.18	18.648
#9	LSTM	MFU	30	0.747	0.828	0.828
#10	MLP	MFU	30	0.937	9.591	1.653
#11	LSTM	MFU	50	0.924	0.596	1.630
#12	MLP	MFU	50	1.131	2.655	4.787

MFU dataset shows better quality compared with KHH. The difference is huge and is probably caused by the quality of data between the two datasets Fig. 41.

When both models are trained with MFU datasets, they produce better metrics and predictions that when trained on KHH datasets. Moreover, an also conclude a difference in term of accuracy between both models MLP and LSTM, regardless which dataset is trained. For the purpose of our tests, we will keep working with LSTM trained on MFU datasets since the combination produced the best results.

### C. SCENARIOS BASED ON WEATHER PROFILES

Following the conclusion of the first evaluation, the LSTM Model will be trained with MFU dataset during the second evaluation and will be tested with different weather profiles, as scenarios for one day weather forecast. Several weather profiles have been observed and selected from MFU dataset and preprocessed before to produce the predictions.

Based on visual observation for each single day in MFU dataset, different scenarios based on different weather conditions have been selected: Sunny, Cloudy, Rainy, Low Light, Half Rainy/Half Sunny, Half Sunny/Half Cloudy.

A sunny weather is characterized by no presence of clouds in the sky, resulting by a smooth irradiance trend from sunrise to sunset. A cloudy weather is characterized by the entire sky covered with cloud, resulting of sudden drops of sunlight over the day. A rainy weather is characterized by the presence of rain, resulting of more noise on the power trend compared with a cloudy weather. A low-light weather is characterized by a very low irradiance line over the day, which can because by a lower elevation and azimuth as illustrated in Fig. 42. The different datasets for each weather profile are shown in Fig. 43.

Finally, two other mixed type of weather were selected, when half of the day is sunny while the other half is either cloudy or rainy. While observing the power signal over one

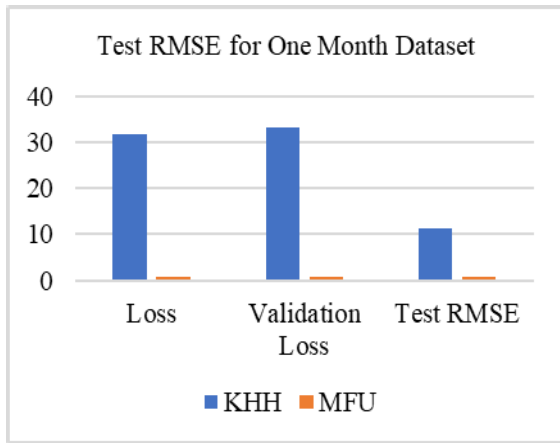


FIGURE 41. Comparison MFU and KHH.

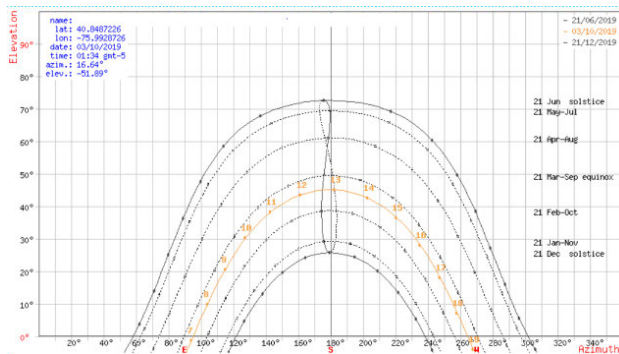


FIGURE 42. Sun Elevation and Azimuth over the year.

TABLE 9. Dataset for test cases #13~#18.

Dataset	MFU
Training	30 days
Validation	20% of Training Datasets
Test	1 Day

day period, we selected one day for each weather profile and listed them in Fig. 43.

LSTM will be trained with one-month dataset from MFU Dataset as presented in Table 9.

Considering the high number of networks configurations possible to configure the LSTM, the following tables, figures and charts display the results and predictions as an example of the expected output values using the model described in the previous chapter.

### 1) RESULTS

Table 10 presents loss, validation loss and validation RMSE values. Chart visualization for the loss and validation loss for each iteration is shown in Fig. 44.

When making a prediction for each case and observing the RMSE value, we notice best results for Sunny and Low Light weather profiles. These two weather profiles have smoother

TABLE 10. Loss, validation loss and RMSE result.

MFU Dataset	Metrics
Loss	0.348
Validation Loss	0.380
Validation RMSE	0.444

TABLE 11. Test RMSE for each case.

Case	Weather Scenario	RMSE
#13	Sunny	0.230
#14	Cloudy	0.724
#15	Rainy	0.646
#16	Low Light	0.160
#17	Half Rainy, Half Sunny	0.604
#18	Half Sunny, Half Cloudy	0.655

TABLE 12. Table metrics for model tuning.

Test Case	Activation	Nodes
#19	Sigmoid	50
#20	Sigmoid	100
#21	Sigmoid	150
#22	Sigmoid	200
#23	Relu	50
#24	Relu	100
#25	Relu	150
#26	Relu	200
#27	Tanh	50
#28	Tanh	100
#29	Tanh	150
#30	Tanh	200

lines and therefore less difficulty to be predicted when compared with other profiles which feature unpredictable patterns such as sudden drops.

### 2) EVALUATION METRICS

We can observe model predicted forecasts depending on the different weather scenarios for the trained model with 30 days for MFU in Fig. 45 to Fig. 50. One can verify in detail the prediction for every scenario. When the model is trained with MFU dataset, the network forecasting results can successfully approximate to the expected outputs and the intra-hour ramping is well captured. Forecast accuracy decays in the middle variations of the day but current model already shows good results, despite the fact the model is not tuned and using default parameters for activation function.

A prediction has been made with each dataset for each weather profile. Table 11 shows the results for each weather profile. We can observe best values for Sunny and Low light weather profiles, which confirm earlier observations.

### D. TUNING OF HYPERPARAMETERS

The next evaluation is to determine the best hyperparameters to tune the model. To conduct the test, two parameters will



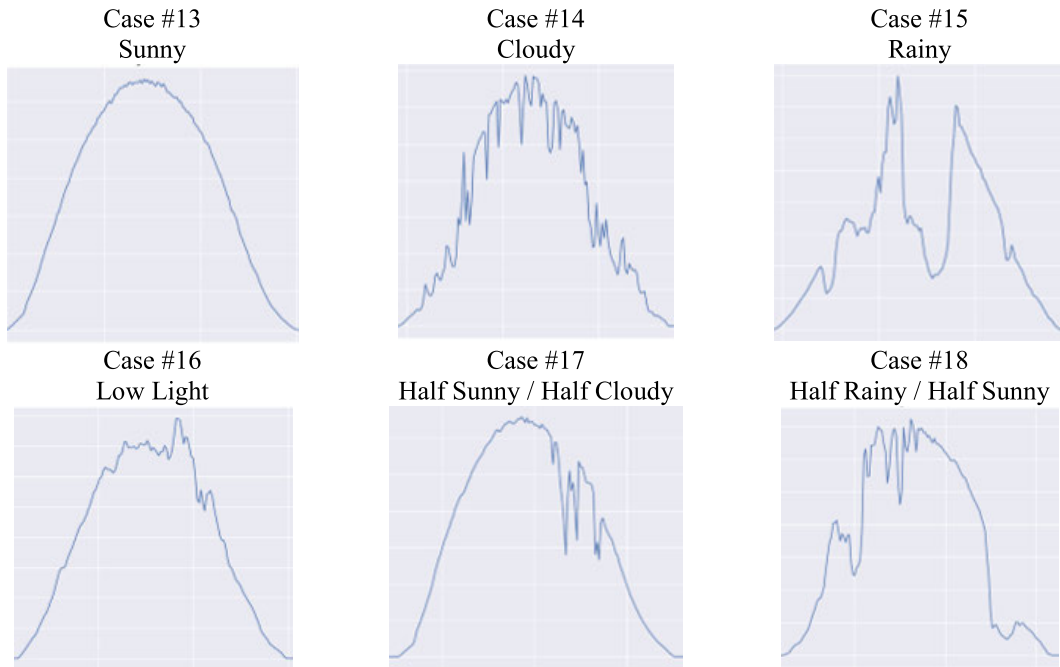


FIGURE 43. Different possible scenario for 1 day forecast weather conditions.

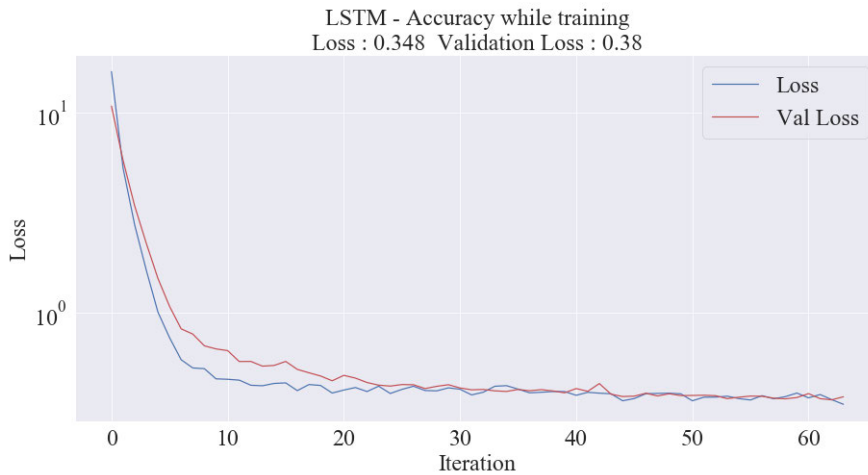


FIGURE 44. Loss and Validation Loss for the trained model with 30 days.

be modified for each test case: the activation function and the number of nodes.

In the previous evaluation, we kept the activation function by default to Relu and set the number of nodes to 100. In order to observe variations in term of accuracy in the metrics and determine what would be the best settings for these two parameters, we defined 12 test cases as described in Table 12.

When running the 12 test cases and producing loss and validation loss charts during each test, we noticed different patterns for each different activation function.

When model is trained with Sigmoid activation function, charts show underfitting as training loss is larger than validation loss. This phenomenon can be observed for 50, 100, 150 and 200 nodes in Fig. 51 to Fig. 54.

When model is trained with Relu activation function, we can notice underfitting with 50 nodes and overfitting for 150 and 200 nodes. However, a satisfying pattern is noticeable with 100 nodes as training and validation loss are close and final values identical. Results can be observed in Fig. 55 to Fig. 58.

When model is trained with activation function Tanh, we can observe overfitting as validation loss values are higher

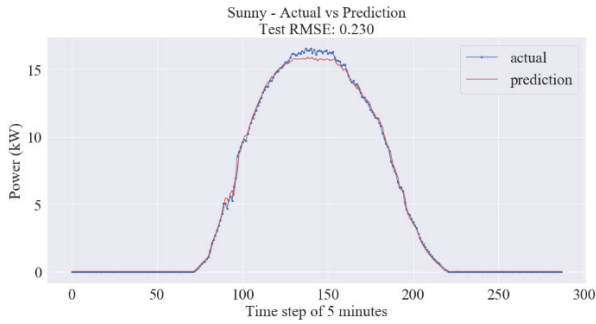


FIGURE 45. Case #13 Sunny.

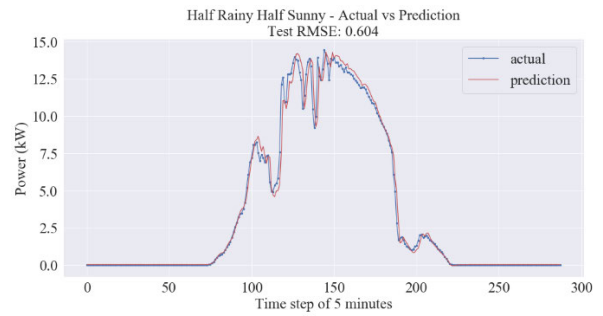


FIGURE 49. Case #17 - Half Rainy / Half Sunny.

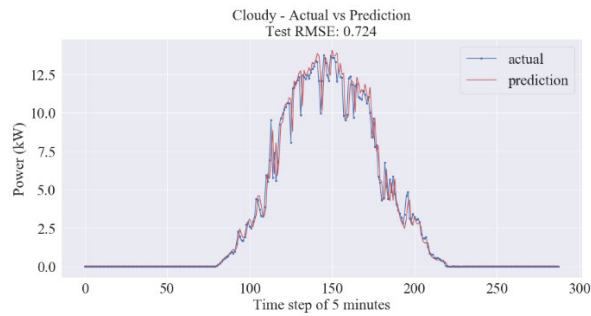


FIGURE 46. Case #14 - Cloudy.

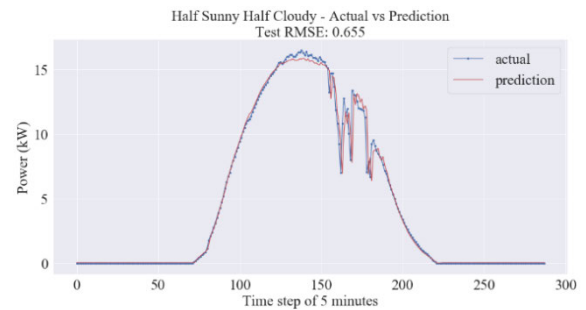


FIGURE 50. Case #18 - Half Sunny, Half Cloudy.

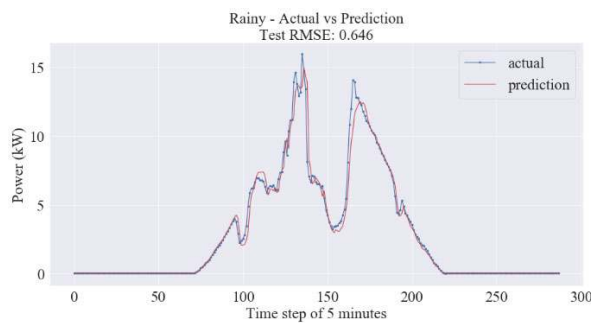


FIGURE 47. Case #15 - Rainy.

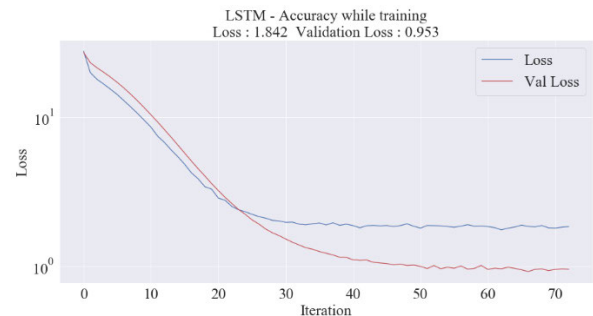


FIGURE 51. Case #19: Sigmoid Activation Function with 50 Nodes.

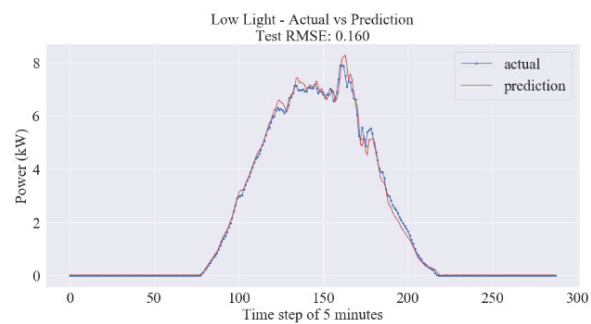


FIGURE 48. Case #16 - Low Light.

than training loss with 100, 150 and 200 nodes. A good pattern is noticeable when the number of nodes is equal to 50 nodes. Results can be observed in Fig. 59 to Fig. 62.

From the loss and validation loss charts, we can deduct that best results would be found when the number of nodes is set to 100 nodes with Relu as activation function or with 50 nodes and the activation function set to Tanh. Further checking with RMSE are necessary to confirm these results and to determine which of the two show best performances.

### 1) RESULTS

Table 13 shows all results by activation functions and Table 14 shows all results by number of nodes.

### 2) COMPARISON OF METRICS

By synthesizing the results from the two tables best average test RMSE are obtained in Table 15.

LSTM with 50 to 200 nodes and Tanh activation function delivers best RMSE among all tested cases.

TABLE 13. Comparison of metrics by activation function.

Activation Function	Sigmoid				Relu				Tanh			
	#19	#20	#21	#22	#23	#24	#25	#26	#27	#28	#29	#30
Test Cases	#19	#20	#21	#22	#23	#24	#25	#26	#27	#28	#29	#30
Nodes	50	100	150	200	50	100	150	200	50	100	150	200
Loss	1.842	1.227	0.945	0.778	0.565	0.430	0.338	0.281	0.364	0.319	0.268	0.289
Validation Loss	0.953	0.801	0.722	0.641	0.400	0.411	0.375	0.363	0.388	0.409	0.376	0.426
Validation RMSE	0.672	0.675	0.663	0.610	0.472	0.450	0.433	0.420	0.411	0.462	0.431	0.000
Weather Scenario	Test RMSE				Test RMSE				Test RMSE			
Sunny	1.247	0.996	0.860	0.761	0.319	0.341	0.237	0.240	0.364	0.264	0.263	0.263
Cloudy	0.772	0.827	0.829	0.814	0.752	0.729	0.737	0.726	0.726	0.743	0.737	0.754
Rainy	0.712	0.754	0.765	0.733	0.631	0.642	0.636	0.626	0.627	0.624	0.644	0.646
Low Light	0.312	0.342	0.314	0.282	0.221	0.236	0.168	0.137	0.147	0.176	0.139	0.206
Half Rainy, Half Sunny	0.692	0.714	0.730	0.695	0.632	0.626	0.608	0.616	0.619	0.628	0.615	0.640
Half Sunny, Half Cloudy	1.226	1.058	0.966	0.896	0.679	0.670	0.664	0.637	0.686	0.655	0.653	0.670
Average	0.827	0.782	0.744	0.697	0.539	0.540	0.508	0.497	0.528	0.515	0.508	0.530

TABLE 14. Comparison of metrics by number of nodes.

Nodes	50			100			150			200		
	Sigmoid	Relu	Tanh	Sigmoid	Relu	Tanh	Sigmoid	Relu	Tanh	Sigmoid	Relu	Tanh
Loss	1.842	0.565	0.364	1.227	0.430	0.319	0.945	0.338	0.268	0.778	0.281	0.289
Validation Loss	0.953	0.400	0.388	0.801	0.411	0.409	0.722	0.375	0.376	0.641	0.363	0.426
Validation RMSE	0.672	0.472	0.411	0.675	0.450	0.462	0.663	0.433	0.431	0.610	0.420	0.000
Weather Scenario	Test RMSE			Test RMSE			Test RMSE			Test RMSE		
Sunny	1.247	0.319	0.364	0.996	0.341	0.264	0.860	0.237	0.263	0.761	0.240	0.263
Cloudy	0.772	0.752	0.726	0.827	0.729	0.743	0.829	0.737	0.737	0.814	0.726	0.754
Rainy	0.712	0.631	0.627	0.754	0.642	0.624	0.765	0.636	0.644	0.733	0.626	0.646
Low Light	0.312	0.221	0.147	0.342	0.236	0.176	0.314	0.168	0.139	0.282	0.137	0.206
Half Rainy, Half Sunny	0.692	0.632	0.619	0.714	0.626	0.628	0.730	0.608	0.615	0.695	0.616	0.640
Half Sunny, Half Cloudy	1.226	0.679	0.686	1.058	0.670	0.655	0.966	0.664	0.653	0.896	0.637	0.670
Average	0.827	0.539	0.528	0.782	0.540	0.515	0.744	0.508	0.508	0.697	0.497	0.530

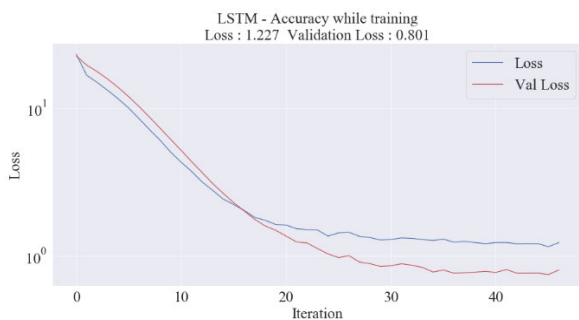


FIGURE 52. Case #20: Sigmoid Activation Function with 100 Nodes.

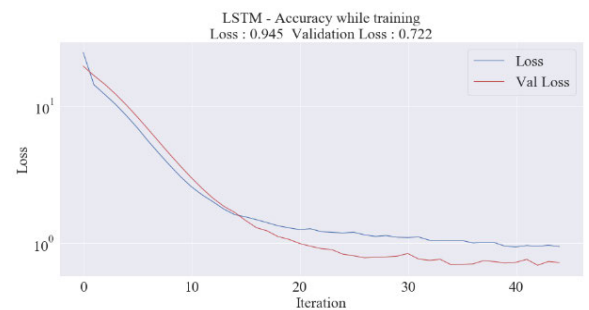


FIGURE 53. Case #21: Sigmoid Activation Function with 150 Nodes.

IV. DISCUSSIONS

This paper presents a single LSTM model to compare with MLP neural network and for performances. Others LSTM models could be compared for instance, with multivariate inputs or different number of past observations.

The results presented in this paper are based on a single run to train the neural networks and to produce predictions.

TABLE 15. Best average test RMSE.

50 Nodes	100 Nodes	150 Nodes	200 Nodes
Tanh	Tanh	Tanh	Relu
0.528	0.515	0.508	0.497

Multiple runs to produce results by average would be needed to consolidate the results presented in this paper.

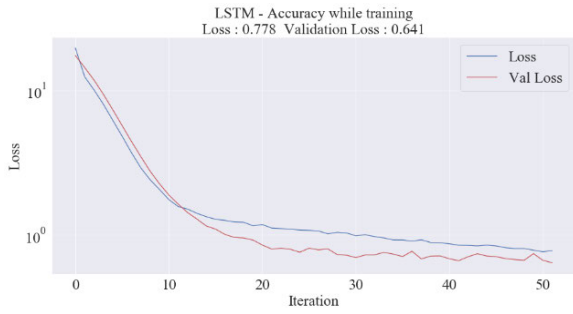


FIGURE 54. Case #22: Sigmoid Activation Function with 200 Nodes.



FIGURE 58. Case #26: Relu Activation Function with 200 Nodes.

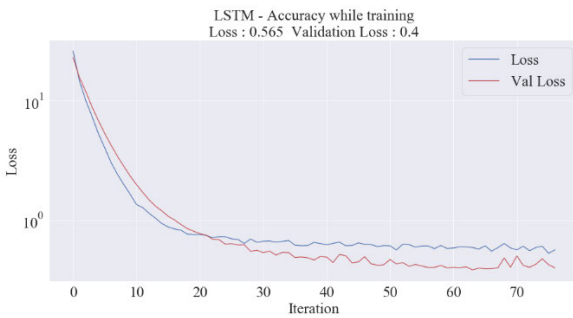


FIGURE 55. Case #23: Relu Activation Function with 50 Nodes.

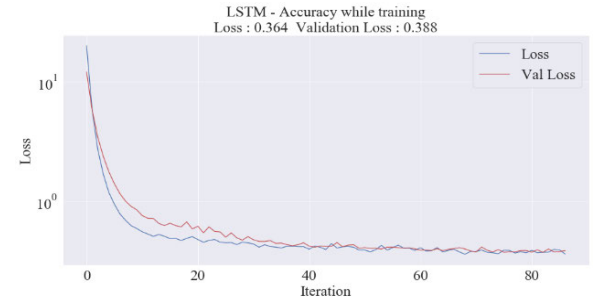


FIGURE 59. Case #27: Tanh Activation Function with 50 Nodes.

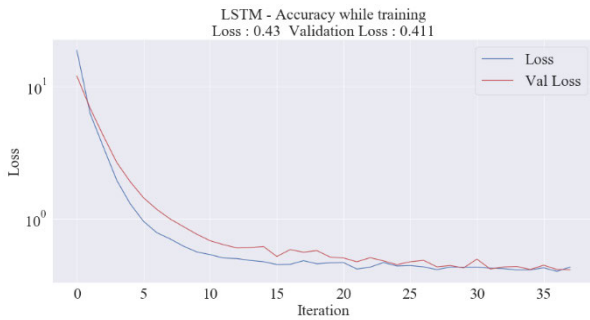


FIGURE 56. Case #24: Relu Activation Function with 100 Nodes.



FIGURE 60. Case #28: Tanh Activation Function with 100 Nodes.

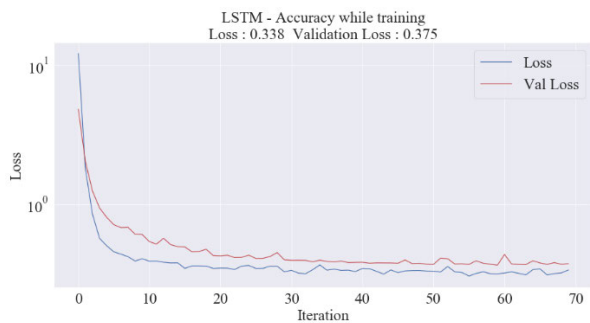


FIGURE 57. Case #25: Relu Activation Function with 150 Nodes.

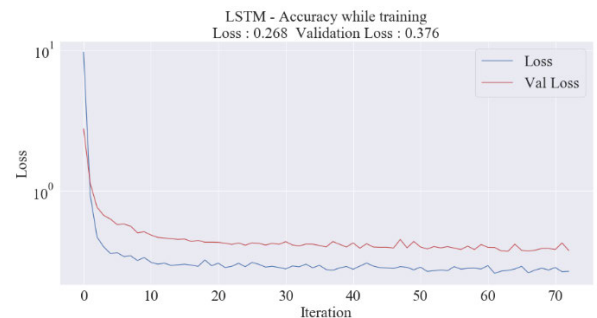


FIGURE 61. Case #29: Tanh Activation Function with 150 Nodes.

During the experiments, MFU and KHH datasets showed differences in term of performance. LSTM when trained with MFU datasets showed better RMSE than when trained with KHH. A further data analysis could be done to compare both

datasets KHH and MFU, including the calculation of data degradation by comparing power with clear sky model for example. Also, the experiments only tuned the number of neurons and the activation function. Other hyperparameters could be also tuned to reach better performances.

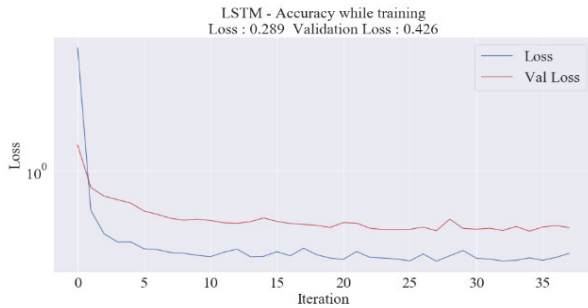


FIGURE 62. Case #30: Tanh Activation Function with 200 Nodes.

## V. CONCLUSION

The author proposes an architecture of machine learning methodologies based on LSTM and MLP with various size of train datasets from two PV sites located in KHH (Kaohsiung) and MFU (Thailand) respectively. The initial test shows that RMSEs of evaluation metrics for 12 cases have no significant difference in term of the size of train dataset. However, it is clear to mention that LSTM is superior to MLP for the forecasting under this test platform. Moreover, it is worth highlighting that the RMSEs of the same LSTM model based on 1-month train dataset of KHH and MFU are 1.653 and 0.828 respectively. After the observation of raw data between KHH and MFU, we can assume that the data quality is likely the major cause to lead to the different test result. Nevertheless, LSTM does perform the promising result for one day ahead solar power forecasting. After the initial test, the further test is focused on LSTM with the cases of weather conditions based on MFU train dataset. The weather conditions available in the train dataset can be categorized into Sunny, Cloudy, Rainy, Low Light, Sunny-Cloudy and Rainy-Sunny. The result shows that the low RMSEs of 0.230, 0.724, 0.646, 0.160, 0.604, 0.655 respectively. When the model is trained with MFU dataset, the network forecasting results can successfully approximate to the expected outputs and the intra-hour ramping is well captured. Forecast accuracy decays in the middle variations of the day but current model already shows good results, despite the fact the model is not tuned and using default parameters for activation function that will be the last test and described. The last evaluation is to tune the model and determine the best hyperparameters for activation functions and number of hidden nodes. The result shows that the best average RMSE is 0.497 under the conditions of 200 hidden nodes and activation function Tanh.

The further study of LSTM for multi-variate time series forecasting has not been tested and developed. This approach will require the weather information from weather station at PV site or the third-party weather websites.

## REFERENCES

- [1] J. Mubiru, "Predicting total solar irradiation values using artificial neural networks," *Renew. Energy*, vol. 33, no. 10, pp. 2329–2332, Oct. 2008.
- [2] I. S. Isa, S. Omar, Z. Saad, N. M. Noor, and M. K. Osman, "Weather forecasting using photovoltaic system and neural network," in *Proc. 2nd Int. Conf. Comput. Intell., Commun. Syst. Netw.*, Jul. 2010, pp. 96–100.
- [3] A. Mellit and A. M. 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," *Sol. Energy*, vol. 84, no. 5, pp. 807–821, May 2010.
- [4] Y. Zhang, G. P. Chen, O. P. Malik, and G. S. Hope, "An artificial neural network based adaptive power system stabilizer," *IEEE Trans. Energy Convers.*, vol. 8, no. 1, pp. 71–77, Mar. 1993.
- [5] H. A. Malki, N. B. Karayiannis, and M. Balasubramanian, "Short-term electric power load forecasting using feedforward neural networks," *Expert Syst.*, vol. 21, no. 3, pp. 157–167, Jul. 2004.
- [6] A. Elamim, B. Hartiti, A. Barhdadi, A. Haibaoui, L. Abderrazak, and P. Thevenin, "Photovoltaic output power forecast using artificial neural networks," *J. Theor. Appl. Inf. Technol.*, vol. 96, no. 15, pp. 5116–5126, Aug. 2018.
- [7] S. P. Durrani, S. Balluff, L. Wurzer, and S. Krauter, "Photovoltaic yield prediction using an irradiance forecast model based on multiple neural networks," *J. Modern Power Syst. Clean Energy*, vol. 6, no. 2, pp. 255–267, Mar. 2018.
- [8] S. Watetakarn and S. Premrudeepreechacharn, "Forecasting of solar irradiance for solar power plants by artificial neural network," in *Proc. IEEE Innov. Smart Grid Technol. Asia (ISGT ASIA)*, Nov. 2015, pp. 1–5.
- [9] M. Gupta, L. Jin, and N. Homma, *Static and Dynamic Neural Networks: From Fundamentals to Advanced Theory*. Hoboken, NJ, USA: Wiley, 2004.
- [10] P. Mandal, S. T. S. Madhira, A. U. Haque, J. Meng, and R. L. Pineda, "Forecasting power output of solar photovoltaic system using wavelet transform and artificial intelligence techniques," *Procedia Comput. Sci.*, vol. 12, pp. 332–337, Jan. 2012.
- [11] A. Yona, T. Senjyu, and T. Funabashi, "Application of recurrent neural network to Short-Term-Ahead generating power forecasting for photovoltaic system," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, Jun. 2007, pp. 1–6.
- [12] M. Benganem and A. Mellit, "Radial basis function network-based prediction of global solar radiation data: Application for sizing of a stand-alone photovoltaic system at al-Madinah, Saudi Arabia," *Energy*, vol. 35, no. 9, pp. 3751–3762, Sep. 2010.
- [13] A. Yona, T. Senjyu, and T. Funabashi, "Application of recurrent neural network to short-term-ahead generating power forecasting for photovoltaic system," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, Jun. 2007, pp. 1–6.
- [14] A. Sözen, E. Arcaklio lu, M. Özalp, and K. EG, "Use of artificial neural networks for mapping of solar potential in Turkey," *Appl. Energy*, vol. 77, no. 3, pp. 273–286, Mar. 2004.
- [15] Z. Wang, F. Wang, and S. Su, "Solar irradiance short-term prediction model based on BP neural network," *Energy Procedia*, vol. 12, pp. 488–494, Dec. 2011.
- [16] M. Ding, L. Wang, and R. Bi, "An ANN-based approach for forecasting the power output of photovoltaic system," *Procedia Environ. Sci.*, vol. 11, pp. 1308–1315, Jan. 2011.
- [17] F. Bizzarri, M. Bongiorno, A. Brambilla, G. Gruosso, and G. S. Gajani, "Model of photovoltaic power plants for performance analysis and production forecast," *IEEE Trans. Sustain. Energy*, vol. 4, no. 2, pp. 278–285, Apr. 2013.
- [18] P. Smolensky, M. C. Mozer, and D. E. Rumelhart, *Mathematical Perspectives on Neural Networks*. New York, NY, USA: Psychology Press, 2013.
- [19] A. Suliman and Y. Zhang, "A review on back-propagation neural networks in the application of remote sensing image classification," *J. Earth Sci. Eng.*, vol. 5, no. 1, pp. 52–65, Jan. 2015.
- [20] M. Alsmadi, K. Omar, and S. A. M. Noah, "Back propagation algorithm: The best algorithm among the multi-layer perceptron algorithm," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 9, no. 4, pp. 378–383, Apr. 2009.
- [21] R. Srivastava, A. N. Tiwari, and V. K. Giri, "Forecasting of solar radiation in india using various ANN models," in *Proc. 5th IEEE Uttar Pradesh Sect. Int. Conf. Electr., Electron. Comput. Eng. (UPCON)*, Nov. 2018, pp. 1–6.
- [22] M. A. Behrang, E. Assareh, A. Ghanbarzadeh, and A. R. Noghrehabadi, "The potential of different artificial neural network (ANN) techniques in daily global solar radiation modeling based on meteorological data," *Sol. Energy*, vol. 84, no. 8, pp. 1468–1480, Aug. 2010.
- [23] 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, Mar. 2018.
- [24] M. Gao, J. Li, F. Hong, and D. Long, "Day-ahead power forecasting in a large-scale photovoltaic plant based on weather classification using LSTM," *Energy*, vol. 187, Nov. 2019, Art. no. 115838.

- [25] H. Zhou, Y. Zhang, L. Yang, Q. Liu, K. Yan, and Y. Du, "Short-term photovoltaic power forecasting based on long short term memory neural network and attention mechanism," *IEEE Access*, vol. 7, pp. 78063–78074, 2019.
- [26] Y. Yu, J. Cao, and J. Zhu, "An LSTM short-term solar irradiance forecasting under complicated weather conditions," *IEEE Access*, vol. 7, pp. 145651–145666, 2019.
- [27] F. Wang, Z. Xuan, Z. Zhen, K. Li, T. Wang, and M. Shi, "A day-ahead PV power forecasting method based on LSTM-RNN model and time correlation modification under partial daily pattern prediction framework," *Energy Convers. Manage.*, vol. 212, May 2020, Art. no. 112766.
- [28] M. G. De Giorgi, M. Malvoni, and P. M. Congedo, "Comparison of strategies for multi-step ahead photovoltaic power forecasting models based on hybrid group method of data handling networks and least square support vector machine," *Energy*, vol. 107, pp. 360–373, Jul. 2016.
- [29] K.-R. Müller, A. J. Smola, G. Rätsch, B. Schölkopf, J. J. Kohlmorgen, and V. Vapnik, "Predicting time series with support vector machines," in *Proc. Int. Conf. Artif. Neural Netw.*, Munich, Germany, 1997, pp. 999–1004.
- [30] I. A. Khan, A. Akber, and Y. Xu, "Sliding window regression based short-term load forecasting of a multi-area power system," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2019, pp. 1–5.



doing his research work for the renewable energy management platform.

**CHUN-HUNG LIU** was born in Kaohsiung, Taiwan, in 1969. He received the B.S. degree in electrical engineering from the National Taiwan University of Science and Technology, in 1994, and the Ph.D. degree in electrical power engineering from the National Taiwan University of Science, in 2019. From 2013 to 2019, he was doing the research works in the field of smart grids and joined the smart grids related research projects with Taiwan Power Company. He is currently



research on power systems protection, arc flash analysis and electrical safety, electric power quality, and IEC 61850 for microgrid protection.

**JYH-CHERNG GU** (Member, IEEE) was born in 1958. He received the B.S. degree from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 1984, and the M.S. and Ph.D. degrees in electrical engineering from the University of Texas at Arlington, Arlington, TX, USA, in 1987 and 1992, respectively. He is currently a Professor with the Department of Electrical Engineering, National Taiwan University of Science and Technology. He has been involved in



Magong, Penghu, Taiwan. His research interests include power systems protection, arc flash analysis and electrical safety, condition-based maintenance (CBM), and IEC 61850 for microgrid protection.

**MING-TA YANG** (Member, IEEE) was born in 1968. He received the Ph.D. degree from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 2006. From 2014 to 2020, he was a Professor with the Department of Electrical Engineering, St. John's University of Science and Technology, New Taipei, Taiwan. He is currently an Associate Professor with the Department of Electrical Engineering, National Penghu University of Science and Technology,

...