Solar Radiation Forecasting using Neural Networks: a Modeling Issues

Yashwant Kashyap
Student, School of Engineering,
Indian Institute of Technology Mandi,

Ankit Bansal
Faculty, Mechanical Engineering Department,
Indian Institute of Technology Roorkee

Anil K Sao
Faculty, School of Computing and Electrical Engineering,
Indian Institute of Technology Mandi

***Abstract*: A review paper has been prepared on the design of neural networks architecture for setting up the number of delay, neurons and activation function. The anticipated model was initiated and validated with solar radiation data. The consequences are evaluated with a variety of statistical errors and this effort verifies the capability of the ANN to accurately simulate hourly global solar radiation. ANN model is a well-organized technique by means of the regular meteorological data to estimate the radiation in case of India. This document used eight special neural network models for assessment. These are, Elman back propagation neural network, feed-forward network, feed-forward back propagation, radial basis neural network, probabilistic neural network, cascaded feed forward back propagation, custom network-1 and custom network-2. On top of six they are well recognized model and frequently applied, last two models are custom based. Outcome demonstrates by means of flexibility of special models and parameter range associated to its accuracy evaluation. This paper added advantages of solar radiation forecasting and model priority.**

Keywords*: Artificial Neural Network (ANN); Forecasting; Modeling; Global Horizontal irradiance (GHI);

I. INTRODUCTION

The solar power generation and synchronization tough for utility due to insecurity in the net accessibility of the solar energy constructs. The influence of diverse climatic situation, utility running in the area of solar generation are required to develop their prediction ability of power generation under such diverse condition. The cloud cover, aerosols, humidity etc. [1, 2] are most strongly affected the solar energy received on Earth's surface. The low solar radiation experience from unexpected short-term deviation due to moving clouds in an atmosphere with spread cloud circulation. Existence of broken clouds leads to regular fluctuations in direct normal incident solar radiation (DNI) in addition to diffuse radiation (DHI) of the atmosphere. The DNI values may experience fall to the extent that 100%, and occasionally even strengthening by reason of cloud perimeter special effects [1]. Clouds a diversity nature (shape, volumes and composition) influence received solar radiation all the way through reflection, absorption and scattering [3, 4].

In solar radiation at different latitudes and climates of specific site and its associated effort carryout by countries such as Saudi Arabia, India, Algeria, UK etc…, of different artificial neural networks (ANN) model are used [5]. At present several queries continually grow- how delays and neurons in every layer should be used, all over arrangement of neural networks regression of the popular purpose. After long years has been passed first launched neural networks, even though no precise method to answer this uncertainty. It has a strong impact on regression consequences, derived from the choice of network architecture. traditionally records of architecture has been formed on trial and error, except for a while it pursue experimental [6]. Since variable nature of radiation, time delay is frequently used to train neural network due to restriction in rate of data processing. For a moment it source for variation and uncertainty of neural network. Therefore, delayed network permanence study has extra significance on practical and theoretical prospective [7]. In recent times, neural dynamics of artificial networks has been raising awareness on the outcome of delay. The delay has consequence on neural network, somewhat it insecure to stable network and slightly it modify dynamics superbly [8]. Thus, it is essential for the neural network has to arrive at globally stable area at balance situation. Numerous things regarding the dynamic presentation of ANN by means of delays have been confirmed. It is set up that balance and stability characteristic of ANN has recognized the different sufficient and unique situation derived from degree of difference delay [7]. In implement, apply the delay such a way that the networks arrive at only one of its kind of balance point for taken as a whole stable neural network. In order that network able of resolve concurrent difficulty. In addition, it is easier to optimize the logical difficulty. moreover achieved the convergence rate and precision of network by means of delayed [9]. At present are numerous methodologies to build the artificial network in a positive or else negative means. a number of familiar method to make a decision whether a certain integer of delay is most favorable, by means of cross validation and early stopping [10, 11].

Furthermore preparation of artificial network the involvement of neuron is extensively significant, depends on network oversetting and under fitting state show based on the range of neuron. To realize most favorable neurons derived from the smallest error has been one of the principles. The dissimilar neuron has been examined derived from error principle that appropriately converge the network. Consequently the architecture of neural network has turn out to be investigated.

Several researchers attempted and recommended methods for setting neuron in artificial neural network. In 2008, Shuxiang et al is inspected novel method for optimizing of neuron in data mining [12]. In 1998, Osamu recognized mathematical estimate of number of neuron by neural network [13]. In 1997, S.Tamura et al is examined supplementary method produced on Akaike information criteria that improve neuron individually [14]. In 2003, Zhange et al practical process is improvement to cover range of neuron [15]. In 1995, Jin Yan Li et a recommended to estimate neurons of forecasting time series [16].

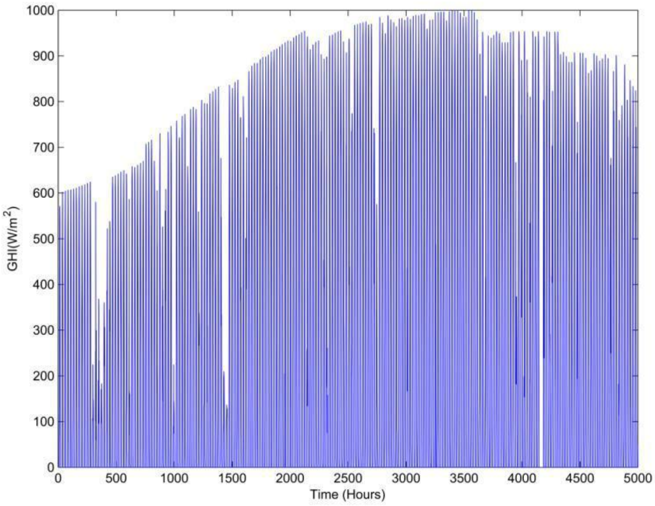

Fig.1 Time series GHI plot of 5000 hours

The varying property of the modeled structure is one of the most desirable feature of ANNs to become standardized their performance using of activation functions [17]. Jordan obtainable the subsequent possibility in a binary categorization difficulty of the logistic function which is a natural illustration and Liu enhanced the sigmoid and Gaussian basis those are construction of generalized networks with two unusual activation function [18, 19]. Sopena et al. obtainable second orders of magnitude is quicker whereas simplification ability enhances a number of test presentation that multilayer feed forward networks through a sine activation function [20, 21]. Major unresolved difficulty of transfer function choice to best execute and afford them nonlinear mapping prospective, is a very important element of ANNs modeling. Certainly, the difficulty among transfer functions is that extra elements of an ANN, sternly nonlinearity usually required for the modeling of natural fact with no hypothetical environment for their choice [22].

This paper considers eight different neural network models for comparison. These models are, feed-forward network (FFN), feed-forward back propagation (FFB), radial basis neural network (RBN), probabilistic neural network (PNN), Elman back propagation neural network (EBNN), cascaded feed forward back propagation (CFB), custom network-1 (C1) and custom network-2 (C2). Above all, six of them are well known model and mostly used, last two models are custom models. All models have been compared as testing parameters of delay, neurons and activation functions.

II. DATA PROCESSING

We use the data provided by "Sunny" satellite, available on the website of Indian ministry of renewable energy (MNRE). The hourly GHI are collected at specific time zone 5.5 as plotted in Fig.1 for 5000 hours of 2008 at location IIT Mandi (31.7069° N, 76.9317° E, Coordinates), HP India. Most significant approach of data mining for neural network is scaling of input and target data. Hence, normalization is used with standard deviation and mean of training data set. The data set used a zero mean and unity standard deviation of following equation:-

$$= (- \qquad\qquad) * (\quad \underline{\qquad}) + \tag{1}$$

where ,,, ,, data sets are, data, mean and slandered deviation of target and data, mean and slandered deviation of training set respectively. After training, produced output has also contained with '0' mean and '1' standard deviation to getting back the original target set.

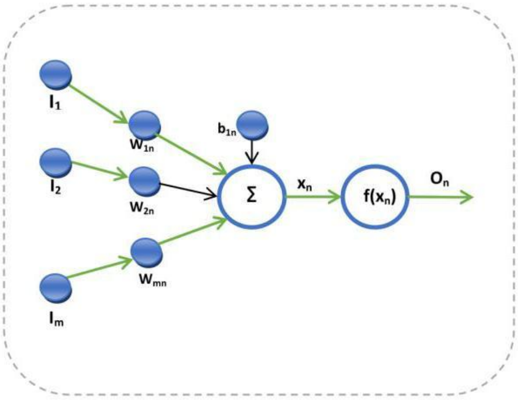

Fig.2 Typical neuron representation

III. NEURAL NETWORK BASICS

A. ANN FOR HOURLY SOLAR IRRADIATION

Artificial neural networks are knowledge developing systems with the ability of learning through instance [23]. It's set up on theory derived from neurobiology, neural are self-possessed by a set of interrelated processing units, describe as *neurons (n)*. The neurons develop the signals offered to the neural network by collecting every incentive and by changing the entirety rate by a function; that is described as the *activation function (f)*. The incentive to and from a neuron are customized by the actual rate is described *synaptic weight (w)*, which describes the individual association among neurons. Fig.2 show a classic exhibition for a typical neuron , where $1, 2, 3, \ldots$, are the incentive signals, $1, 2, 3, \ldots \ldots$, are the synaptic weights is a bias value is the activation potential is the neuron output signal, and (.) is the activation function (usually implemented as a non-linear sigmoid function). Afterward, as of Fig.2, one can monitor that the neuron productivity is given by:

$$= \{ \qquad + \Sigma_{=1} \qquad \} \tag{2}$$

Network architecture is the name specified to the preparations of neurons into layers and there association. Usual neural networks have the subsequent architecture: *input layer,* where the input incentive is obtained to the network; *Output layer,* the last layer of the network, where the outputs are given. *Hidden layers,* internal layers of a network or another layer in between them is hidden layer. It has no communication with the outer word. Such usual network architecture is usually referred to as a multilayer neural network [24]. Once trained, one can presume that the network accumulated the information supplied to it. However, the information in a neural network is not accumulated in a particular localization. It based on its topology and the amount of the weights in the input layer. The simplification of an artificial neural network is the capacity to replicate preferred signals for different input signals that have not been used throughout the network training, or also, that it is capable to hold the dynamics of the system being followed [25].

However considering the number of neurons in each layer is a compound problem. An Artificial intelligent network contains following layers neurons: *Input Neurons*: Which interconnects with the peripheral environment and offerings a design to the neural network is called *Input layer neurons*. When a design is obtainable to the input layer neurons, the output layer neuron will create additional design. Each input neuron should characterize nearly dependent variable that has an impact above the output of the network. *Hidden Neurons:* The group of neurons of hidden layer in which has activation function applied on it, and deliver a transitional neuron between the input and the output. However still nobody was perfect even several studies have been finished in estimating the number of neurons in the hidden layer. Additional query rises that how numerous hidden neurons has to be used when allocating with multiple problem. The case where under fitting may occur, when the numbers of neurons are less as associated to require for complex data set and also over fitting may occur, if excessive neurons are used in the network. For computing the number of neurons in each hidden layers the numerous approaches have been used, which do not deliver the appropriate method. *Output Neurons:* The output neurons essentially used to extant a design to the external environment. The output neuron can be straight copied back to the input neuron from obtainable design. The number of output neurons that should be straight linked to the kind of work by which the neural network is implement [26].

Correspondingly, the ultimate outcome of ANN would be in group of weights and input variables for linear and nonlinear process. The multilayer perception networks are frequently utilized ANN model. Alternatively, ANN can be classify such as radial basis, general regression, probabilistic, recurrent, cascaded correlation, hybrid, functional link, Kohonen, hetero associative, Gram-Charlier, Adaline, learning vector quantization and Hebb networks [27, 28]. One output nodes at outer layer were applied for forecasting of GHI. The precision of the eight unusual ANN models are compared to conclude the most correct model of hourly solar radiation, as described below:

Feedforward Network (FFN): It design with a sequence of layers are, the network input has association with the first layer, the preceding layer has an association from each successive layer, finally the network's output constructs by the last layer. Consequently, weighted informational relocate one layer to a new layer that why it called *Feedforward*. It can be utilized for any type of input to output planning with one hidden layer and sufficient neurons in the hidden layers can fit any fixed input-output planning difficulty. Specific description of the Feedforward network comprises fitting and pattern recognition networks and cascade forward network are also considered below. Feedforward networks form with hidden neurons (default = 10) and transfer function (default = 'LMA') acquires these influence [29, 30].

Feed-Forward Back propagation Network (FFB): It is design of multi layers using the 'dot product' weight function, 'net sum' input function, and the particular transfer functions. The weights coming from the input in the primary layer, similarly the preceding layer and all layers have biases taken from each following layer weight and biases are initialized with 'Nguyen Widrow layer initialization' method. Adaption revises weights with the particular learning function, which is done with 'sequential order incremental training with learning function'. FFB obtain optional inputs: transfer function of distinct layer, default is 'hyperbolic tangent' for hidden layers, and 'linear' for output layer. The back propagation network training function is LMA very fast as default, however it involve memory requirement to train and the weight/bias learning function, default is 'Gradient descent'. Performance function is 'mse' as default. Data division function is divide targets into three sets using random indices, and returns an multilayer feed-forward back-propagation network [29, 30].

Cascade-Forward Back propagation Network (CFB): It is comparable to FFN, except contain an association from the input, each previous layer to following layers as shown in Fig.3. As with FFN, any fixed input output association randomly well given sufficient hidden neurons layer can study with two or additional layer cascade network [27, 28].

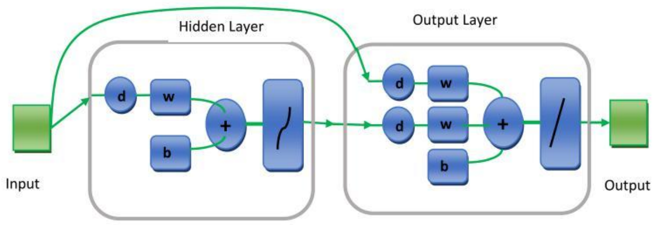

Fig.3 Typical neuron representation of Cascade-Forward Back propagation Network

Elman Back Propagation Network (ELM): The Elman Neural Network is comparable to partial recurrent neural network. It designs with two layer of back propagation, as shown in Fig.4 feedback from the output to hidden layer and input layer. ELMS are extensively used for categorization and regression problem. In several cases competence of ELM are not excellent for definite state like memory ability with back propagation. Consequently may unusual network design have been developed to recover the competence of ELM with back propagation. Several time it is significant to add more feedback to ELM that significantly useful for junction and local minima problem. The 'LMA' and 'resilient back propagation' etc are not optional for ELM, which are algorithm receiving large step sizes [31-33].

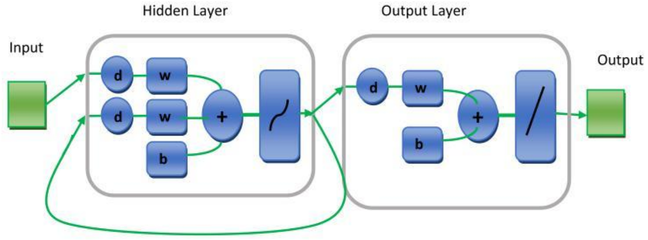

Fig.4 Typical neuron representation of Elman Network

Radial Basis Neural Network (RB): It constructs a '2' layer networks the primary layer has radial basis neurons using Euclidean distance to estimate weighted inputs and product for net inputs. Similarly second layer used purelin with dot product for weighted input and summation for net inputs, both layers have biases. The hypothesis of function estimate used to plan the radial basis network. It is motivated by MLP network to learn with unusual hidden layer and approximate function. Radial basis function has rather unrelated approach, which comprise two layer of FFN. The set of Gaussian function (radial basis) is utilizes to realize the hidden neuron. Besides this the output layer neuron are executes with linear summation function just like MLP. Throughout the training processes, weight of each inputs equivalent to hidden layer are determined, the hidden to outer layer weight could be recognize. It is much faster than other networks and good in interpolation [27-30].

Probabilistic Neural Network (PNN): It is FFN that can be trained to classify inputs respective to target classes. The objective for PAT should summation of inputs of all zero values except for a 1. The *probability density function (PDF)* is used as assessment for probabilistic neural network supported on the *Bayesian classification* model presented by Donald F. Specht in 1990 [34]. It has strong statistical approach for resolve unusual type of problem with simplicity. PNN is somewhat dissimilarity from back propagation and it is valuable for network architecture plan. It is comparable in some situation of Feedforward but dissimilarity in term of learning; it has no weight in hidden layer in addition that learning is complete in supervised mode. In every hidden neuron can be present as trial vector. These trial vectors are used to hidden neuron in lack of weight. PNN are typically used for resolving categorization problem [27, 28].

Custom Network-1and-2(C-1 and C-2): To begin modify network using neural network toolbox offer special network choice. This is feasible to use unusual function in same database, since its flexibility and object orientated design. These types of afferent assist to apply extensively in numerous special custom ways of network demonstration. To construct custom arrangements, start with an unfilled network and set its feature as favored. The network used many function features that have been set in many ways, as desired for network architecture. It begins with simplest model and goes to complex network architecture. In this part face the slight additional complex network than normal network mentioned earlier. The input network recognizes normalized value range from -1 to +1 of radiation. The number of layer used for this network is six, start with the Nguyen Widrow method and trained with the Levenberg-Marquardt back propagation as described below. Any input and output vectors in output layer will learn to associate the connected target outputs with minimal mean squared error (mse) including weights and biases. The only difference between custom one (C -1) and custom two (C-2) model are hidden layer. As point out before custom one has six (6) layers together with input and output layer and custom two has three (3) layers [27-30].

B. NEURAL NETWORK TRAINING

Levenberg-Marquardt algorithm (LMA) is a difference of the Newton's technique for reducing tasks which are sums of squares of new nonlinear tasks. The LMA delivers improved performance as soon as associated with classic back propagation processes. Since Newton's technique the networks modernize law is:

$$ {}_{+1} = - {}^{-1} \qquad (3) $$

$$ {}_{+1} = - [+]^{-1} \qquad (4) $$

Where , , , , , are the network weight matrix, number of repetition, the Hessian matrix, the gradient matrix, the Jacobian matrix, the identity matrix and a scalar, respectively. The Hessian matrix is estimated in relations of the Jacobian matrix, . The scalar is offering significant part to the LMA. When equivalent to zero the weight are essentially the Gauss Newton methods. When adequately large Eq. (5) are converts gradient descent with minimum stage of magnitude. In selecting the correct rate of the LMA delivers an effective cooperation among the excessive performance of the Newton's technique then the certain merging of the gradient descent method [24]. The projected networks were trained by 70% of the delivered data, whereas the continuing 15% was usages to validate and remaining 15% was for test the trained network. Thereafter, the trained networks were used for forecast using last 100 days data. The suggested ANNs forecast the solar radiation value (GHI), and at that time the forecast effects were equated through the measured data. Individually network was verified separately by the hourly radiation values, and all of the suggested ANNs were matched collectively by the RMSE values of solar radiation. These different networks were compared with training RMSE error.

IV. MODEL EVALUATION CRITERIA

To assess the capability of the model in two parts, one set of inputs variable belongs to training processes and second data set for the validation. These validation sets are called testing set. Testing processes involve comparison of different model result. Finally, model has been selected based on lowest forecasting error. The estimation of error can be many forms such as root mean square error (RMSE), MAE (Mean Absolute Error), Mean Bias Error (MBE), correlation coefficient (r) and Factor of determination (R^2).

$$= \frac{1}{} \Sigma | \hat{} - | \qquad (5)$$

$$= \frac{1}{} \Sigma - \hat{} \qquad (6)$$

$$\sqrt{\frac{1}{} \hat{}\ {}^{2}} \qquad (7)$$

$$= \frac{- \hat{} \quad \bar{\hat{}}}{\sqrt{(\Sigma \quad \Sigma (\hat{} \ {}^{2} \quad (\quad)^{2})}} \qquad (8)$$

$$^{2} = 1 - \frac{\Sigma \quad (-)^{2}}{\Sigma =1 (-)} \qquad (9)$$

In (5-9) are different expressions for error estimation where, represent measured value at forecasted horizon and ^ is forecasted value. Also n represent the total number test sample. There are many time series data set model development need to follow standard procedure. It is suggested to follow three basic processes which are identification, assessment and validation checks. This validation process defines the model accuracy and stop iteration process for ANN model [35].

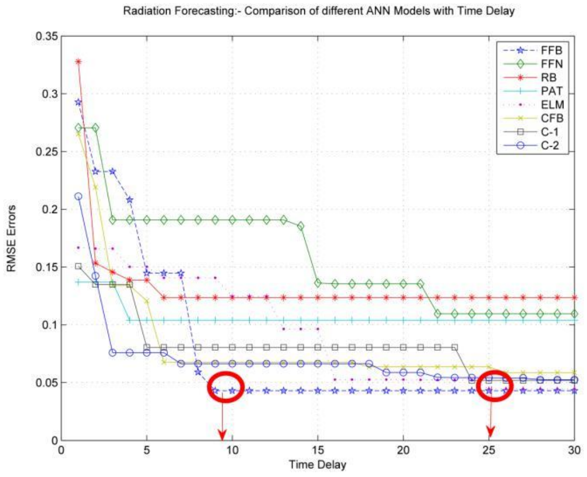

Fig.5 Radiation Forecasting: - Comparison of ANN Models with Delay

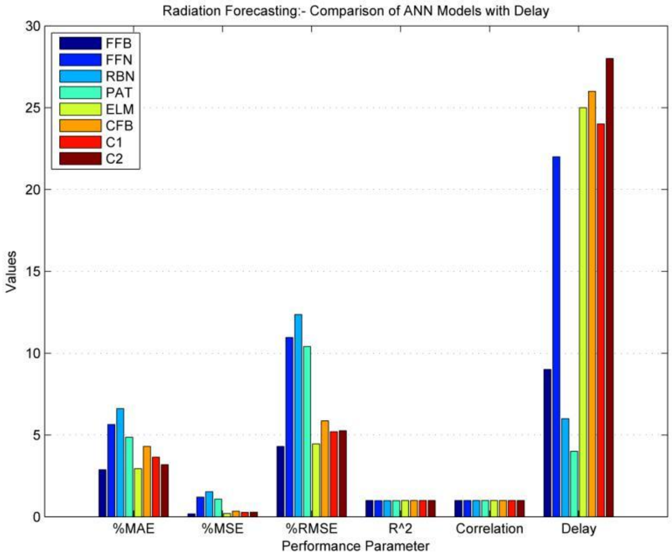

Fig.6 Radiation Forecasting: - Comparison of ANN Models with Delay

A. MODEL BEHAVIOR WITH DELAYS

During training the networks, with diverse groupings of the network parameters have been observed. These adaptable factors are number of layers, neurons and delays. In Altogether different cases a single layer neural network executed superior than a more layers. In these sections used one input with time delay among time series data of global solar radiation. The hidden layer required as often *(10)* neurons, and the input delay were of varies from one to thirty *(30)* of equal interval (that is one). For GHI, the wrong alarm frequency was constantly low while trained on the maximum number of past data [36].

A neural network with time delay can be explained with activation function as follows:

$$( ) = (\Sigma \quad \Sigma \quad (-) . (-)) \qquad (10)$$

Here ( ) is output of neurons at time . ( ) is the weight between and neuron at same time . The and are the total number of delay and neuron that connect with neuron. Again actuation function normally prefer logistic sigmoid. In this case, delay is used for input layer only and considered configuration : : 1, where = 1, = 10 and delay D= 30 for all different models. All configuration models are tested for 30 times at continuous time delay 1 to 30. The network has to optimize at minimum root mean square (RMSE) on the training data set [37].

Fig.5 illustrated an evaluation of observed and forecasted data by the recommended ANNs on hourly global irradiations. Established upon FFB and ELM beaten the additional techniques by a very small dissimilarity in global irradiation forecasting. For now, the FFB was superior to the ELM in forecasting radiation, whereas both had comparable values in forecasting global radiation. The RB was the poorest between the suggested ANN with a high error in global irradiation forecasts. It stat with 32% RMSE error at one delay and end with 12% at six delays further remains constant for all other delay. **Fig.6**, demonstrated a summary evaluation of the suggested models by percentage performance of the MAE, MSE and RMSE as well as R^2 and Correlation coefficient (r). These results shows the solar radiation forecasting differ too much respective of ANN models. Forecasting is effected based on number of time delay applied in different ANN models. The delay decides the convergence property of ANN models. Fig.5 shows that some of the model converges very fast with minimum error as of FFB with 9 time delay and around 4.3 percent of RMSE error. Similarly other comparable method of ELM shows the 4.5 percent of RMSE error but long time delay *(25)*. In other hand both the model FFB and ELM start with 29% and 16% of error at one time delay. If apply the curve fit for all model in Fig.5 follow the six order of polynomial. The custom networks follow considerably good profile at 24 and 28 delays of 5.2 and 5.26 percent error of C1 and C2 models, respectively. The C1 model with delay 24 is well understood because the radiation follows the 24 hour of time interval every day. The hourly correlation factor between two clear skies day are almost one day, which is perfect for time series prediction considering the same interval.

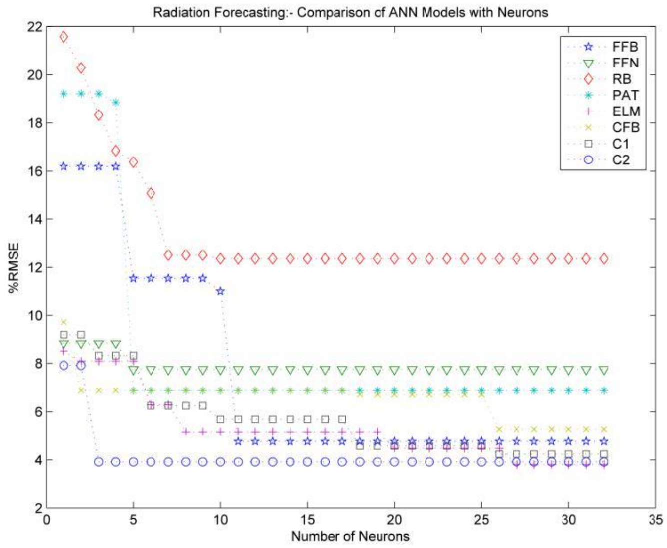

Fig.7 Radiation Forecasting: - Comparison of ANN Models with Neurons

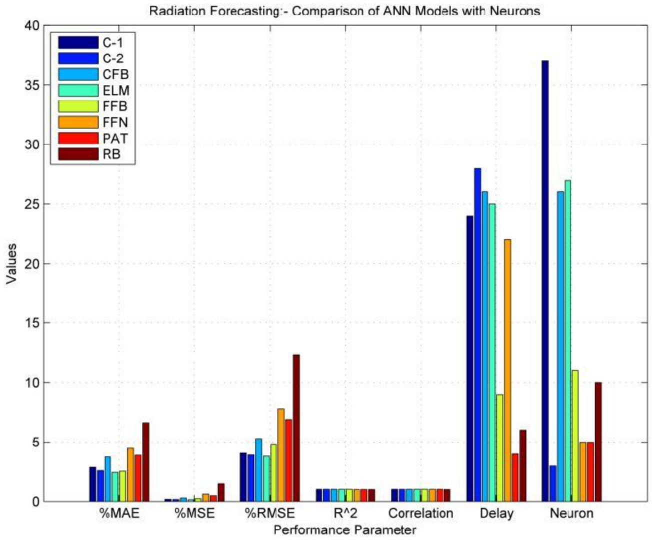

Fig.8 Radiation Forecasting: - Comparison of ANN Models with Neurons

B. MODEL BEHAVIOR WITH NEURONS

To fix the number of neurons in network, many researchers suggested several methodologies. The network can be starts with positive methodology that undersized network and at that time increases extra neuron. To be securing the difficulties of appropriate number of neurons for a specific problem, the relevant opinions of the projected method are argued. The outcome with least inaccuracy is definite as finest explanation for setting neurons. Simulation outcomes of expected solar radiation are in decent agreement through the measured values [38, 39]. As earlier explained about data are distributed into training and testing set, that executes to evaluate the error from the training set executes in ANN learning and testing set. The testing act ends to control the neurons continues to increase; training has created to fitting the training records and over fitting arises. As the results shows, it is detected that the some suggested method offers superior outcomes when applied different neurons. The numerical errors are considered to calculate the accuracy of network. The examination of solar radiation calculation is accepted through the suggested criteria. **Fig.7** demonstrates that the projected model offers superior values for numerical errors in evaluation with additional neurons. Trial-error basis is the current technique to control number of neurons. Starts by minimum number of neurons and increases neuron to its maximum limit. The drawback is that it is time taking and here is no surety of setting the neuron. The particular measures for eight models used maximum 40 neurons and target for a minimal RMSE value in comparison [15, 16, 38, 39].

In order to find accuracy of the results was tested by using the training sets and accuracy ratio was computed in each case. Fig.8 shows the results with all errors, performance coefficients, delays and different neurons for each model. It is perfect that simplification capability is increasing while the number of neurons is increasing. In solar radiation estimation problem, the precision degree of the productivity was 96.19% while27 neurons of ELM model at same delay (25). Comparatively same result obtained in Custom model 2 with 96.08% accuracy at 2 neuron and 28 delay. Though, to train greater number of layers it needs additional time. While there are simply 2, 3 and 6 layers in solar radiation network for default, custom-1 and custom-2 models that can be trained for 100 epochs. On the other hand the numbers of training epoch involve to train network through3 or 6 layer are considerable larger than that (about 100). i.e. the upgrading of the precision is insignificant, equating with the effort and time involved to train the neural networks. It can be described by the accuracy, in this problem when there 2 or 3 layers is much higher than that of 6 layers. Hence, we can consider the network with 2 hidden layers, which provides the highest value, can be considered as the most appropriate network for this problem.

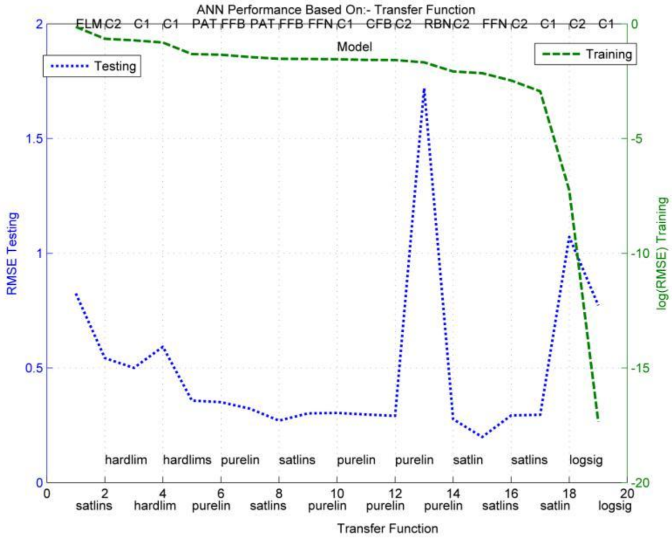

Fig.9 Radiation Forecasting: - Comparison of ANN Models with Transfer Function

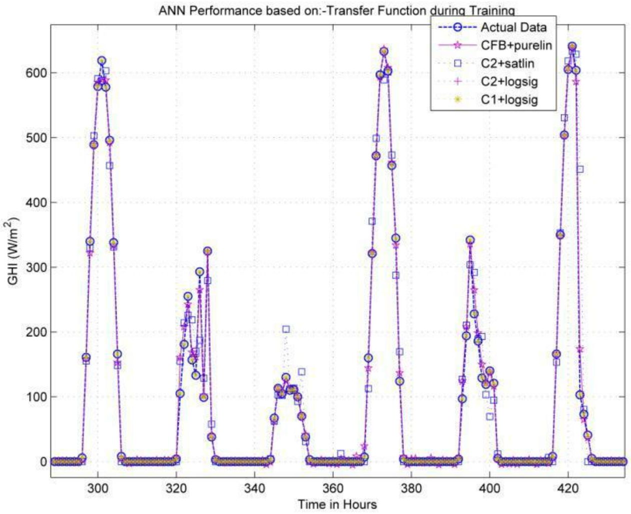

Fig.10 Radiation Forecasting: - Comparison of ANN Models during training with Transfer Function

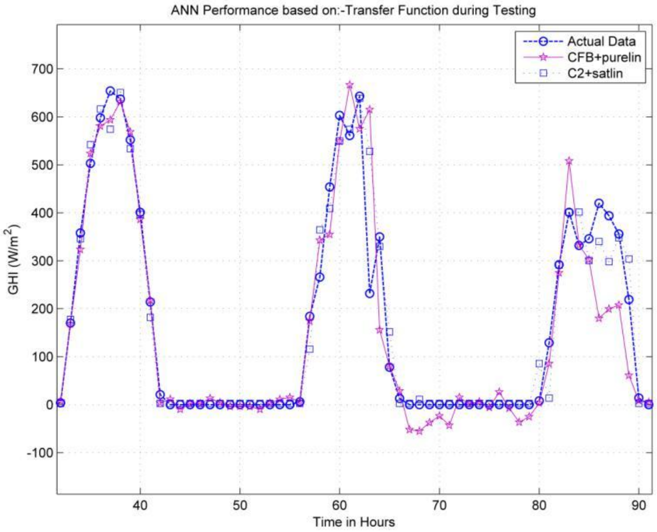

Fig.11 Radiation Forecasting: - Comparison of ANN Models during testing with Transfer Function

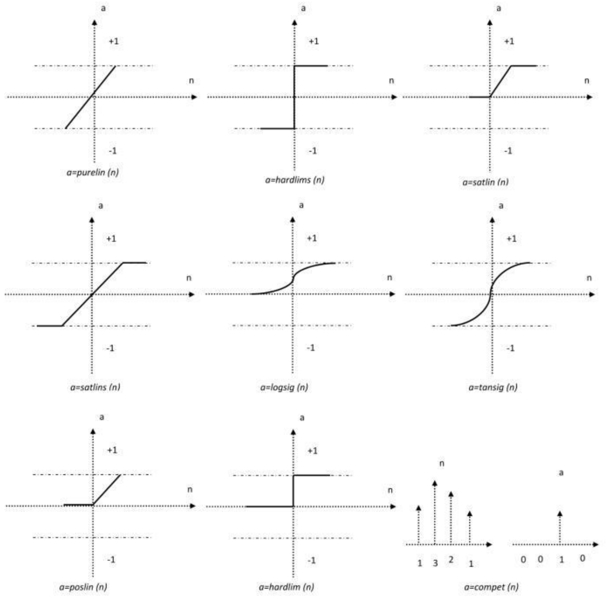

Fig.12 ANN Models used with different Transfer Function

C. MODEL BEHAVIOR WITH TRANSFER FUNCTIONS

Numerous general type activation functions employ in ANN are as described below. These are defined, respectively, as hard (*hardlim*), symmetric hard (*hardlims*), linear or bipolar *(purelin*), saturating linear (*satlin*), symmetric saturating linear (satlins), sigmoidal or log-sigmoid *(logsig*), hyperbolic tangent sigmoid (*tansig*), positive linear (*poslin*) and competitive (*compet*) shown in Fig.12. The magnitude of these functions varies between 0 and 1, -1 or +1, the *linear* can be *unipolar* or *bipolar* with rate of infinity transforms to *threshold* or *signum* function, respectively. The *tansig* and *purelin* are usually utilizing, all of these are squashing in character and they control the neuron reply among the asymptotic rates. *Nonlinear* add to the nonlinear character of neuron that allow nonlinear in-out mapping, though with linear function lost is nonlinearity [17, 20, 21].

In this section, to evaluate their performance by using the same data for different functions rely on special number of iterations. For all the functions utilized the number of neurons in the hidden layer as mention in neuron section for different models in Fig.8. After presenting the graphs for different parameters Fig.10, explanation of their outcome will track here. The different number of delay in first section, different number in second section of hidden neurons to compare different models with differences activation functions described here. According to the first graph (Fig.10**)** generated by the all models compare with the different function, each model perform significantly different behavior with its one kind of function. *Satlins* functions effect in the mainly booming for all the experiment outputs. The group (Fig.11) of effect is comparable to the Fig.10. But, the accuracy of the algorithms was totally different in training and testing cases, higher accurate result (lowest RMSE value) show the inverse effect during testing case. This may the case of over or under fitting of model with data. The remains functions utilized in this learning were not booming or precise sufficient in this set. According to the Fig.9, we found *logsig* function with C-1 model the best activation function for training. Still with observe to testing create that the precision of *satlins* function was a lot improved then other functions in case of C-2 model. This state clarify that entirely Root Mean Square Error (RMSE) according to iterations cannot decide the network precision. Therefore, to get the actual precision result is from testing. Fig.9 demonstrates the error plot for every function for special models, which might differ from 1 to 8 models, and the iterations from 100 to 1000. Fig.10 and Fig.11 demonstrated in both cases clear and cloudy days of model performance, however model perform well during cloud day are preferred most. In the training case shows six continuous day plot including cloudy cases, four models most suitable out of 72 different groups of model. Same for the testing cases shows three days of data (dot line with circle) and forecasted plot (dotted line with rectangle) of most accurate custom model (C-2) with satlin activation function, nearly same result of CFB with purelin (slid line with star).

V. CONCLUSION

This paper concludes the modeling feature of artificial neural network. Solar irradiations data are used with anticipated model were initiates and tested. The results are evaluated with various statistical errors. This work authenticates the capability of the ANN to accurately replicate hourly global solar radiation. Through using the usual meteorological data, accuracy assessment of the hourly solar radiation can be accomplished. From decade this model has been extensively used for special application, due to its dynamic nature, modeling need professional advice. In these section put more importance on parameter that illustrated the progress of ANN architecture, which are delay, neuron, and transfer function. Results show with so much flexibility in different models and parameter selection related to its accuracy comparison. This paper contributes in the field of solar radiation forecasting and appropriate model selection.

REFERENCE

[1] R. Marquez and C. F. Coimbra, "Intra-hour DNI forecasting based on cloud tracking image analysis," *Solar Energy,* vol. 91, pp. 327-336, 2013.

[2] M. J. Ahmad and G. Tiwari, "Solar radiation models—a review," *International Journal of Energy Research,* vol. 35, pp. 271-290, 2011.

[3] V. Badescu, "Correlations to estimate monthly mean daily solar global irradiation: application to Romania," *Energy,* vol. 24, pp. 883-893, 1999.

[4] K. Bakirci, "Models of solar radiation with hours of bright sunshine: a review," *Renewable and Sustainable Energy Reviews,* vol. 13, pp. 2580-2588, 2009.

[5] T. Khatib, A. Mohamed, K. Sopian, and M. Mahmoud, "Assessment of artificial neural networks for hourly solar radiation prediction," *International Journal of Photoenergy,* vol. 2012, 2012.

[6] D. Stathakis, "How many hidden layers and nodes?," *International Journal of Remote Sensing,* vol. 30, pp. 2133-2147, 2009.

[7] H. Zhang, Z. Wang, and D. Liu, "Global asymptotic stability of recurrent neural networks with multiple time-varying delays," *Neural Networks, IEEE Transactions on,* vol. 19, pp. 855-873, 2008.

[8] L. Wang and X. Zou, "Convergence of Discrete–Time Neural Networks with Delays."

[9] X.-G. Liu, R. R. Martin, M. Wu, and M.-L. Tang, "Global exponential stability of bidirectional associative memory neural networks with time delays," *Neural Networks, IEEE Transactions on,* vol. 19, pp. 397-407, 2008.

[10] L. Prechelt, "Early stopping—But when?," in *Neural Networks: Tricks of the trade*, ed: Springer, 2012, pp. 53-67.

[11] R. Setiono, "Feedforward neural network construction using cross validation," *Neural Computation,* vol. 13, pp. 2865-2877, 2001.

[12] I. Gonzalez-Carrasco, A. Garcia-Crespo, B. Ruiz-Mezcua, and J. L. Lopez-Cuadrado, "Dealing with limited data in ballistic impact scenarios: an empirical comparison of different neural network approaches," *Applied Intelligence,* vol. 35, pp. 89-109, 2011.

[13] O. Fujita, "Statistical estimation of the number of hidden units for feedforward neural networks," *Neural Networks,* vol. 11, pp. 851-859, 1998.

[14] S. i. Tamura and M. Tateishi, "Capabilities of a four-layered feedforward neural network: four layers versus three," *Neural Networks, IEEE Transactions on,* vol. 8, pp. 251-255, 1997.

[15] Z. Zhang, X. Ma, and Y. Yang, "Bounds on the number of hidden neurons in three-layer binary neural networks," *Neural networks,* vol. 16, pp. 995-1002, 2003.

[16] J.-Y. Li, T. W. Chow, and Y.-L. Yu, "The estimation theory and optimization algorithm for the number of hidden units in the higher-order feedforward neural network," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, 1995, pp. 1229-1233.

[17] B. DasGupta and G. Schnitger, "Analog versus discrete neural networks," *Neural Computation,* vol. 8, pp. 805-818, 1996.

[18] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," *Machine learning,* vol. 37, pp. 183-233, 1999.

[19] X. Yao and Y. Liu, "A new evolutionary system for evolving artificial neural networks," *Neural Networks, IEEE Transactions on,* vol. 8, pp. 694-713, 1997.

[20] J. M. Sopena, E. Romero, and R. Alquezar, "Neural networks with periodic and monotonic activation functions: a comparative study in classification problems," in *Artificial Neural Networks, 1999. ICANN 99. Ninth International Conference on (Conf. Publ. No. 470)*, 1999, pp. 323-328.

[21] B. Karlik and A. V. Olgac, "Performance analysis of various activation functions in generalized MLP architectures of neural networks," *International Journal of Artificial Intelligence and Expert Systems,* vol. 1, pp. 111-122, 2011.

[22] H. Yonaba, F. Anctil, and V. Fortin, "Comparing sigmoid transfer functions for neural network multistep ahead streamflow forecasting," *Journal of Hydrologic Engineering,* vol. 15, pp. 275-283, 2010.

[23] S. Haykin, *Neural networks: a comprehensive foundation*: Prentice Hall PTR, 1994.

[24] E. Belo and R. Ortolan, "Application of Time-Delay Neural and Recurrent Neural Networks for the Identification of a Hingeless Helicopter Blade Flapping and Torsion Motions."

[25] N. Saravanan, A. Duyar, T.-H. Guo, and W. Merrill, "Modeling space shuttle main engine using feed-forward neural networks," *Journal of guidance, control, and dynamics,* vol. 17, pp. 641-648, 1994.

[26] S. Karsoliya, "Approximating number of hidden layer neurons in multiple hidden layer BPNN Architecture," *Internat J Eng Trends Technol,* vol. 3, pp. 714-717, 2012.

[27] M. Caudill and C. Butler, *Understanding neural networks: computer explorations: a workbook in two volumes with software for the macintosh and pc compatibles*: Mit Press, 1994.

[28] K. Mehrotra, C. K. Mohan, and S. Ranka, *Elements of artificial neural networks*: MIT press, 1997.

[29] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural network design*: Pws Pub. Boston, 1996.

[30] R. R. Trippi and E. Turban, *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real World Performance*: McGraw-Hill, Inc., 1992.

[31] N. Bhat and T. J. McAvoy, "Use of neural nets for dynamic modeling and control of chemical process systems," *Computers & Chemical Engineering,* vol. 14, pp. 573-582, 1990.

[32] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation,* vol. 1, pp. 270-280, 1989.

[33] T. Koskela, M. Lehtokangas, J. Saarinen, and K. Kaski, "Time series prediction with multilayer perceptron, FIR and Elman neural networks," in *Proceedings of the World Congress on Neural Networks*, 1996, pp. 491-496.

[34] D. F. Specht, "Probabilistic neural networks," *Neural networks,* vol. 3, pp. 109-118, 1990.

[35] H. T. Pedro and C. F. Coimbra, "Assessment of forecasting techniques for solar power production with no exogenous inputs," *Solar Energy,* vol. 86, pp. 2017-2028, 2012.

[36] E. W. Saad, D. V. Prokhorov, and D. C. Wunsch, "Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks," *Neural Networks, IEEE Transactions on,* vol. 9, pp. 1456-1470, 1998.

[37] T. Taskaya-Temizel and M. C. Casey, "A comparative study of autoregressive neural network hybrids," *Neural Networks,* vol. 18, pp. 781-789, 2005.

[38] J. Zhang and A. J. Morris, "A sequential learning approach for single hidden layer neural networks," *Neural networks,* vol. 11, pp. 65-80, 1998.

[39] K. G. Sheela and S. Deepa, "Review on methods to fix number of hidden neurons in neural networks," *Mathematical Problems in Engineering,* vol. 2013, 2013.

Author Profile

Mr. Yashwant Kashyap is a scholar, pursuing his Engineering in Indian Institute of Technology. His research interests include Fuzzy logic and Neural Networks. He has published many research papers in International Journals

Ankit Bansal is working as faculty in Mechanical Engineering of Indian Institute of Technology. His research interests include Fuzzy logic and Neural Networks. He has published many research papers in International Journals.

Anil K Sao is working as faculty in School of Electrical Engineering of Indian Institute of Technology. His research interests include Fuzzy logic and Neural Networks. He has published many research papers in International Journals.