Highlights

Using Neural Network to Predict Diffuse Fraction k_d (Ratio of the Diffuse-to-Global Solar Radiation)*

Oscar Alberto Santos Muñoz

- Feedforward Neural Network was created with the aim of being sub-neural network from other works since the outputs values can be used as input values in other research.

- The first time a multi-class classification solution is used as if it was regression solution for the diffuse fraction forecast. Despite model accuracy very accuracy forecasts were obtained.

- Comparable results with other ANN models.

- Real (observed) and predicted (forecast) values were compared.

- Hourly values of irradiation measurements of Nagaoka city, Niigata, Japan, recorded during 1981–2000 were used.

Using Neural Network to Predict Diffuse Fraction k_d (Ratio of the Diffuse-to-Global Solar Radiation)

Oscar Alberto Santos Muñoz[a,b,1,*]

[a] *Energy Engineering Laboratory, Niigata 940-2188, Japan*
[b] *Universidad de Guanajuato, 36885 Guanajuato, Mexico*

Abstract

In this paper, an Artificial Neural Network was used to predict hourly the diffuse fraction at Nagaoka city in Japan, using global radiation, date and time as input parameters. The data set used was obtained by the system (Japan Meteorological Agency, AMeDAS) for the period (1981-2000), of which 15 years (1981-1995) were taken for training, 2 years for validation (1996-1997) and 3 years as an isolated data set (1998-2000) for testing. The model did not show perfect accuracy, however it made very close predictions, due to the way the neural network was created. The forecast was made by multi-class classification instead of non-linear regression with Feedforward Neural Network architecture composed of 3 hidden layers with 128 neurons each. The aim of this research is to develop a model to correlate diffuse fraction and global irradiance and thus provide researchers a sub-neural network in their neural networks since these results can be used as inputs to other neural networks, this will contribute to create more accurate models with a higher number of inputs and that they can improve certain applications and solar devices.

Keywords: Artificial neural networks, deep learning, diffuse fraction, diffuse horizontal irradiance, global horizontal irradiance, machine learning, meteorological parameters, multi-class classification, predictive analytics.

1. Introduction

The use of renewable energy is rapidly increasing and the number of technologies that are emerging to replace grid electricity continues to grow. For this reason, the prediction of certain variables has become a necessity.

The literature shows that there are many analytical models for making predictions in solar energy field, most of them correlate hourly diffuse fraction k_d and the hourly clearness index k_T (Erbs et al., 1982).

Neural networks have been a focus of interest since they emerged as a solution to time series prediction problems. In addition, organizations are looking for more precision in their processes and this is achieved through information collected in their processes.

The way to collect information is through monitoring systems, either in the photovoltaic system itself or in research centers. This information can be used to make forecasts. Most of these systems monitor electrical and meteorological variables. And according to Yang et al. (2018), the most important one is solar irradiance, because useful relationships can be calculated from it, such as the diffuse fraction (k_d), on which this article is focused.

Certainly, there are measurement errors, inconsistencies and bad approaches due to small measurement issues, but a good model is achieved from high quality data (Pełech-Pilichowski, 2018; Pełech-Pilichowski and Duda, 2008). Past work has shown that accurate predictions are achieved through past data from the model or function to be predicted (Chandra, 2018; Chakraborty et al., 1992).

There are research centers that are dedicated to collect data, some of them measure the radiation from the sun each hour of the day. Due to the high costs that this generates, there is not an accurate record for all areas. Radiation varies according to the conditions of each site and therefore measurements are limited, resulting in poor approximations or lack of data. Over time, models have been created to correlate variables (Zhou et al., 2019) such as Global Horizontal Irradiance (GHI) and Diffuse Horizontal Irradiance (DHI).

Once, the correlations have being obtained, is easier to create models over the collected data rather than measure again with high costs. These relations help to establish solar models for certain applications such as the placement of photovoltaic panels and sun tracking. The simulations and performance on solar devices can be improved as well (Zhou et al., 2019), knowing for example k_d.

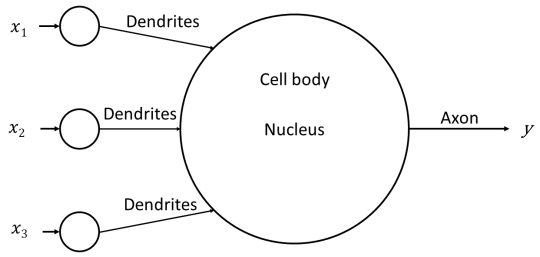
An artificial neural network (ANN) is a way of solving non-linear relationships in Multi-Inputs-Multi-Outputs (MIMO) systems. Neural networks are interconnections between neurons that are connected to each other and these connections have assigned weights and biases (Meenal and Selvakumar, 2017). The branches that come out of a biological neuron are called dendrites, which receive the afferences from other neurons or

*This work was supported by grants from the Nagaoka University of Technology by JASSO Program and the University of Guanajuato.

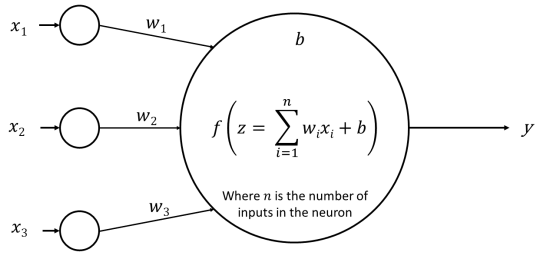
*Corresponding author
Email address: oa.santosmunoz@ugto.mx (Oscar Alberto Santos Muñoz)

[1] The code can be seen in the author's GitHub profile.

from receptor cells (Fox, 2014, p. 12). The same occurs in an ANN, the hidden layer neurons receive information from the dendrites in the input layer and produce a corresponding output if the sum is greater than threshold value (Meenal and Selvakumar, 2017), this value is the bias. This would be the equivalent of the cell body of the biological neuron, which contains the nucleus and functions as the metabolic center of the cell. And finally, the axon of the cell is the one that specializes in conducting impulses from the cell body to another neuron or effector cell (Fox, 2014, p. 12). This is the same as the outputs of each neuron in an ANN. Once the calculations with the weights and biases have been made the outputs go to other neurons or the final response of the network.



(a) Representation of the parts of a biological neuron



(b) Representation of an artificial network process

Figure 1: Analogy between a biological neuron and an artificial neuron.

This analogy with the biological neuron can be represented as shown in Fig. 1a, where x_i are the nerve stimulation received at the brain from a receptor cell and y is the response to that stimulus. Fig. 1b shows an artificial neural network process, where x_i are the inputs, w_i are the weights of each connection and b is the bias or threshold value.

The present paper is organized as follows: Section 2 describes the data set; it includes explanations in the general behavior of the data, the methods applied to pre-process the data and how the data set was handled before training. The architecture of the ANN is presented in Section 3. In Section 4 the methodology for the prediction algorithm is described as well as the explanation of how the multi-class classifier works as a non-linear regression. Section 5 shows the results and besides, statistical indicators are included with their respective results. Finally, Section 6 summarizes the work done, conclusions were drawn, and the advantages of this work were added in that Section.

2. Data

For this article, the solar irradiance data of Nagaoka city, Niigata, Japan, were used as a basis and as an example. The data set was obtained from the system (Japan Meteorological Agency, AMeDAS) for the period 1981-2000. This data set consists of 20 excel files (1 file by year), with the 8760 hours of the year information. It contains geographical information about the city of Nagaoka. In the excel files it is possible to find data of the Direct Normal Irradiation/Irradiance (DNI) ($kJ/(h \cdot m^2)$), DHI ($kJ/(h \cdot m^2)$), GHI ($kJ/(h \cdot m^2)$), zenith (°), azimuth (°), Direct Normal Irradiation/Irradiance with a tilt angle ($kJ/(h \cdot m^2)$), Diffuse Tilted Irradiance ($kJ/(h \cdot m^2)$), GTI ($kJ/(h \cdot m^2)$) and the tilt angle (°).

In this case only GHI will be used as one of the inputs of the ANN to predict k_d, however, if GTI values are available, they can replace GHI to predict k_d. And even though the data set that was used does have the GTI values, it was decided to use GHI because in most locations, the meteorological data consist of measurements of GHI. Besides DHI is not always available in the data set to calculate GTI with computational formulas that include Beam Irradiance as well (Tapakis et al., 2016a).

It should be mentioned that this data set is an example and this model can be used for any area, no matter if the weather and geographical conditions are different from the city of Nagaoka nor does it matter if the desired predictions turn out to be cloudy days, since this model only uses the date, time and solar radiation as inputs. Hence, if the information taken from the past is of good quality, good results will be obtained.

2.1. Understanding the behavior

First, it was considered prudent to conduct an analysis on the first year's data. This is in order to better understand our data set. Although every year shows an irradiance with values typical of the conditions of the date, all assume an average behavior over time. To appreciate this, it was considered only the first year to understand the behavior of irradiance.

2.1.1. First year analysis

The first thing that was done was to create a python list that contains information needed to plot the whole year. To do this, a list was created that repeats the number "1" 24 times, then 24 times the number "2", then 24 times the number "3" and so on until "365". This indicates that each 24 elements in the list belong to one day of the year, since in the data set the information is given by hour and it is required to section by day to have a record of the date.

In Fig. 2 it can be seen a plot of the days of the year vs irradiance. It includes 2 types of radiation, GHI and DHI, which are given by the data set.

The plot of the diffuse fraction had also been plotted; however, it was difficult to analyze its behavior given the high amount of data and the peaks of each day. To better appreciate and understand the data it was decided to take the first day analysis instead of the first year.

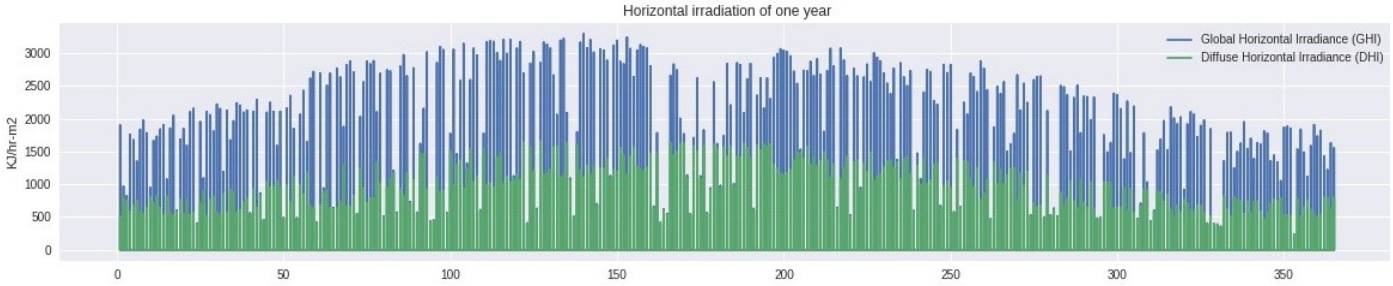


Figure 2: General radiation behaviour during the first year of analysis.

2.1.2. First day analysis

As before, a Python list was created for which a 24-hour account is repeated. Since item 25 in the data set corresponds to hour 1 of day 2, the list stores values from 1 to 24 every 24 items. This way, the data set can be sectioned in hours of the day according to the date without having to handle hours of the whole year.

Similarly, it was considered prudent to visualize the radiation again over the course of a day. This occasion is shown in Fig. 3 for both cases; GHI and DHI vs hours of the day and Diffuse fraction vs hours of the day.

2.1.3. Deep analysis on diffuse fraction

During the analysis process, the question arose whether GHI could be used as the only input, as the only training parameter. However, to visualize whether a linear regression could be used, the diffuse fraction vs GHI was plotted and DHI as a transparency parameter in that scatter plot type. When analyzing the results, it is clear that a linear regression cannot be used during the year nor during the day and therefore more input parameters must be taken for this prediction model. The results are shown in Fig. 4 for the first year and the first day as well.

In most of the works non-parametric regression analysis with ANN's is adopted to solve this kind of problem, but this paper shows that ANN multi-class classifier could solve and learn the key information pattern for the multivariate inputs even better (Tapakis et al., 2016b). Besides, the noise does not interfere much with the identification of patterns (Elminir et al., 2007), since the network itself adapts and learns these errors.

2.2. Data format

As mentioned in the section 2, a data set consisting of 20 years of accurate information on the city of Nagaoka was used. But in order to use all this information as an input to our network some adjustments had to be made. One is the rearrangement of data, other is the scaling that will be given to the wide range of values in the data and finally split into training, validation and test data.

2.2.1. Rearrange of data

To do this, each year of data that had been stored in an individual list was added to a single Python list. The list is 175200 items long, that is 8760 hours of the year multiplied by 20 years. Once the list with the 175200 elements was taken, the lists of days of the year and hours of the year that had been created to graph the first year and day were multiplied by 20 so that they contained 175200 elements as well. After that, numpy arrays were handled that facilitated the operations to create the input matrix. This matrix contains the GHI data, date and time.

In addition, another of the training parameters is the diffuse fraction, which was calculated directly in each Excel file for the 20 years and handled through the same process of storing the 20 years in a single Python list. For this case it was not necessary to add date and time in a matrix, simply the diffuse fraction will be used, since the input matrix already contains such parameters.

The first column represents GHI, the second is the date of the year, therefore, that column indicates the day of the year that represents that row, the third is the time of the day that the second column indicates, so, for example, during the hour 1 and 2 of day 1 there is a value of 0 for GHI, this because they are the measurements of the night; 12:00 am and 1:00 am respectively. Row 24 is the 24th hour and it is the same case as the night; 11:00 pm. Row 25 as mentioned in previous sections, hour 25 represents hour 1 of day 2, so it is 12:00 am, but this time of day 2 and again it is night and the value for GHI is 0. Now the example where the value for GHI is not 0, hour 13 of day 2 which corresponds to 2:00 pm of day 2 contains a value of 952.623046875.

The following is a representative example for the input matrix and the array that stores the diffuse fraction.

$$InputMatrix = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 2 \\ \vdots & \vdots & \vdots \\ 0 & 1 & 24 \\ 0 & 2 & 1 \\ \vdots & \vdots & \vdots \\ 952.623046875 & 2 & 13 \\ \vdots & \vdots & \vdots \\ 0 & 365 & 24 \end{pmatrix}$$

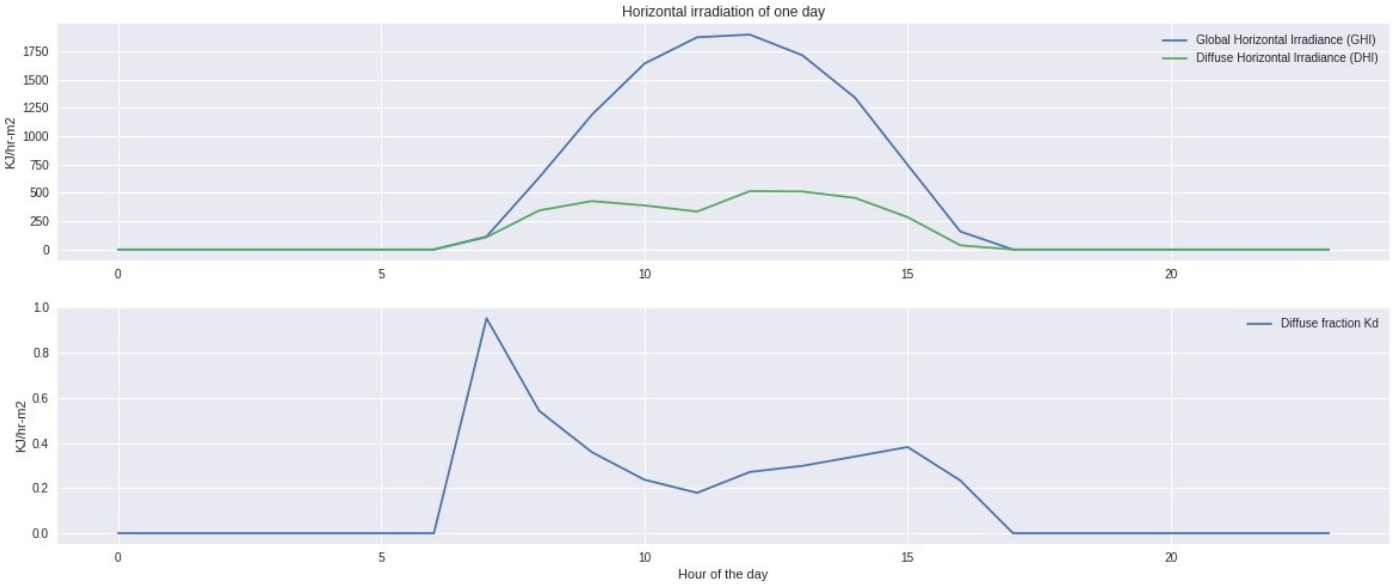


Figure 3: General radiation behaviour during the first day of analysis.

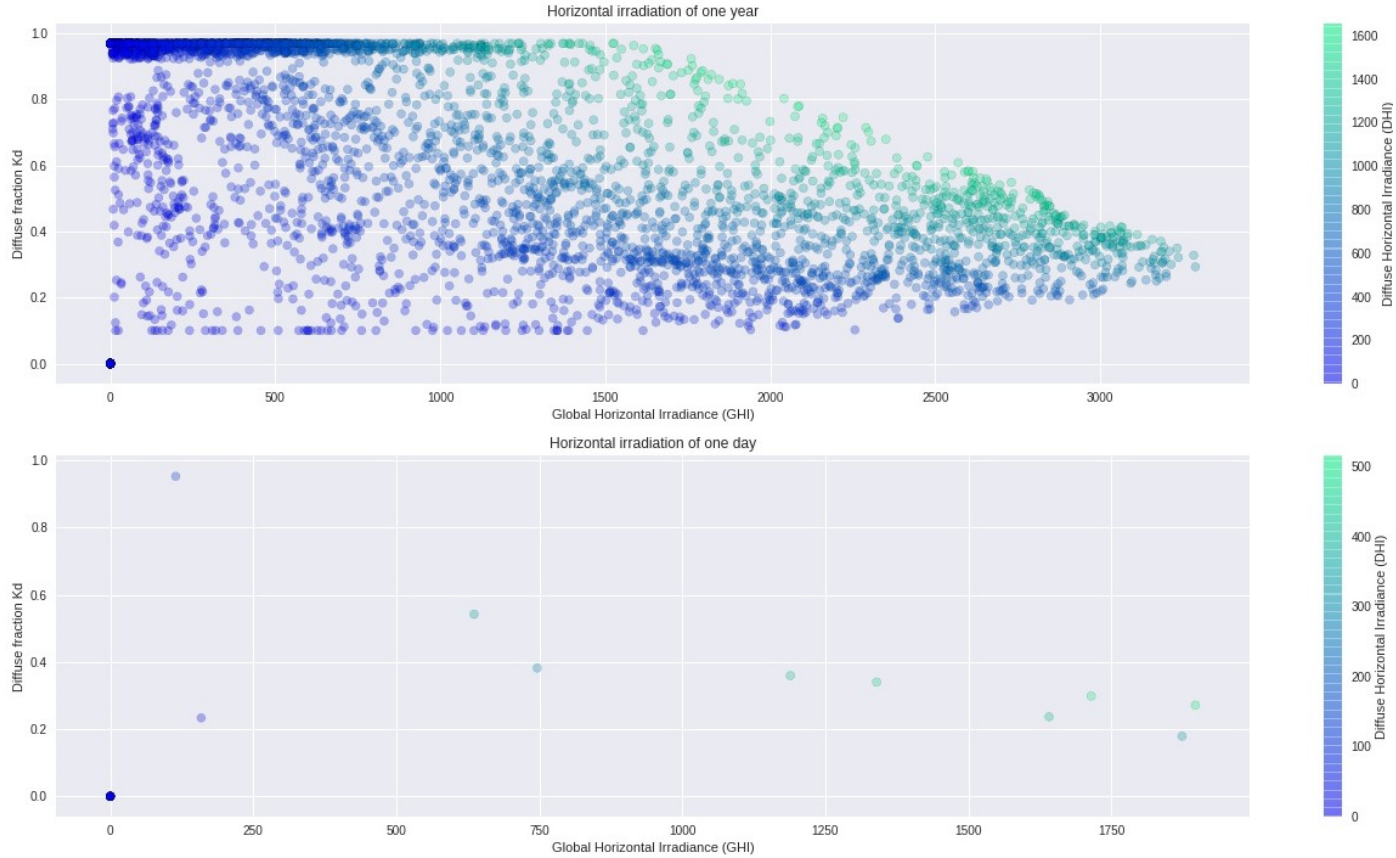


Figure 4: General behaviour of the diffuse fraction during the first year and first day of analysis.

$$k_d array = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0.810689492212087 \\ \vdots \\ 0 \end{pmatrix}$$

In the case of the array it works in the same way, element 37 represents the 13th hour of the second day, and since there is a radiation value for that row, there is a value for the diffuse fraction which is 0.810689492212087 for this row.

2.2.2. Data standardization

Standardization allow us to have better results on our model as well as being trained faster. Standardizing the data involves rescaling the distribution of values so that the mean of observed values is 0 and the standard deviation is 1. This is useful when your data has input values with differing scales. A value is standardized as follows:

$$z = \frac{(x - \mu)}{\sigma} \tag{1}$$

2.2.3. Splitting the data

To prevent overfitting is better to split in training data, validation data and test data. It was decided to separate into 15 years of training (1981-1995), 2 years for validation (1996-1997) and 3 years as independent data set (1998-2000) to test the ANN with data it did not see during the training. This separation may be changed in future work to try to train the network with more or less training data, more validation data or less test data. This could be easily changed with the library Sklearn available in python.

After separation, there are now 3 matrices and 3 arrays, a matrix of 131400 rows (15 times 8760 hours of the year) x 3 columns (GHI, Date and time) for the training data, a matrix of 17520 rows (2 times 8760) x 3 columns for the validation data and a matrix of 26280 rows (3 times 8760) x 3 columns for the test data. The sum of all the rows of these matrices is 175200 (20 times 8760) which are the rows that the matrix had before separation. Each row contains one element with GHI, Date and time. For the array is the same but without the date and time columns. Therefore, each row of the array contains one element only with GHI.

3. Architecture of the neural network

A Feedforward neural network was chosen as the ANN type. The architecture of the ANN can be seen in Fig. 5.

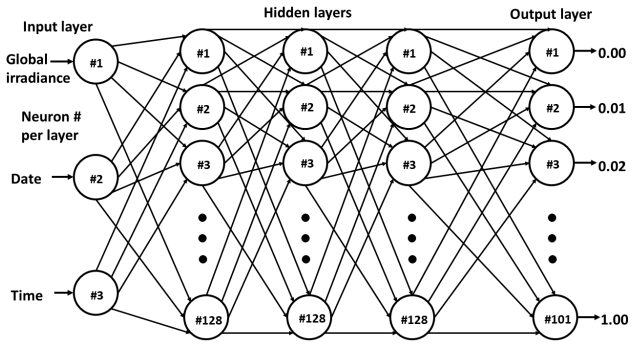


Figure 5: Architecture of the ANN.

The activation functions for the 3 hidden layers is ReLU and for the output layer is Softmax, this will create an arrangement of probabilities for each prediction that the ANN makes. The 3 hidden layers have 128 neurons each, and the reason why there are 101 neurons at the output is due that the problem will be solved by classification instead of regression, this means that each neuron will have a value assigned to it for each class and there are 101 classes. In Fig. 5 it can be observed that the outputs are values between 0 and 1 with increments of 0.01 between each output.

4. Prediction algorithm

4.1. Optimizer

Adam optimizer was chosen because of its benefits and it suits perfectly to our case due to is computationally efficient, has little memory requirements and is well suited for problems that are large in terms of data (Kingma and Ba, 2014), since the data set is a lot of information this requires a lot of memory.

4.2. Loss

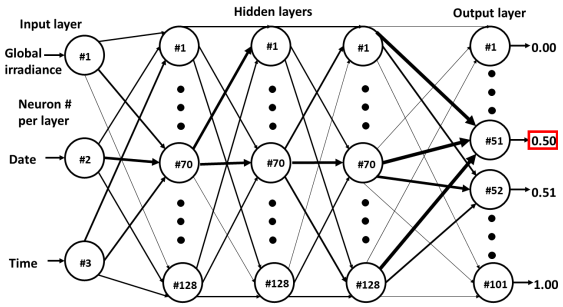
The loss sparse_categorical_crossentropy was used. And the values of the diffuse fraction array were transformed to integers to be able to use this loss and make multi-class predictions with classification. The outputs will be in softmax activation function that was assigned to the output layer.

4.3. Metrics

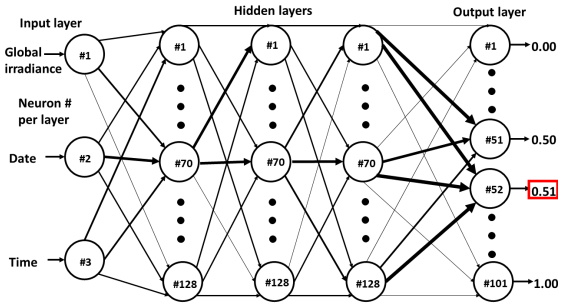
Performance evaluation is undoubtedly one of the most important points in a forecast. Therefore, several metrics will be compared. An explanation will be given of the accuracy achieved by the ANN given the good predictions it made for the diffuse fraction value. For the verification of the forecast many factors must be taken into consideration (Lorenz et al., 2009). From the scientific point of view, verification allows us to understand and improve the models, and depending on the application, the interpretation given to those results may vary (Jolliffe and Stephenson, 2012). As mentioned by Hossin and Sulaiman (2015), there are various types of evaluation metrics that can be used to evaluate the quality of classifiers with different aims.

And here is an important point to mention, despite solving the problem with multi-class classification, conventional metrics for this type of solution do not help to understand the true performance of this ANN. For example, the accuracy metric is one of the most used to evaluate the generalization capability of classifiers, however, Hossin and Sulaiman (2015) mentioned "accuracy has several weaknesses which are less distinctiveness, less discriminability, less informativeness and bias to majority class data." And this paper will prove that this is true. To this end, accuracy was chosen to evaluate the ANN and after comparison between the accuracy that it has with the validation and the test data, some regression error metrics will be calculated to evaluate the real performance since the objective of the network is to predict the closest to real value of the diffuse fraction and not whether the class chosen was the right one.

To clarify this, Fig. 6 shows a hypothetical case of 2 predictions of the diffuse fraction where the thickness of the lines represents the weights assigned to those connections. Suppose



(a) Prediction in state "A"



(b) Prediction in state "B"

Figure 6: Hypothetical case of 2 predictions of the diffuse fraction value with the ANN

that the real value of the diffuse fraction is 0.50 for state "A" defined by time, date and irradiance. And likewise, a value of 0.50 for the state "B" defined by another time, date and irradiance.

Fig. 6a shows that the connections with the highest weights are those of neuron number 51 and when making the calculations mentioned in Fig. 1b the class selected is 0.50. In this case the class prediction would be correct for state "A", however, in state "B", for Fig. 6b the neuron with the connections that have the highest weights is neuron number 52, and when making the calculations again from Fig. 1b the class selected is 0.51. In this case the prediction of the class was incorrect, and the accuracy of the ANN would be bad, since in state "B" it did not choose the correct class, which in this example would be neuron number 51. However, this does not mean that it was a bad prediction, it only means that it chose the incorrect class, but the predicted value of 0.51 is a value close to 0.50, so the accuracy does not say much about whether the prediction of the real values was good, it only indicates whether it chose the classes well in each prediction. Similarly, Hossin and Sulaiman (2015) said that "the trained classifier is measured based on total correctness which refers to the total of instances that are correctly predicted by the trained classifier when tested with the unseen data."

4.4. Training parameters

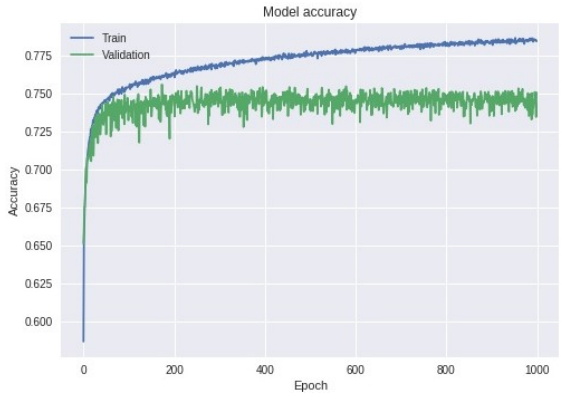
To train the ANN, the hyper-parameters were changed a few times, finally a batch_size of 256 was selected with 1000 epochs. This took about 45 minutes of training using the GPU from Google Colaboratory environment.

5. Results and Discussion

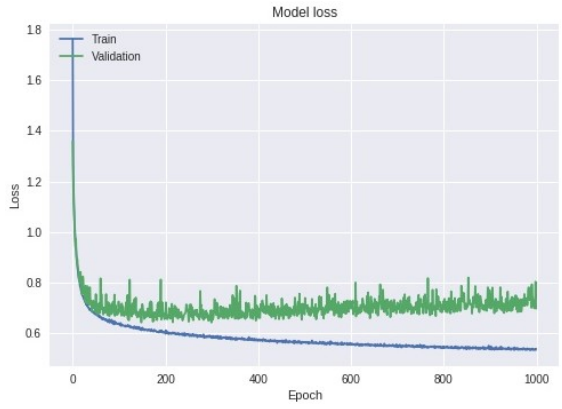
The accuracy value that the ANN had was 78.41% during the training, and it could be thought that it is a not so accurate result, however, as each class has 0.01 increase from the previous one it turns out that very close results are obtained.

The analysis for the accuracy values with the training and validation data is shown in Fig. 7a. It is evident that the value increases as the ANN learns with each epoch it performs. Each iteration the ANN trains with an amount of data equal to the input matrix divided by the batch_size. This is 131400 elements (The 15 years that were separated in a previous section) divided by 256 which was the number taken as the value of batch_size. Different configurations could be tried in future works, however, in this report it was considered that with the proposed architecture and the proposed hyper-parameters good results were obtained given the short time of training.

Xiao and Si (2017) states that when errors accumulate during training there is a high impact on neural networks based on iterate-based approaches. And computational costs are factors to consider.



(a) Model accuracy



(b) Model loss

Figure 7: Analysis and comparison between training and validation data metrics.

As shown in Fig. 7a there is an increase in the accuracy of the model with each iteration, however it is considered that 45 minutes is an excellent time to have trained with 15 years. And given the results, it can be seen that this model does not require high computational costs nor do the errors have much impact.

Table 1: Model accuracy percentages

Validation data	Test data
75.03995299339294%	74.77169036865234%

If any author, researcher or organization would like to use this network more accurately an option to improve it could be to implement techniques such as dropout, changing hyperparameters or the architecture of the network to reduce even more the number of errors.

Fig. 7b shows the analysis for the loss and validation loss, as it can be seen, the ANN is reducing the value as the network is training and it is just what we are looking for.

Accuracy was also evaluated with validation data and test data. Table 1 shows the results that were obtained with the model predictions.

The result of the test data is lower than the validation data since the validation data was seen by the ANN during the training, therefore, a lower percentage is achieved in data that the network has not seen.

To better appreciate the results, the first week of predictions was plotted along with the actual value of the diffuse fraction. Comparisons for the predictions with the validation data are shown in Fig. 8a and comparisons with the test data are shown in Fig. 8b.

As shown in Fig. 8a and Fig. 8b there is a very close prediction even when it fails. That is, if the actual value does not match, the predicted value is mostly of some class nearby, i.e. the output that had high weight values in its connections is a neuron of a class nearby.

First, only 11 outputs had been established, which were equally in the range of 0 to 1 but with 0.1 increments between each output neuron. This resulted in much greater accuracy in choosing the right class. However, having a low decimal accuracy resulted in more distant predictions of the actual value. For that reason, it was decided to take 101 classes and in that way 101 outputs could be used with 0.01 increments as mentioned in a previous section.

5.1. Statistical indicators

To prove this, what was done was to evaluate the results mathematically. The results were exported to a CSV file and several errors were calculated such as Mean Absolute Error (MAE), Mean Bias Error (MBE), Mean Square Error (MSE), Root Mean Square Error ($RMSE$), Normalized Root Mean Square Error ($NRMSE$) and Coefficient of Determination (R^2). All of this was calculated in that csv file and saved it in xlsx file. The errors were calculated using the following equations proposed by Pal (2016, pp. 83-88) in his book:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{2}$$

$$MBE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i) \tag{3}$$

Table 2: Error values with the test data

MAE	MBE	MSE	$RMSE$	$NRMSE$
0.006170463	-0.000098717	0.001223792	0.034982738	0.088337497

Table 3: Coefficient of determination and correlation coefficient

R^2	R
0.992196487	0.99609536

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5}$$

$$NRMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{6}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{7}$$

Where n is the number of data points in the data set and the data set being evaluated is the test data, in this case test data has 26280 elements as mentioned in Subsection 2.2.3, y_i is the ith real (observed) diffuse fraction value, $\hat{y}_i$ is the ith predicted diffuse fraction value and $\bar{y}$ is the mean of the real (observed) diffuse fraction value.

And correlation coefficient R that measures the linearity between the model and the experiment data is stated by (Posadillo and Luque, 2010) as follows:

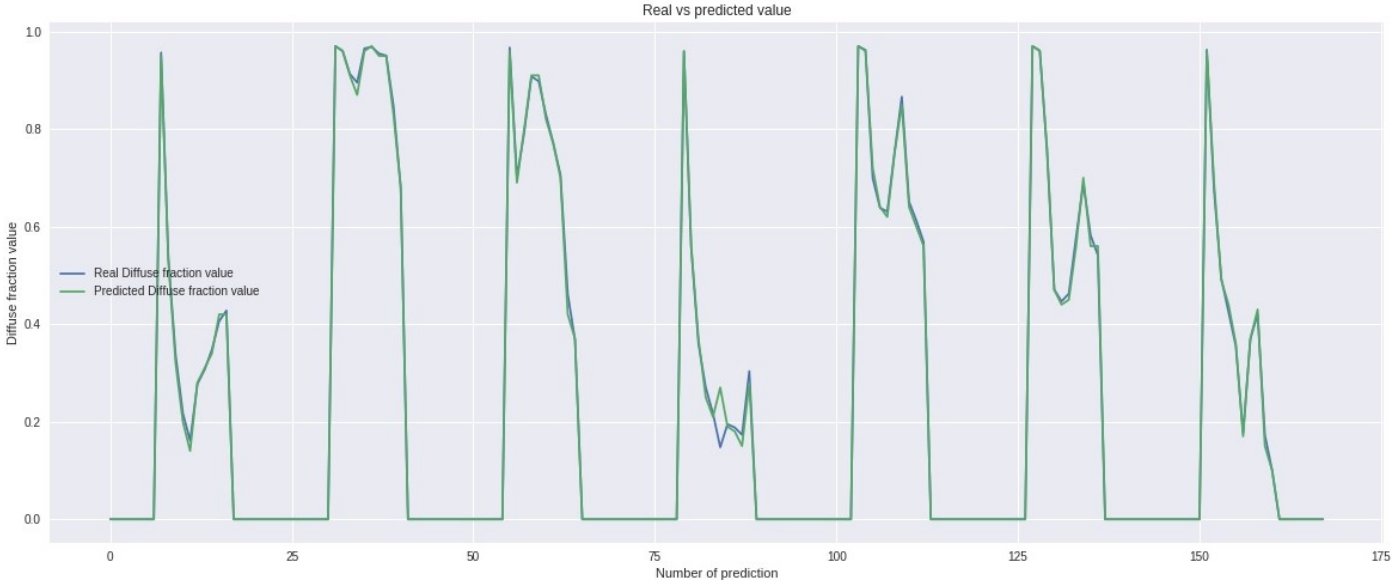
$$R = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{p})(y_i - \bar{y})}{\{[\sum_{i=1}^{n}(\hat{y}_i - \bar{p})^2][\sum_{i=1}^{n}(y_i - \bar{y})^2]\}^{\frac{1}{2}}} \tag{8}$$

Where $\bar{p}$ is the mean of the predicted diffuse fraction value. The other letters kept their meanings defined before Eq. 8. For original nomenclature of this formula (see Posadillo and Luque, 2010, Section 2.).

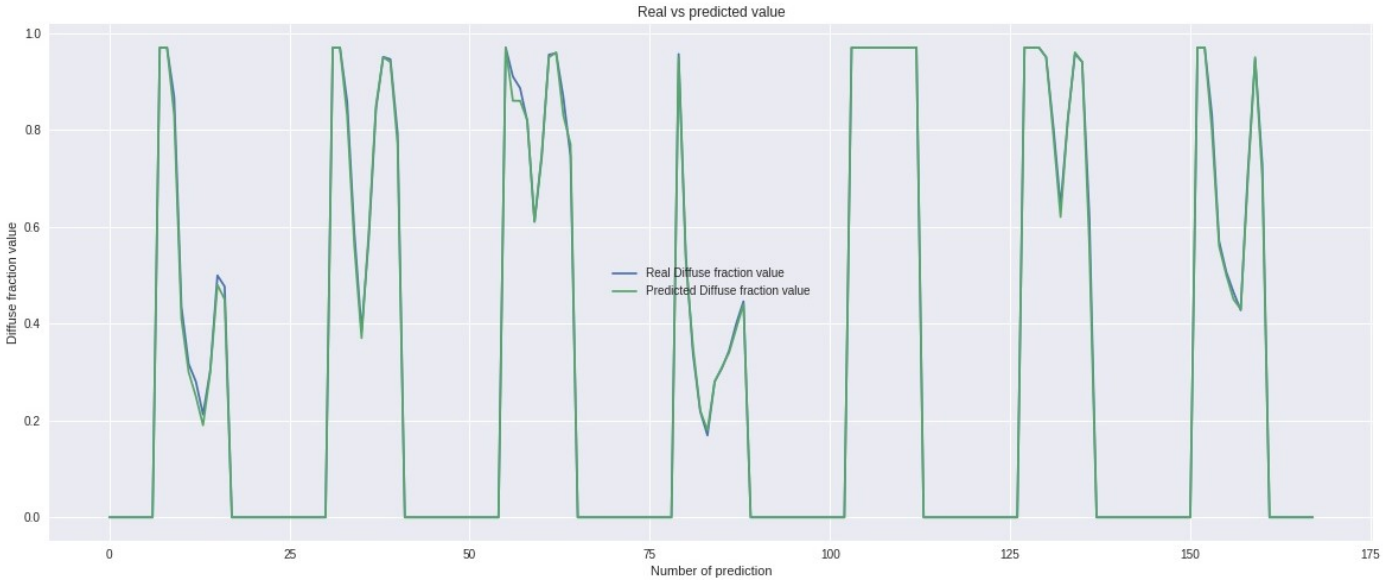
The results for the Eqs. (2)-(6) are shown in Table 2. And results for Eqs. (7) and (8) in Table 3.

Some authors prefer to express their results as percentages, but not all of them display the formula with which they calculated their statistical indicators, and there is a great deal of ambiguity in the literature, for example Tapakis et al. (2016a,b) expresses their results by using the statistical indicators MBE, $RMSE$ as percentages and R^2 from the paper of Posadillo and Luque (2010) which defines only Relative Mean Bias Error ($RMBE$), Relative Root Mean Square Error ($RRMSE$) and correlation coefficient R through formulas, in addition the formulas defined by David et al. (2012) differ in that $RMBE$ and $RRMSE$ are not multiplied by a factor of 100.

The formulas for $RMBE$ and $RRMSE$ by Posadillo and Luque (2010) are included below. And to avoid confusion, the same nomenclature as in the formulas defined in this paper will be used, and in addition $RMBE$ and $RRMSE$ will be written as $rMBE$ and $rRMSE$ respectively to avoid confusion with the



(a) Comparison between validation data and real value.



(b) Comparison between test data and real value.

Figure 8: Plot of predictions and real values of the diffuse fraction.

capital letter R of the Root Mean Square Error ($RMSE$) nomenclature. For original nomenclature (see Posadillo and Luque, 2010, Section 2.). The results of Eqs. (9) and (10) are shown in Table 4.

$$rMBE = \frac{[\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)]}{\bar{y}_i}(100) \tag{9}$$

$$rRMSE = \frac{[\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2]^{\frac{1}{2}}}{\bar{y}_i}(100) \tag{10}$$

5.2. Comparison between other models

Now to compare with other works and demonstrate the capability that the multi-class ANN classifier has. Real (observed) k_d vs predicted (forecast) k_d were plotted in Fig. 9.

Table 4: Error metrics percentages

$rMBE$	$rRMSE$
-0.027292270%	9.67166531%

Elminir et al. (2007) compare their ANN with the analytical model of Erbs et al. (1982), they determined coefficient of determination R^2 in a range of 0.95-0.89 and the average of their correlation coefficient R was 0.96, results which are slightly less than the obtained values of this paper shown in Table 3. Moreover their predicted hourly k_d values are more scattered (see Fig. 2 Elminir et al., 2007, Section 4) than Fig. 9. Ihya et al. (2015) compare their work equally with Erbs et al. (1982), they

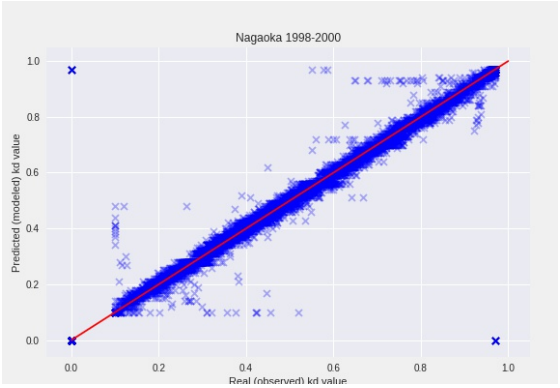


Figure 9: Scatter plot of the Real (observed) diffuse fraction vs predicted (forecast) diffuse fraction.

present hourly and daily scales. For the hourly scale the correlation coefficient R was 0.94-0.97 and additionally $rRMSE$ about 20% and for the daily scale the correlation coefficient R was 0.9596 and $rRMSE$ equal to 18.73% for the best model. Once again, scatter plot is presented in (see Fig. 5 and Fig. 8 Ihya et al., 2015, Section 6) one for the hourly scale and one for daily scale. However better results are achieved with the proposed ANN of this paper as it is shown in Table 3, Table 4 and Fig. 9.

According to the above arguments, formulas and results this model proved multi-class classification could have better predictions than non-linear regression model.

5.3. Further analysis with different separation of the data

Finally, a different configuration was carried out in the separation of the data, since usually 20 years of information is not available. For this reason, the last 5 years (1996-2000) were separated from the 20 years of the data set. The same ANN described above was used with the only difference that now the data entry was 3 years (1996-1998) for training, 1 year for validation (1999) and 1 year for testing (2000). For comparative purposes the same calculations of Eqs. (2)-(10) were made for the output file of this new ANN training. The results are described in Table 5.

Table 5: Results of Eqs. (2)-(10)

MAE	0.006501448
MBE	-0.000340721
MSE	0.001052089
$RMSE$	0.032435927
$NRMSE$	0.080849531
R^2	0.993463353
R	0.99673367
$rMBE$	-0.093127311%
$rRMSE$	8.86552495%

As can be seen in Table 5, the results are close to what was obtained in the analysis with the 20 years. This shows that even if there is not a large amount of data, good predictions are achieved using this ANN.

6. Conclusion

In this paper, a Feedforward Neural Network is proposed. This model estimates the diffuse fraction value over the time. The advantages and results of this model are summarized below.

- This paper proves that non-linear regression model is not the only way to forecast hourly diffuse fraction and that the forecasts could be more accurate in some cases. Henceforth, further research into multi-class classification solution should be done.
- One of the advantages is that the ANN learns from the data, so it does not require any analytical model involving more input parameters that could involve more economic expenditure in obtaining the data.
- Although the analysis was carried out for the city of Nagaoka, this model is not site-specific. Ideally, the network should be trained with past data from the forecast location. If no records exist for that location, other models that predict hourly solar irradiance such as GHI or GTI can be used.
- Even with a smaller amount of data for training it can work well.
- This work could be further improved by changing the architecture, the type of ANN, modifying the hyperparameters or separating the data differently.

Acknowledgement

The author wishes to express his gratitude to Dr. YAMADA Noboru, Manager of the Energy Engineering Laboratory, for his support and encouragement, to B.S. in M.E. YAMAGATA Yuki, to B.S. in M.E. OTAKI Daiki for their explanations on Artificial Neural Networks for a classification problem and their constant support, and finally, to University of Guanajuato and Nagaoka University of Technology for giving to the author the wonderful opportunity to research in Energy Engineering Laboratory, Niigata, Japan.

References

Chakraborty, K., Mehrotra, K., Mohan, C.K., Ranka, S., 1992. Forecasting the behavior of multivariate time series using neural networks. Neural networks 5, 961–970.

Chandra, R., 2018. Multi-task modular backpropagation for dynamic time series prediction, in: 2018 International Joint Conference on Neural Networks (IJCNN), IEEE. pp. 1–7.

David, M., Diagne, M., Lauret, P., 2012. Outputs and error indicators for solar forecasting models, in: Proceedings of the World Renewable Energy Forum (WREF), pp. 13–17.

Elminir, H.K., Azzam, Y.A., Younes, F.I., 2007. Prediction of hourly and daily diffuse fraction using neural network, as compared to linear regression models. Energy 32, 1513–1523.

Erbs, D., Klein, S., Duffie, J., 1982. Estimation of the diffuse radiation fraction for hourly, daily and monthly-average global radiation. Solar energy 28, 293–302.

Fox, S.I., 2014. Fisiología humana. (13a. ed.) ed., McGraw-Hill Interamericana. URL: https://ebookcentral.proquest.com/lib/ugtomhe/detail.action?docID=3221049.

Hossin, M., Sulaiman, M., 2015. A review on evaluation metrics for data classification evaluations. International Journal of Data Mining & Knowledge Management Process 5, 1.

Ihya, B., Mechaqrane, A., Tadili, R., Bargach, M., 2015. Prediction of hourly and daily diffuse solar fraction in the city of fez (morocco). Theoretical and applied climatology 120, 737–749.

Jolliffe, I.T., Stephenson, D.B., 2012. Forecast verification: a practitioner's guide in atmospheric science. John Wiley & Sons.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

Lorenz, E., Hurka, J., Heinemann, D., Beyer, H.G., 2009. Irradiance forecasting for the power prediction of grid-connected photovoltaic systems. IEEE Journal of selected topics in applied earth observations and remote sensing 2, 2–10.

Meenal, R., Selvakumar, A.I., 2017. Review on artificial neural network based solar radiation prediction, in: 2017 2nd International Conference on Communication and Electronics Systems (ICCES), IEEE. pp. 302–305.

Pal, R., 2016. Predictive modeling of drug sensitivity. Academic Press.

Pełech-Pilichowski, T., 2018. On adaptive prediction of nonstationary and inconsistent large time series data, in: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), IEEE. pp. 1260–1265.

Pełech-Pilichowski, T., Duda, J., 2008. Adaptacyjne algorytmy detekcji zdarzeń w szeregach czasowych. WEAIiE AGH, Kraków (rozprawa doktorska) .

Posadillo, R., Luque, R.L., 2010. The generation of hourly diffuse irradiation: A model from the analysis of the fluctuation of global irradiance series. Energy Conversion and Management 51, 627 – 635. URL: http://www.sciencedirect.com/science/article/pii/S0196890409003197, doi:https://doi.org/10.1016/j.enconman.2009.08.034.

Tapakis, R., Michaelides, S., Charalambides, A.G., 2016a. Computations of diffuse fraction of global irradiance: Part 1–analytical modelling. Solar Energy 139, 711–722.

Tapakis, R., Michaelides, S., Charalambides, A.G., 2016b. Computations of diffuse fraction of global irradiance: Part 2–neural networks. Solar Energy 139, 723–732.

Xiao, Q., Si, Y., 2017. Time series prediction using graph model, in: 2017 3rd IEEE International Conference on Computer and Communications (ICCC), IEEE. pp. 1358–1361.

Yang, D., Yagli, G.M., Quan, H., 2018. Quality control for solar irradiance data, in: 2018 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia), IEEE. pp. 208–213.

Zhou, Y., Liu, Y., Chen, Y., Wang, D., 2019. General models for estimating daily diffuse solar radiation in china: Diffuse fraction and diffuse coefficient models. Energy Procedia 158, 351–356.