



**Research Article**

## Solar Energy Prediction Using Particle Swarm Optimiser Trained Recurrent Neural Network

\*Ibrahim Makama Dattijo and Salihu Aish Abdulkarim  
Department of Computer Science, Federal University Dutse, Nigeria

\*Corresponding Author: [idattijo@gmail.com](mailto:idattijo@gmail.com)

**ARTICLE INFO:**

**Keyword:**

Artificial Neural Networks,  
Backpropagation,  
Forecasting,  
Particle swarm optimisation,  
Recurrent Neural Network,  
Time series

**ABSTRACT**

In recent years, renewable energies have been covering an increasing part of the worldwide electrical power demand. The amount of solar radiation received by the earth's surface is very important to numerous renewable energy applications. However, direct measurement of solar data is not widely available, especially in developing countries such as Nigeria. Numerous statistical methods have been developed and used in the areas of solar energy prediction. Statistical approaches such as; Auto Regressive Moving Average (ARIMA), and linear regression (LR) are used. The challenges of using these statistical approaches are Linear methods (works best when the data is linear), high mathematical complexity and dependence on prior knowledge of how the time series was generated. In the meantime, data and computational power availability have enabled machine learning algorithms to perform improved predictions. The need to predict solar energy output is essential; machine learning models can be employed. In this work, Particle Swarm Optimization (PSO) and Back propagation (BP) algorithms were used to train a Recurrent Neural Network (RNN) to model monthly mean daily solar radiation values. The inputs to the network are month, sunshine duration, latitude, longitude, and altitude of the location. The BP algorithm is compared with the PSO algorithm trained by the RNN. The PSO outperformed the BP in most cases with average accuracy for the BP at about 0.71, 0.725, 0.76, 0.735 and 0.78, 0.8, 0.84, 0.815 for the PSO algorithm for the four (4) datasets used.

**Corresponding author:** Ibrahim Makama Dattijo, Email: [idattijo@gmail.com](mailto:idattijo@gmail.com)  
Department of Computer Science, Federal University Dutse, Nigeria.

## INTRODUCTION

Solar energy is the portion of the sun's energy available at the earth's surface for useful applications, including raising the temperature of the water or exciting electrons in a photovoltaic cell, supplying energy to natural processes (Xue, 2017). The amount of solar radiation received by the earth's surface is very important to numerous renewable energy applications. However, direct measurement of solar data is not widely available, especially in developing countries such as Nigeria (REMP, 2007). At the moment, only 44 weather stations in Nigeria measure climatic parameters like sunshine hours, temperature, rainfall, atmospheric pressure, and humidity (Osinowo *et al.*, 2015). However, out of these 44 weather stations, only 16 of them measure solar radiation, leaving most of the locations in the country with no solar radiation. This dependency on weather adds uncertainty and variability to solar power generation. To deal with this uncertainty, solar forecasting is employed to predict solar radiation.

Forecasting is a process of determining what will happen in the future by analysing what happened in the past and what is going on now. Forecasting is an important problem that spans many fields, including business and industry, government, economics, environmental sciences, energy, medicine, social science, politics, and finance (Armstrong, 2001). Most forecasting problems involve the use of time-series data. Time series is a time-oriented or chronological sequence of observations on a variable of interest (Adhikari and Agrawal, 2012).

Over the past couple of years, solar energy has become one of the most popular renewable energy sources (RES). Unfortunately, solar energy generation depends completely on sun radiation. Using the time-series approach for forecasting, forecasters collect and analyse historical

observations to determine a model to capture the underlying data generating process. Then the model is extrapolated to forecast future values. This approach is useful for applications in many domains such as business, economics, industry, engineering, and science. Over the past three decades, much effort has been devoted to developing and improving time series forecasting models (Abdulkarim and Garko, 2015). Due to the importance of time series forecasting, much research effort has gone into developing forecasting models and improving prediction accuracy. Statistical methods were the largely used time series forecasting tools (Chatfield, 2003). However, statistical methods are known to have two main drawbacks, namely, high mathematical complexity and dependence on prior knowledge of how the time series was generated (Douglas *et al.*, 2015).

Artificial Neural Network (ANN) is one of the most widely used data-driven modelling techniques with lots of good features like easy applications, high accuracy in prediction, and predicting the multi-output complex non-linear relationships (Raza *et al.*, 2014). Neural networks (NN), as an alternative approach to time series modelling, address these issues and have shown to be more efficient in terms of prediction accuracy (Hinton, 2006). The interest in using NNs to model and forecast time series has been growing. The most popular type of NN is arguably the feedforward NN (FNN). FNNs have structures capable of learning input-output mappings suitable for predicting non-linear time series (Abdulkarim and Garko, 2015). NN as an artificial intelligence approach has been recently introduced and applied in solar radiation estimations. This method can solve complex nonlinear problems which cannot be solved easily. It is possible to train the NNs via different

algorithms such as Gradient Descent (GD) and backpropagation (Ram *et al.*, 2012). Although NNs have done well in many time series forecasting problems, their performance depends on selecting the right network architecture and an appropriate training algorithm (Andrade and Bessa, 2017).

In (Dorffner, 1996), provides an outline of various forecasting techniques used for energy forecasting. The study emphasises that many different time series and machine learning (ML) techniques have been used in various energy contexts. The study is based on a competition for energy forecasting and evaluates the methods used in the Global Energy Forecasting Competition (Kaggle, 2014).

In recent work (Andrade and Bessa, 2017), they explored ML techniques and evaluated them in combination with developing features that supposedly should improve the performance. The main approaches used in the study were principal component analysis (PCA) and a feature engineering methodology in combination with a gradient boosting tree (GBT) model. The authors used different smoothing approaches to create features from their numerical weather prediction (NWP) data. Specifically, the authors used a grid of NWP data around the location of the photovoltage (PV) installation and computed spatial averages and variances of weather parameters. Besides creating features based on a local grid of points, they also computed variances for different predictors for different lead times. When constructing variance features based on lead times, the underlying idea was that the feature would indicate the variability of the weather. However, these methods are inefficient because the model may perform better when trained on one weather season or location.

A model used by (Persson *et al.*, 2017) employs gradient boosted regression trees (GBRT) to forecast solar energy generation 1-6 hours ahead. The data used was historical power output and meteorological features for 42 PV installations in Japan. The GBRT model performed better than the adaptive recursive linear autoregressive (AR) time series model, persistence model and climatology model on all forecast horizons. For shorter forecast horizons, it was shown that lagged power output values had a larger predictive ability. However, for longer forecast horizons, the weather forecasts increased in importance, thereby the model performed very poor.

In their work (Mohammadi *et al.*, 2015), they proposed a new hybrid approach by combining the support vector machine (SVM) with a wavelet transform (WT) algorithm to predict horizontal global solar radiation. They concluded that the proposed SVM-WT model shows higher performance than other well-known techniques such as; genetic programming (GP), autoregressive (AR) and autoregressive moving average (ARMA) models. They adopted an adaptive neuro-fuzzy inference system (ANFIS) model to estimate the daily horizontal global solar radiation. They concluded that the proposed ANFIS model could be utilised to estimate the horizontal global solar radiation with a favourable level of reliability and precision. However, the result shows a way of using SVM-WT models to train models on specific climatic conditions without using machine learning.

A similar model used by (Olatomiwa *et al.*, 2015), investigated the potential of support vector regression (SVR) for solar radiation prediction in Nigeria. They concluded that SVR with polynomial basis outperforms SVR with radial basis function and other methodologies in

solar radiation estimation of the studied site. They proposed a hybrid machine learning approach by combining a support vector machine (SVM) with a firefly algorithm (FFA) to predict the monthly mean horizontal global solar radiation. They concluded that the developed SVM-FFA model could be adjudged as an efficient machine learning technique for accurately predicting horizontal global solar radiation. However, the model performs well on linear data, not a multivariate dataset.

In their work (Mendes *et al.*, 2002) proposed the application of particle swarms to the training of neural networks. Three different models for training feed-forward neural networks were considered. It includes particle swarm model, gradient-based model and evolutionary programming models. Classification tasks and regression analysis is used to compare the different models. Results point out that, while BP was robust, PSO showed to be valuable in several cases. However, it has slow convergence and over-fitting problems.

This paper validates the performance of the particle swarm optimisation algorithm over the backpropagation algorithm for the prediction of solar radiation.

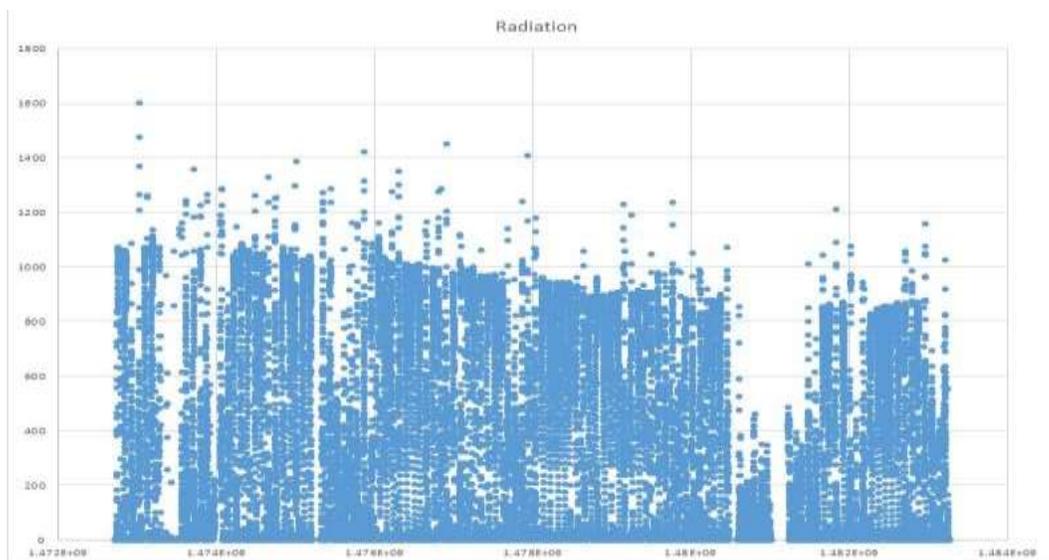
**MATERIALS AND METHODS**

**Dataset**

A set of four (4) different datasets, namely Nigerian Meteorological (NAMET) Agency (Nigeria), National Renewable Energy Lab (NREL) China, Bangladesh and Global Heat Irradiance (GHI). The four datasets are real-world time series obtained online from the data world website (Data World, 2018).

**Table 1:** Datasets

S/No.	Datasets
1.	NIMET
2.	NREL (China)
3.	NREL (Bangladesh)
4.	GHI



**Figure 1:** Diffuse Solar radiation

Figure 1 shows the pictorial visualization of the Nigerian Meteorological (NAMET) Agency – Solar Dataset. The meteorological data used in this study are the recorded solar radiation values during the period extending from September to December 2016 from a meteorological station in Nigeria, as shown in Figure 1. Daily meteorological variables, including the month of the year, sunshine duration (h), mean temperature (0C), rainfall (mm), wind speed (m/s), relative humidity (%), daily global solar radiation  $H_g$  ( $MJ\ m^{-2}\ d^{-1}$ ), and daily diffuse solar radiation  $H_d$  ( $MJ\ m^{-2}\ d^{-1}$ ). The datasets were rigorously quality controlled by NAMET.

**Artificial Neural Networks**

Artificial neural networks (ANNs) are flexible modelling tools with the capabilities of learning the mathematical mapping between input and output variables of nonlinear systems (Abdulkarim, 2016).

The basic objective of ANNs was to construct a model for mimicking the intelligence of the human brain into a machine (Naylor *et al.*, 1972). Similar to the work of a human brain, ANNs try to recognise regularities and patterns in the input data, learn from experience and then provide generalised results based on their known previous knowledge. Although the development of ANNs was mainly biologically motivated, afterwards, they have been applied in many different areas, especially for forecasting and classification purposes (Han *et al.*, 2018).

**Back propagation**

Backpropagation is a widely used algorithm in training feed-forward neural networks for supervised learning. Generalizations of backpropagation exist for other artificial neural networks (ANNs), and for functions generally, a category of algorithms is mentioned generically as "backpropagation". BP computes the gradient of the loss function for the weights of the network for one input-output.



**Figure 2:** Backpropagation flow chart (Mohandes, 2012)

## Particle Swarm Optimization

PSO is a Meta-heuristic algorithm originally proposed by Kennedy and Eberhart (Kennedy and Eberhart, 1995a). The algorithm simulates the behaviour of birds' flocks flying together in multi-dimensional space to search for some optimum place, adjusting their movements and distances for better search. PSO is an evolutionary computation method similar to the Genetic Algorithm (GA). Swarms called particles are initialised randomly and then searched for optima by updating generations. The algorithm mimics a particle flying in the search space and moving towards the global optimum. A particle in PSO can be defined as:

$$\begin{aligned} \text{Vid}(it + 1) = & \text{Vid}(it) + C1 * \text{Rnd}(0,1) * \\ & (\text{pbid}(it) - \text{Xid}(it)) + \text{Rnd}(0,1) * \\ & (\text{gbid}(it) - \text{Xid}(it)) \end{aligned} \quad (1)$$

$$\text{Xid}(it + 1) = \text{Xid}(it) + \text{Vid}(it + 1) \quad (2)$$

where,  $i$ -particle's index,  $d$ -Particle's dimension,  $it$ -Iteration number,  $\text{Xid}$ -Position of particle  $i$  in dimension  $d$ ,  $\text{Vid}$ -Velocity of particle  $i$  in dimension  $nd$ ,  $C1$ -acceleration constant for the cognitive component,  $\text{Rnd}$  - Stochastic component of the algorithm (a random value between 0 and 1).

## Parameter Selection

To ensure that all the algorithms exhibited efficient performance and to facilitate fair comparison among the training algorithms, the relevant algorithm parameters were optimised and set up as follows:

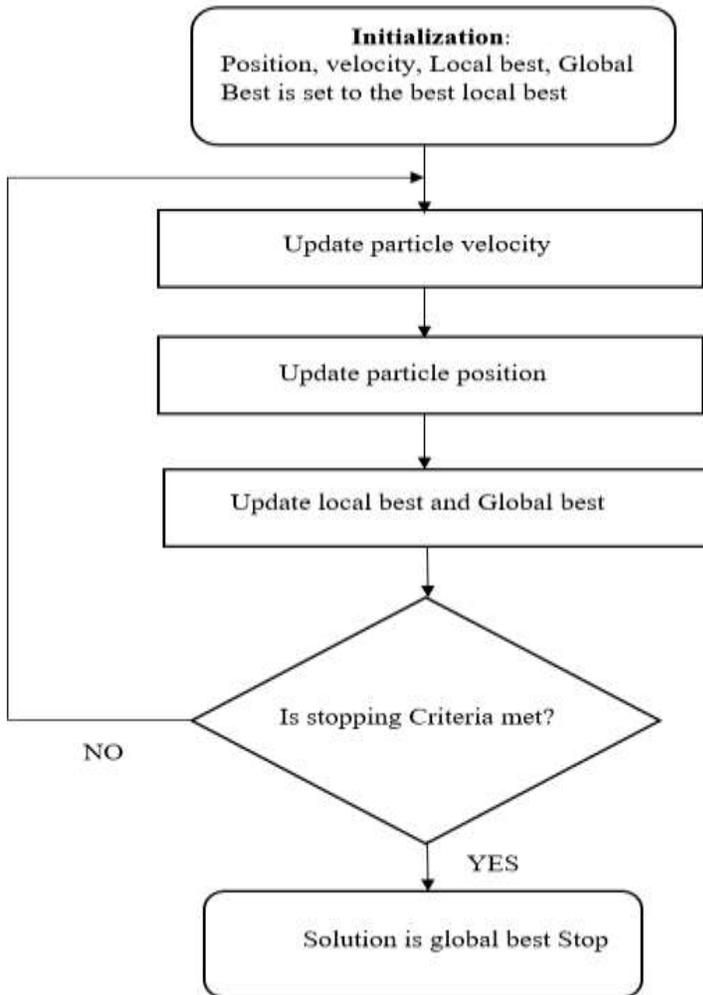
### *NN configuration*

The NN for each dataset was determined as follows:

- **Input layer:** For datasets collected annually, 10 input nodes were used, each representing a year in the decade. For monthly datasets, 12 input nodes were used, each representing a month of the year. For quarterly/weekly datasets, four input nodes were used, each representing a quarter of the year or a week of the month respectively. For data collected hourly, 24 input nodes were used.
- **Hidden layer:** A single hidden layer was used for all the NNs and the number of hidden nodes was iteratively optimised on the validation set, where discrete numbers in the range (Elman, 1990) were considered. For every value within the range, 30 independent runs were conducted, and the value that yielded the minimum average validation error was chosen as optimal.
- **Output layer:** A single output node with a modified hyperbolic tangent function was used for all the NNs (one step ahead forecasting was considered).

### *PSO Setup*

For all the experiments, a linearly decreasing inertia weight was used, with an initial value of 0.9 and an initial value of 0.5. Acceleration coefficients values were fixed at 'c1': 0.5, 'c2': 0.3, 'w': 0.9 to ensure convergence, based on (Eberhart and Shi, 2000).



**Figure 3:** PSO flow chart (Mohandes, 2012)

In figure 4: The basic structure of an ANN consists of many interconnected computing processors, called neurons or nodes, grouped into input, hidden, and output layers. Each node in an ANN, takes values from its inputs, multiply them by the corresponding weights and sums up all the results plus a constant bias value. The summation then passes a transfer function and produces the output of the node.

**Recurrent Neural network**

Recurrent Neural Networks (RNNs) are bidirectional networks specially designed for learning sequential or time-varying patterns (Grossberg, 2013). RNNs have feedback connections (which serves as internal memories)

that allow the networks to incorporate a trace of what has been processed by the networks (i.e., experience). The reason for using this recurrent mechanism is that, theoretically, any past observation in a time series can influence the predicted future values of the series. Also, the memory capacity of networks not employing the recurrent mechanisms is limited by the size of the window or the number of delay connections.

The class of RNN models that are updated with a single update cycle (as in the FNNs) are called simple recurrent NNs (SRNNs). Among the different types of SRNNs developed, the Elman NN (Elman, 1990) and the Jordan NN (Jordan, 1986) are the most common types applied in time series processing. While the Elman NN extends

an FNN by adding feedback connections from the hidden layer to a context layer, as shown in Figure 2.5, the Jordan NN have its feedback connections from the output layer to a state layer,

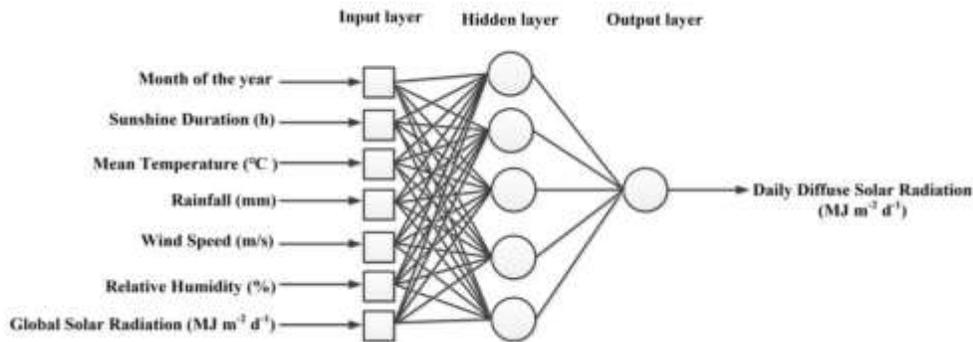
as illustrated in Figure 2.6. The context /state layers serve as extensions of the input layer and are used to keep the previous state of the hidden/output layers, respectively.

**Performance Evaluation**

Model evaluation metrics are required to quantify model performance. The choice of evaluation metrics depends on a given machine learning task (such as classification, regression, ranking, clustering, topic modelling, among others). Some metrics, such as precision-recall, are useful for multiple tasks. Supervised learning

tasks such as classification and regression constitute a majority of machine learning applications.

In this paper, accuracy is used as a metric for these supervised learning models. Accuracy is a common evaluation metric for classification problems. It's the number of correct predictions made as a ratio of all predictions made.



**Figure 4:** Structure of ANN

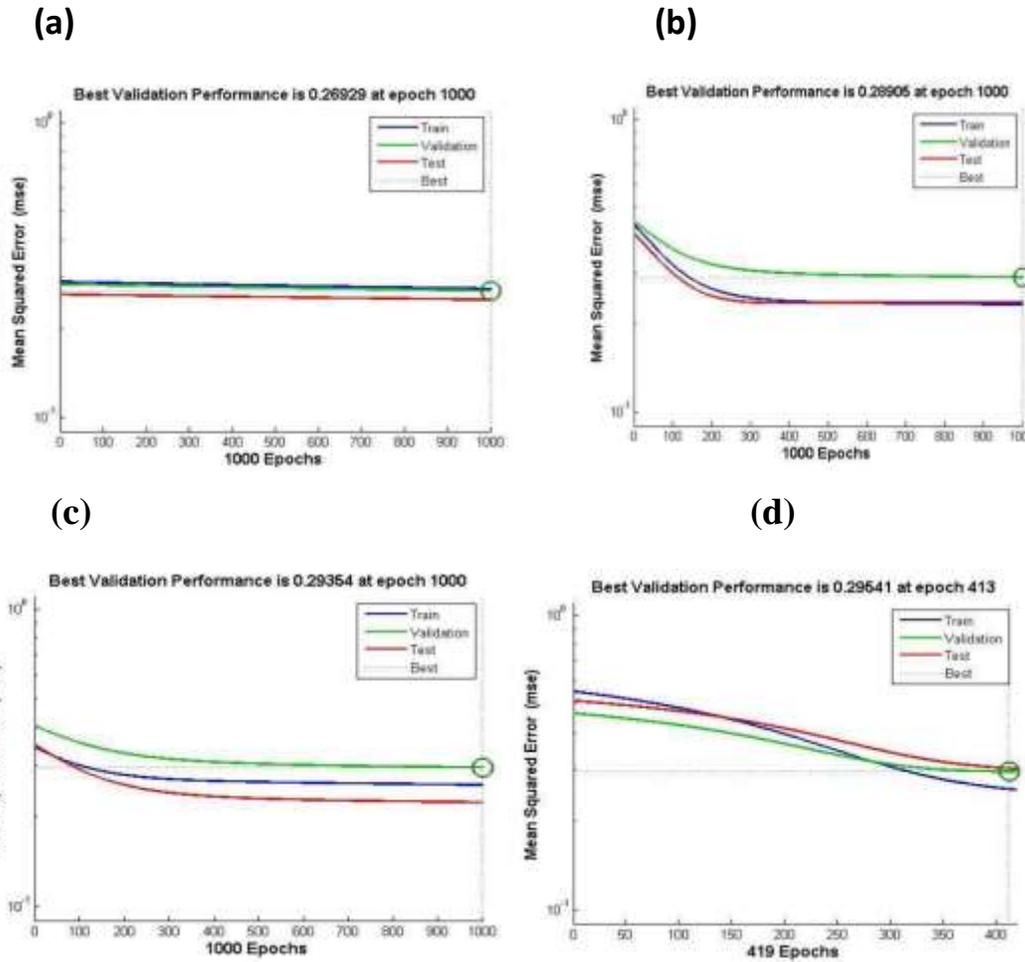
**RESULTS**

**Performance Evaluation**

This section evaluates the performance of PSO and BP algorithms. The experiment is conducted using Google Colab (Google, 2018). To investigate the performance of the training algorithms over time, both training and

generalisation errors were plotted against the iteration of the algorithms.

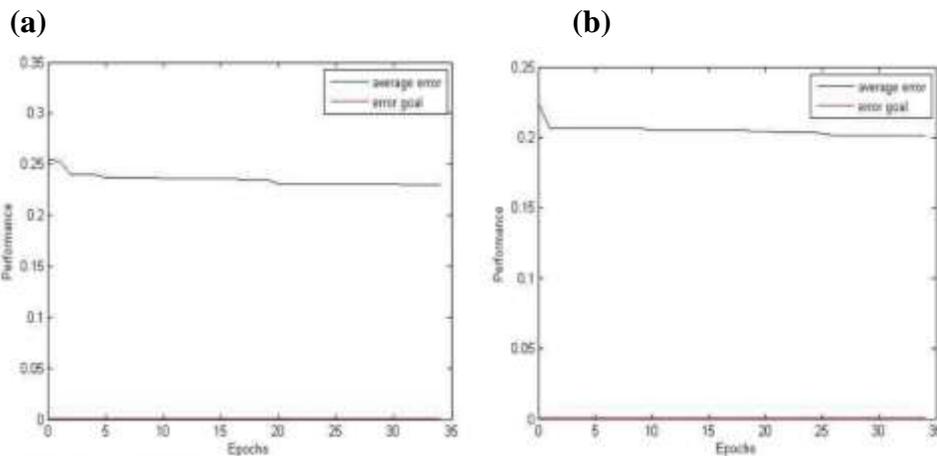
The mean squared error (MSE) of BP and PSO are represented respectively in Figure 5 and Figure 6. The performance plot is an indicator of the value of the performance function versus the iteration number.

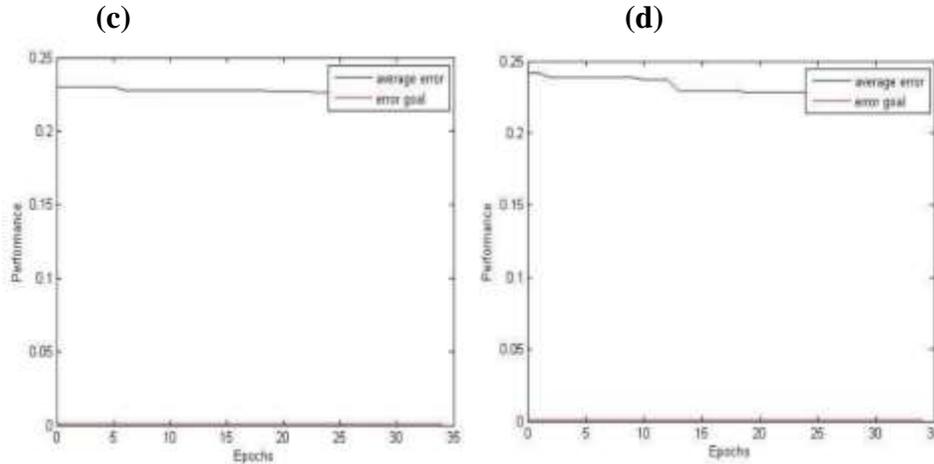


**Figure 5:** Convergence of Backpropagation (a) NIMET (Nigeria) (b) NREL (China) (c) NREL (Bangladesh) (d) GHI

Training, validation and test performances are plotted. The best validation performance reached

a minimum faster and more optimal in the case of PSO when compared with BP.





**Figure 6:** Convergence of particle swarm optimization (a) NIMET (Nigeria) (b) NREL (China) (c) NREL (Bangladesh) (d) GHI

Tables 2 and 3 show the corresponding values of the error convergence for both the algorithms across different data sets used for evaluation for a comparative study. Each

dataset has been populated from the main dataset with uniform distribution of both the classes to ensure proper evaluation.

**Table 2:** Error Convergence during training across different datasets

Datasets	Back Propagation		Particle Swarm Optimization	
	Epochs	Error	Epochs	Error
NIMET (Nigeria)	1000	0.26929	34	0.22805
NREL (China)	413	0.29541	34	0.22278
NREL (Bangladesh)	1000	0.28905	35	0.20107
GHI	1000	0.29354	34	0.23002

The results show that the number of epochs consumed and the error convergence value for the backpropagation algorithm is comparatively greater than particle swarm optimisation. This indicates that PSO converge faster. By faster convergence, training the neural networks becomes faster and the results produced are sufficiently accurate without much adaptation. Table 2 shows the corresponding values of the error convergence for both algorithms across

different datasets used for evaluation for a comparative study. Each dataset has been populated from the main dataset with uniform distribution of both the classes to ensure proper evaluation. The results show that the number of epochs consumed and the error convergence value for the backpropagation algorithm is comparatively greater than particle swarm optimisation. This indicates that PSO converges faster for BP neural networks. By faster convergence,

training the neural networks becomes faster and the results produced are sufficiently accurate without much adaptation. In addition to differentiating BP and PSO based on convergence, further analysis was performed to support the research. Sensitivity, Specificity, Positive precision value and accuracy was computed for the validation dataset used. The results have been summarised in Table 3. The sensitivity analysis will be able to quantify the capability

of the technique to identify a condition correctly, whereas specificity analysis will be able to quantify its capability to exclude a condition correctly. The positive predictive value is a statistical measure of the proportion of positive results. The accuracy of a technique refers to its classification performance. The results indicate that PSO not only converged but also was able to classify better for the given data set

**Table 3:** Convergence of backpropagation

Analysis	Back Propagation				Particle Swarm Optimization			
	NIMET	NREL (China)	NREL (Bangladesh)	GHI	NIMET	NREL (China)	NREL (Bangladesh)	GHI
<b>Sensitivity</b>	0.7849	0.7932	0.8154	0.7438	0.8181	0.8175	0.8429	0.818 1
<b>Specificity</b>	0.6448	0.633	0.6661	0.6967	0.7326	0.7249	0.728	0.745
<b>Positive Precision Value</b>	0.6576	0.649	0.6689	0.6514	0.75	0.7654	0.8076	0.782 8
<b>Accuracy</b>	0.71	0.725	0.76	0.735	0.78	0.80	0.84	0.815

The accuracy of a technique refers to its classification performance. The results indicate that PSO not only converges but also was able to classify better for the given datasets.

## CONCLUSION

This paper focuses on solar energy prediction. Among the several classification models, neural networks were considered for evaluation. PSO technique inspired by the social behaviour of biological swarms was considered for evaluation. PSO proved to be an optimal replacement for BP in predicting solar energy. The classification accuracy can be improved by consistent training. However, achieving further accuracy needs more

research. The research scope of PSO can be scaled across different domains. The proposed work that is considered just another application of PSO primarily focuses on identifying the significance of using PSO over BP. However, further research can be undertaken to research in-depth the impact of different variants of PSO by varying the neighbourhood topologies such as the pyramid and Von Neumann topology characterized based on the degree of

connectivity and the amount of clustering. In some cases, a hybrid of two algorithms performs better when compared with an algorithm implemented individually. Hence,

identifying whether a hybrid of different algorithms can be used in place of the existing techniques can be a possible line of research.

## REFERENCES

- Abdulkarim, S. A. (2016). Time series prediction with simple recurrent neural networks. *Bayero Journal of Pure and Applied Sciences*, 9(1), 19-24.
- Abdulkarim, S. A., & Garko, A. B. (2015). Forecasting maternal mortality rate using particle Swarm optimisation based artificial neural network. *Dutse J Pure Appl Sci*, 1(1), 55-59.
- Adhikari, R., & Agrawal, R. K. (2012). Forecasting strong seasonal time series with artificial neural networks.
- Andrade, J. R., & Bessa, R. J. (2017). Improving renewable energy forecasting with a grid of numerical weather predictions. *IEEE Transactions on Sustainable Energy*, 8(4), 1571-1580.
- Armstrong, J. S. (Ed.). (2001). *Principles of forecasting: a handbook for researchers and practitioners* (Vol. 30). Boston, MA: Kluwer Academic.
- Chatfield, C. (2003). *The analysis of time series: an introduction*. Chapman and hall/CRC.
- Data World (2018). Solar Monthly Data. Accessed: Jun. 21, 2018. [Online]. Available: <https://data.world/doi/solar-monthly-annual-dni>
- Dorffner, G. (1996). Neural Networks for Time Series Processing.
- Douglas C. Montgomery, C. L. J. and Kulahci, M. (2015), *Introduction to Time Series Analysis And Forecasting*, John Wiley Sons, Inc.
- Eberhart, R. C., & Shi, Y. (2000, July). Comparing inertia weights and constriction factors in particle swarm optimisation. In *Proceedings of the 2000 congress on evolutionary computation. CEC00 (Cat. No. 00TH8512)* (Vol. 1, pp. 84-88). IEEE.
- Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2), 179-211.
- Google (2018). Colaboratory: Frequently Asked Questions. Accessed: Jun. 21, 2018. [Online]. Available: <https://research.google.com/colaboratory/faq.html>
- Grossberg, S. (2013), 'Recurrent neural networks', *Scholarpedia* 8(2), 1888. revision #138057.
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786), 504-507.
- Han, H. G., Lu, W., Hou, Y., & Qiao, J. F. (2016). An adaptive-PSO-based self-organizing RBF neural network. *IEEE transactions on neural networks and learning systems*, 29(1), 104-117.
- Jordan, M. I. (1986), Attractor dynamics and parallelism in a connectionist sequential machine, in 'Proceedings of the Eighth Annual Conference of the Cognitive Science Society', Hillsdale, NJ: Erlbaum, pp. 531-546.
- Kaggle (2004). Global Energy Forecasting Competition. Accessed: Jun. 21,

2018. [Online]. Available: <https://kaggle.com/global-energy-competition>
- Kennedy, J. and Eberhart, R. (1995a), Particle swarm optimization, in 'Proceedings of ICNN'95 – International Conference on Neural Networks', Vol. 4, pp. 1942–1948 vol.4.
- Mendes, R., Cortez, P., Rocha, M., & Neves, J. (2002, May). Particle swarms for feedforward neural network training. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290)* (Vol. 2, pp. 1895-1899). IEEE.
- Mohandes, M. A. (2012). Modeling global solar radiation using Particle Swarm Optimization (PSO). *Solar Energy*, 86(11), 3137-3145.
- Mohammadi, K., Shamshirband, S., Anisi, M. H., Alam, K. A., & Petković, D. (2015). Support vector regression based prediction of global solar radiation on a horizontal surface. *Energy Conversion and Management*, 91, 433-441.
- Naylor, T. H., Seaks, T. G., & Wichern, D. W. (1972). Box-Jenkins methods: An alternative to econometric models. *International Statistical Review/Revue Internationale de Statistique*, 123-137.
- NIMET. (2018). Numerical Weather Data. Accessed: Jun. 21, 2018. [Online]. Available: <https://nimet.gov.ng>.
- Olatomiwa, L., Mekhilef, S., Shamshirband, S., & Petkovic, D. (2015). RETRACTED ARTICLE: Potential of support vector regression for solar radiation prediction in Nigeria. *Natural Hazards*, 77(2), 1055-1068.
- Osinowo, A. A., Okogbue, E. C., Ogungbenro, S. B., & Fashanu, O. (2015). Analysis of global solar irradiance over climatic zones in Nigeria for solar energy applications. *Journal of solar energy*, 2015, 9.
- Persson, C., Bacher, P., Shiga, T., & Madsen, H. (2017). Multi-site solar power forecasting using gradient boosted regression trees. *Solar Energy*, 150, 423-436.
- Ram, K., Sarin, M. M., Sudheer, A. K., & Rengarajan, R. (2012). Carbonaceous and secondary inorganic aerosols during wintertime fog and haze over urban sites in the Indo-Gangetic Plain. *Aerosol and Air Quality Research*, 12(3), 359-370.
- Raza, K., & Jothiprakash, V. (2014). Multi-output ANN model for prediction of seven meteorological parameters in a weather station. *Journal of The Institution of Engineers (India): Series A*, 95(4), 221-229.
- Renewable Energy Master Plan (REMP). Energy commission of Nigeria, Federal Ministry of Energy Resources, Nigeria; 2007.
- Xue, X. (2017). Prediction of daily diffuse solar radiation using artificial neural networks. *international journal of hydrogen energy*, 42(47), 28214-28221.