Original paper

Soft Computing 8 (2003) 10–18 © Springer-Verlag 2003
DOI 10.1007/s00500-002-0241-3

Improved generalized neuron model for short-term load forecasting

D. K. Chaturvedi, M. Mohan, R. K. Singh, P. K. Kalra

Abstract The conventional neural networks consisting of simple neuron models have various drawbacks like large training time for complex problems, huge data requirement to train a non linear complex problems, unknown ANN structure, the relatively larger number of hidden nodes required, problem of local minima etc. To make the Artificial Neural Network more efficient and to overcome the above-mentioned problems the new improved generalized neuron model is proposed in this work. The proposed neuron models have both summation ($\sum$) and product (π) as aggregation function. The generalized neuron models have flexibility at both the aggregation and activation function level to cope with the non-linearity involved in the type of applications dealt with. The training and testing performance of these models have been compared for Short Term Load Forecasting Problem.

Keywords Generalized neural network, Back propagation, Load forecasting

1 Introduction

The reliable power supply available at reasonable cost is important for economic growth and development. The gap between generation and demand, forces the electricity boards to compromise on quality of the power supply by allowing voltage fluctuation/drops and shortage of electric power.

To bridge this shortage, load management is a mandatory requirement. In the process of planning effective load management strategies, load forecasting may be resorted to predict the load pattern in advance [1].

The short term forecast is needed for control, unit commitment, security assessment, optimum planning of power generation, planning for both spinning reserves and energy exchange, and also as inputs to load flow study or contingency analysis and many more decisions. There are several factors which affect the short term load forecasting such as meteorological, climatic, light intensity, price schemes, tariff structure and many others [3, 15].

A number of procedures exist for short term load forecasting techniques are :

- Multiple linear regressions.
- Stochastic time series.
- General exponential smoothing.
- State space and Kalman filter and
- Knowledge based approach.

These methods include non-whether sensitive and whether sensitive models. In India, sudden changes in climate seldom experienced. The whether load relationship is embedded in the load shape. Therefore, for the purpose of prediction, it is sufficient to use past load data only.
Of these five methods, time series technique is the most popular approach. It has been applied, and still being applied, to short-term load forecasting problems in electric power industry. The short-term load forecasting techniques may further be broadly categorized as follows:

1. Models independent of weather parameters.
2. Models including weather parameters.
3. Stochastic methods.

The main concern is to improve the accuracy of short-term load forecasting procedures. Forecasting has been mentioned as one of the most promising application areas of artificial neural network. Several authors have attempted to apply the back propagation-learning algorithm to train ANN for forecasting time series [2]. Neural networks have the remarkable ability to derive meaning from complicated or imprecise data.

Neural networks process information in a similar way as the human brain does. The network is composed of a large number of highly interconnected processing elements (neurons) working in parallel to solve a specific problem. Neural networks learn by example. They cannot be programmed to perform a specific task. The examples must be selected carefully otherwise useful time is wasted or even worse the network might be functioning incorrectly.

The existing neural networks have various drawbacks like large training time, huge data requirement to train for a non linear complex load forecasting problem, the relatively larger number of hidden nodes required etc.

2 Drawbacks of conventional ANN

The conventional neural network model suffers from serious drawbacks.

Published online: 14 July 2003

D. K. Chaturvedi (✉), M. Mohan, R. K. Singh
Department of Elect. Engineering, Faculty of Engineering,
Dayalbagh Educational Institute,
Dayalbagh, Agra (UP) 282 005, India
e-mail: dkc_foe@rediffmail.com

P. K. Kalra
Department of Electrical Engineering,
Indian Institute of Technology, Kanpur (UP) 208016, India
e-mail: kalra@iitk.ac.in

1. Training time for the conventional neural network is too large, which results in the slower response of the system.
2. Number of hidden layers and hidden neurons can't be predicted accurately, and also they are large in number for complex function approximation.
3. The existing neuron model performs only the operation of summation of its weighted inputs; it does not perform the operation of product on its weighted inputs.
4. There is a effect of threshold (activation) function on training time; also, accuracy of test results depends on threshold function.
5. Back propagation learning also has some shortcomings, like:
   i. Slow learning.
   ii. Problem of local minima may occur in the system.
6. There is a effect of normalization range of training data. Hence selection of suitable range (i.e. maximum and minimum values) is of great importance as it affects the results of the neural network training.
7. Training time of neural network depends on the mapping of input-output pattern (I/O-mapping) presented to the network.
8. Training time of the network also depends on the sequence of presentation of data.

To overcome the above drawbacks number of variants has been developed in the past decades. Most of the variants are either burden on the learning algorithms or/and increases the computational labor. In this paper, a new improved generalized neuron model is proposed, which overcome the drawbacks of conventional neural network by performing various possible variations and modification in the generalized neuron model to find the effect of activation function on the neural network output for the problem of short term load forecasting. The model should incorporate non-linearities present in the system.

The model should also incorporate following features:

1. The improved generalized neural network should consist of characteristics of simple neuron and also high order neuron characteristics.
2. There is no need of the selection of number of hidden layers and the number of neurons i.e. the complexity of the network should reduce.
3. The input output mapping should not affect the response of the network.
4. Normalizing effect should not be there.

3 Development of improvemented generalized neuron

Existing models of neuron in the structure of artificial neural network use the sigmoidal activation function and ordinary summation as aggregation functions. These models face problems in training when non-linearity involved in real life problems.

To deal with the above, the proposed models have both summation ($\sum$) and product (π) as aggregation function. The generalized neuron models have flexibility at both the aggregation and activation function level to cope with the non-linearity involved in the type of applications dealt with. The product and power non-linearity in problems made them complex for training, but with the help of product aggregation function it is quite easy to train. Author tried product neuron layers along with summation neuron layers in ANN and found that the training time is drastically reduced for mapping the non-linear starting characteristic of induction motor. Hence, in this paper, the different combinations of summation and product functions have been explored.

3.1 Generalized neuron model-1

The Generalized Neural network is developed on the basis of the Boolean algebra. It is the well known that with the help of Sum of Product and Product of Sum one can implement any given function [1]. Similarly in the generalized neuron structure summation and product as aggregation functions have been incorporated and the aggregated outputs pass through a non-linear squashing / thresholding function as shown in the Fig. 1.

$\sum$-part have the summation of weighted input with sigmoidal activation function f_1, while the Π-part have the product of weighted input with Gaussian activation function f_2. The final output of the neuron is a function of the weighted outputs O_Σ and O_π.

The output of summation ($\sum$) part of the generalized neuron is

$$O_\Sigma = f_1\left(\sum W_{\Sigma i}X\mathrm{i} + Xo_\Sigma\right)$$

The output of product (Π) part of the generalized neuron is

$$O_\pi = f_2(\pi W_{\pi i}X\mathrm{i} + Xo_\pi)$$

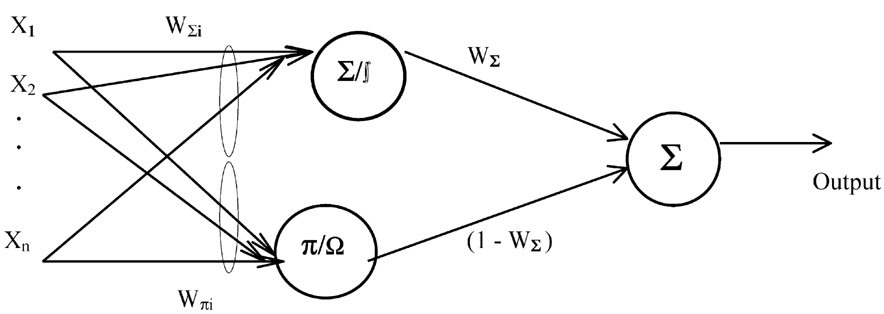


Fig. 1. Generalized Neuron Model-1

Finally, the outputs are summed up to get the neuron output. The output of the neuron can be mathematically written as

$$O_{\mathrm{i}} = O_{\Sigma} * W_{\Sigma} + O_{\pi} * (1 - W_{\Sigma})$$

3.2 Generalized neuron model-2

This model is similar to the above developed generalized neuron model-1. The only difference is that in this model the linear activation function is used. Here, f_1 and f_2 are straight line (ramp) function with slope unity.

3.3 Generalized neuron model-3

In earlier models of Generalized Neuron weighted input $(X\mathrm{i} * W\mathrm{i})$ is taken in $\sum$ and Π part. These types of neurons can produce single hyper plane or a line to separate the two classes of data. If we use $(X\mathrm{i} + W\mathrm{i})^{\mathrm{n}}$ instead of $X\mathrm{i} * W\mathrm{i}$ then a closed surface can be generated depending of the value of n. If n is equal to 2, then elliptical shape may be obtained. Hence it will be quick and easy to train non-separable and non-linear problems. The value of n may also be changed in training make it generalized.

In this model summation $(\sum)$ part have summation of $(X + W)^2$ with the sigmoidal activation function f_1 while the product (Π) part consisting of product of weighted input have the Gaussian activation function f_2. The final output of the neuron is a function of the two outputs O_{Σ} and O_{π} with the weights $W_{\Sigma\mathrm{i}}$ and $W_{\pi\mathrm{i}}$ respectively. The output of summation $(\sum)$ part of the generalized neuron is

$$O_{\Sigma} = f_1\left(\sum (W_{\Sigma\mathrm{i}} + X\mathrm{i})^2 + Xo_{\Sigma}\right)$$

The output of product (Π) part of the generalized neuron is

$$O_{\pi} = f_2(\pi W_{\pi\mathrm{i}} * X\mathrm{i} + Xo_{\pi})$$

Finally, the outputs are summed up to get the neuron output. The output of the neuron can be mathematically written

$$O\mathrm{i} = O_{\Sigma} * W_{\Sigma} + O_{\pi} * (1 - W_{\Sigma})$$

3.4 Generalized neuron model-4

This model is similar to the above developed generalized neuron model-3. The only difference is that in this model the linear activation function is used. Here, f_1 and f_2 are straight-line (ramp) function with slope unity.

3.5 Generalized neuron model-5

In this model summation $(\sum)$ part have summation of $(X + W)^2$ with sigmoidal activation function f_1, while the product (Π) part consisting of product of $(X + W)$ with gaussian activation function f_2. The final output of the neuron is a function of the two outputs O_{Σ} and O_{π} with the weights $W_{\Sigma\mathrm{i}}$ and $W_{\pi\mathrm{i}}(= 1 - W_{\Sigma})$ respectively.

3.6 Generalized neuron model-6

This model is similar to the above developed generalized neuron model-5. The only difference is that in this model the linear activation function is used. Here, f_1 and f_2 are straight line (ramp) function with slope unity.

3.7 Generalized neuron model-7

In this model summation $(\sum)$ part consisting of summation of $(X + W)$ with sigmoidal the activation function f_1, while the product (Π) part have product of $(X + W)$ Gaussian activation function f_2. The final output of the neuron is a function of the two outputs O_{Σ} and O_{π} with the weights $W_{\Sigma\mathrm{i}}$ and $W_{\pi\mathrm{i}}$ respectively.

3.8 Generalized neuron model-8

In this model summation $(\sum)$ part and the product (Π) part both have $(X + W)^2$ as input with sigmoidal and Gaussian functions respectively. The final output of the neuron is a function of the two outputs O_{Σ} and O_{π} with the weights $W_{\Sigma\mathrm{i}}$ and $W_{\pi\mathrm{i}}$ respectively.

3.9 Generalized neuron model-9

This model is similar to the above developed generalized neuron model-8. The only difference is that in this model the linear activation function is used. Here, f_1 and f_2 are straight line (ramp) function with slope unity.

4 Learning algorithm of generalized neuron model-1 using back propagation

The following step are involved in the training of Generalized Neural Network –

4.1 Modification of W_{Σ}

After calculating the output of generalized neuron in the forward pass of feed forward back propagation neural networks, it is compared with the desired output to find the error and then it is minimized to train the Generalized Neural Network (GNN) model. Hence in this step the output of the GNN with a single flexible generalized neuron model is to be compared with the desired output to get error for ith set of input.

$$\text{Error } E_{\mathrm{p}} = (D_{\mathrm{i}} - O_{\mathrm{i}})$$

The error that is minimized by the generalized delta rule is the sum of squares of the errors for all output units, a multiplication factor of 0.5 has been taken for simplifying the calculations:

$$E_{\mathrm{p}} = 1/2 \sum (D_{\mathrm{i}} - O_{\mathrm{i}})^2$$

$$-\partial E_{\mathrm{p}}/\partial W_{\Sigma} = \sum (D_{\mathrm{i}} - O_{\mathrm{i}})(O_{\Sigma\mathrm{i}} - O_{\pi\mathrm{i}})$$

$$\Delta W_{\Sigma} = \eta \sum (D_{\mathrm{i}} - O_{\mathrm{i}})(O_{\Sigma\mathrm{i}} - O_{\pi\mathrm{i}}) + \alpha \Delta W_{\Sigma}$$

$$W_{\Sigma} = \Delta W_{\Sigma} + W_{\Sigma}$$

4.2
Modification of $W_{\Sigma i}$

$$\partial E_p/\partial W_{\Sigma i} = \sum (D_i - O_i) W_\Sigma O_\Sigma (1 - O_\Sigma) \cdot Xi$$
$$\Delta W_{\Sigma i} = \eta \sum (D_i - O_i) O_{\Sigma i}(1 - O_{\Sigma i}) \cdot W_\Sigma \cdot Xi + \alpha \Delta W_{\Sigma i}$$
$$W_{\Sigma i} = \Delta W_{\Sigma i} + W_{\Sigma i}$$

4.3
Modification of $W_{\pi i}$

$$\partial E_p/\partial W_{\pi i} = -2 \sum (D_i - O_i) \times (1 - W_\Sigma) \cdot O\pi \cdot (\text{net\_pi}) \cdot (\text{net\_pi}/W_{\pi i})$$
$$\Delta W_{\pi i} = -2 \cdot \eta \cdot (1 - W_\Sigma) \times \sum (D_i - O_i) \cdot O\pi \cdot (\text{net\_pi})(\text{net\_pi})/W_{\pi i} + \alpha \Delta W_\pi$$
$$W_{\pi i} = \Delta W_{\pi i} + W_{\pi i}$$

Table 1. Training File

0.2690	0.1389	0.1461	0.1000	0.2201
0.1883	0.1768	0.1461	0.1068	0.1745
0.1526	0.1676	0.1154	0.1192	0.1248
0.1000	0.1000	0.1398	0.1733	0.1580
0.1019	0.1891	0.1000	0.1372	0.1000
0.2512	0.3622	0.2600	0.3133	0.3567
0.5845	0.6982	0.5178	0.7172	0.6320
0.8117	0.8068	0.4431	0.8425	0.6889
0.8343	0.8580	0.6142	0.8312	0.7965
0.8624	0.8283	0.7484	0.8300	0.8617
0.8709	0.7720	0.7603	0.7804	0.8700
0.7657	0.6460	0.6743	0.7375	0.7375
0.5620	0.4544	0.5576	0.5310	0.4829
0.7000	0.4636	0.6610	0.7398	0.5067
0.7150	0.5446	0.7016	0.7262	0.6019
0.7310	0.6091	0.7672	0.7702	0.6692
0.7451	0.6265	0.7421	0.7003	0.6485
0.8042	0.6449	0.7400	0.7567	0.7799
0.9000	0.8416	0.8420	0.8571	0.7965
0.8606	0.9000	0.9000	0.9000	0.8876
0.8192	0.7709	0.8497	0.8898	0.9000
0.6277	0.6306	0.7728	0.6924	0.7189
0.4962	0.5425	0.6890	0.6958	0.7220
0.3488	0.4370	0.5493	0.4870	0.5305

Table 2. Training performance for different neuron models for short term load forecasting problem

Learning rate = 0.05; Momentum = 0.65; Epochs = 200	
Model	Sum-Squared Error
Model 1	0.0420
Model 2	0.0268
Model 3	0.0322
Model 4	0.0495*
Model 5	0.0329
Model 6	Not trained
Model 7	0.0371
Model 8	0.0328
Model 9	Not trained

* Model 4 is not trained at the learning rate=0.05, but is capable to perform training at the value below 0.01, the result shown is at the value of 0.005

4.4
Modification of $X_{O\Sigma}$

$$-\partial E_p/\partial Xo_\Sigma = \sum (D_i - O_i) W_\Sigma O_\Sigma (1 - O_\Sigma).$$
$$\Delta Xo_\Sigma = \eta \sum (D_i - O_i) O_{\Sigma i}(1 - O_{\Sigma i}) \cdot W_\Sigma + \alpha W_{\Sigma i}$$
$$Xo_{\Sigma i} = \Delta Xo_{\Sigma i} + Xo_{\Sigma i}$$

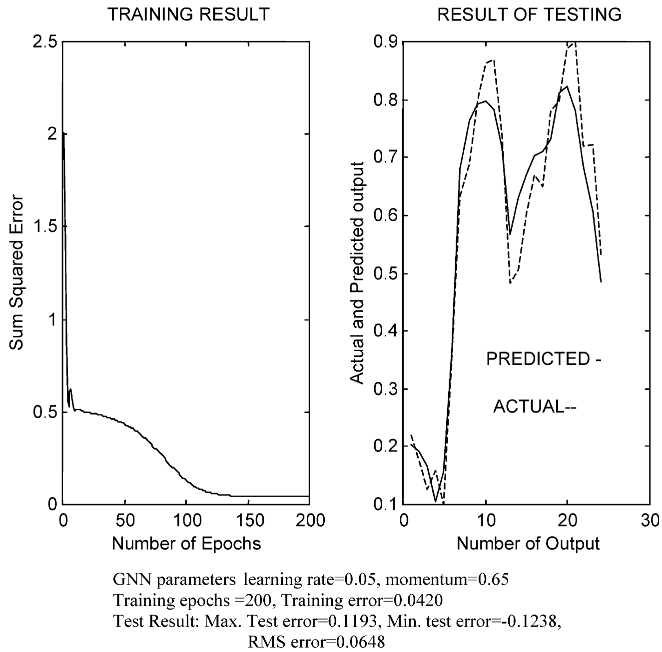


Fig. 2. Results of model-1

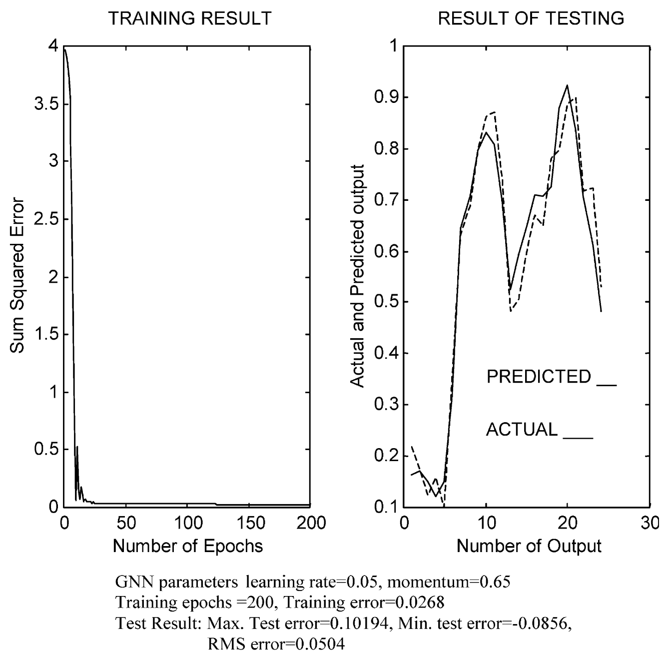


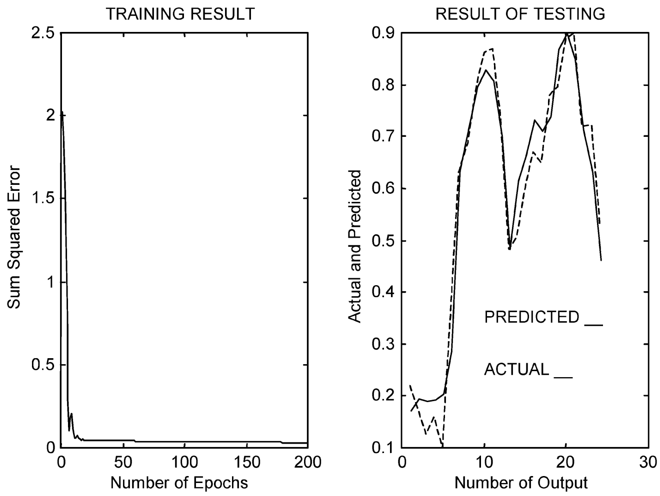
Fig. 3. Results of model-2

4.5 Modification of $X_{o\pi}$

$$\partial E_p/\partial X_{o\pi} = -2\sum(D_i - O_i)(1 - W_\Sigma)\cdot O\pi\cdot(\text{net\_pi})$$

$$\Delta X_{o\pi} = -2\cdot\eta\cdot(1 - W_\Sigma)\sum(D - O_i)\cdot O\pi\cdot(\text{net\_pi}) + \alpha\Delta X_{o\pi}$$

$$X_{o\pi} = \Delta X_{o\pi} + X_{o\pi}$$



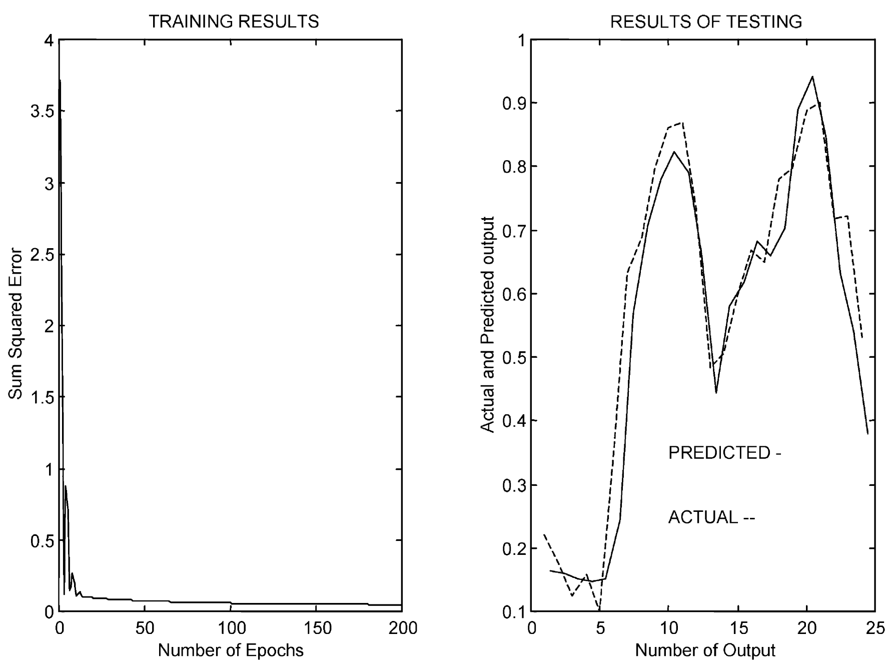
GNN parameters learning rate=0.05, momentum=0.65
Training epochs =200, Training error=0.0322
Test Result: Max. Test error=0.0918, Min. test error=-0.174,
RMS error=0.0566

Fig. 4. Results of model-3

5 Electrical short-term load forecasting using improved generalized neuron models

For the purpose of optimal planning and operation of the electric power system, the active power demands at various load buses need to be estimated ahead of time. Therefore, load forecasting plays an important role in power system operation and control.

Rajurkar and Nissen [3] introduced stochastic modeling and analysis methodology called data-dependent systems (DDS) for Short Time Load Forecasting (STLF), while Goh and Ong [8] refined the approach through stochastic time series analysis so that with routinely available data from a number of key substations, the substation demand patterns are separately characterized. Hwang and Moon [5] discussed a power load forecasting system based on a temporal difference (TD) method. Dillon [6], Ishibashi [12], and Matsumoto [8] presented method of short term load forecasting using Artificial Neural Networks. Azzam-ul-Asar et al. [13] in 1992 the effectiveness of an ANN approach to short-term load forecasting in power systems was investigated. Examples demonstrate the learning ability of an ANN in predicting the peak load of the day by using different preprocessing approaches and by exploiting different input patterns to observe the possible correlation between historical load and temperatures. In 1993, Guangxi et al. [7] presented a method of changing a topological ANN to forecast the load of a power system. The model is almost an all-round reflection of various factors, which affect the changing of load. Papalexopoulos [17, 18] presented an ANN based model for the calculation of next day's load forecasts. The most significant aspects of the model fall into the following two areas: training process



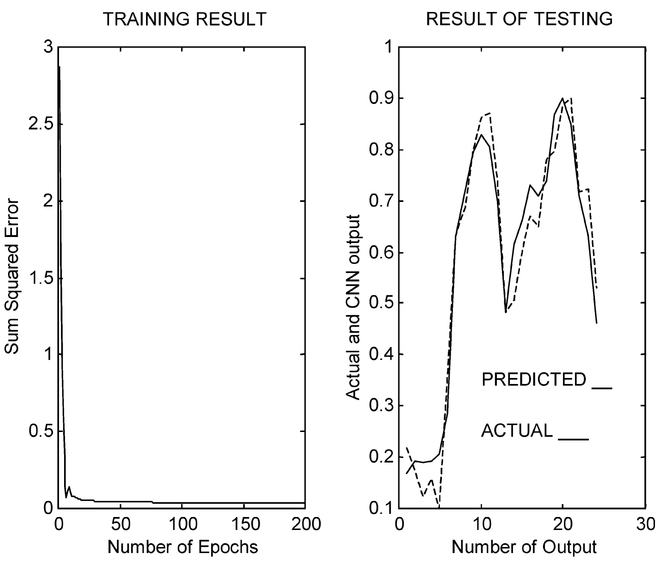
GNN parameters learning rate=0.05, momentum=0.65
Training epochs =200, Training error=0.0495
Test Result: Max. Test error=0.1596, Min. test error=-0.1149,
RMS error=0.0677

Fig. 5. Results of model-4

and selection of the input variables. At the same time Lee et al. [11] presented a diagonal recurrent ANN with an adaptive learning rate. In 1993, Chaudhary et al. [4] presented a fast and accurate method of STLF using combinations of self-organising maps (SOM) and multi-layer perception model. The SOM recognizes the type of day examining the variation of load which, along with past load, temperature, humidity etc. Peng et al. [14] used a linear adaptive neuron or adaptive linear combiner called Adaline for STLF. Hence, it is very clear that ANN is gaining momentum in load forecasting due to various reasons like ability to cope up with non-linearity, adaptivity, intelligent and simplicity. Chaturvedi et al. [1] used the generalized neural network (GNN) approach for electrical STLF Problem to overcome the problem of ANN. The performance of GNN has been again improved using $(Xi + Wi)^n$ instead of weighted input $(Xi * Wi)$, so that a closed surface may be generated depending on the requirement, by selecting the proper value of n.

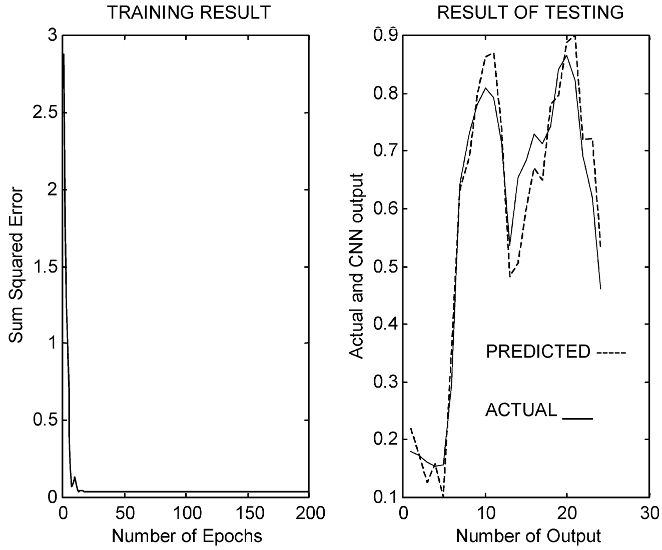
The short-term demand of Gujarat Electricity Board has been collected and arranged in proper normalized format. The four past histories of electrical demand have been used as four inputs to the neural network models and present demand as the output of the models as shown in Table 1.

The training performance of various models have been compared in terms of sum-squared error achieved in 200 epochs as given in the Table 2. Also the trajectories of sum-squared error with the epochs are plotted for different models and given in the Figs. 2–8. In terms of training, the model-2 provides the least sum-squared error, i.e. 0.0268 in 200 epochs. Models-3, 5 and 8 are also able to train upto a tolerable error level. Model-4 is unable to train at the value of learning rate 0.05, but it is trained efficiently at the learning rate of 0.005. Models-6 and 9 are not trained at all.



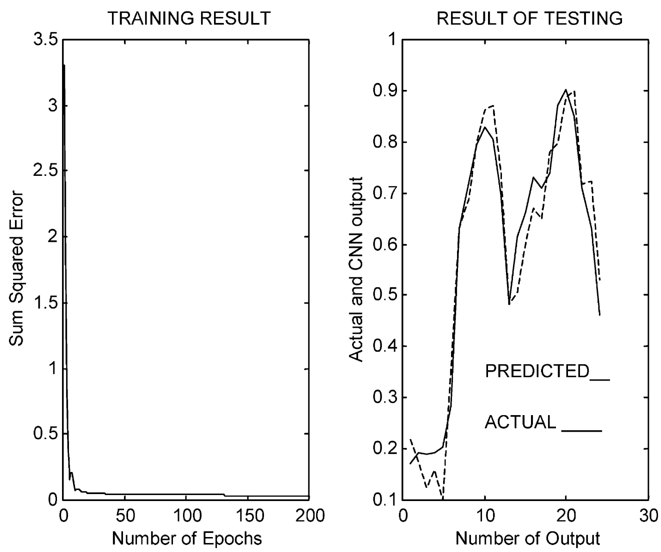
GNN parameters learning rate=0.05, momentum=0.65
Training epochs =200, Training error=0.0329
Test Result: Max. Test error=0.0919, Min. test error=-0.1093, RMS error=0.0572

Fig. 6. Results of model-5



GNN parameters learning rate=0.05, momentum=0.65
Training epochs =200, Training error=0.0371
Test Result: Max. Test error=0.1025, Min. test error=-0.1477, RMS error=0.0601

Fig. 7. Results of model-7



GNN parameters : learning rate=0.05, momentum=0.65
Training epochs =200, Training error=0.0328
Test Result: Max. Test error=0.0, Min. test error=-0.1088, RMS error=0.0571

Fig. 8. Results of model-8

Table 3. The performance of different models while testing for short term load forecating problems

Model	RMS error	Min. error	Max. error
Model 1	0.0648	0.1238	0.1193
Model 2	0.0504	−0.0856	0.10194
Model 3	0.0566	−0.174	0.0918
Model 4	0.0677	−0.1149	0.1596
Model 5	0.0572	−0.1093	0.0919
Model 6	–	–	–
Model 7	0.0601	−0.1477	0.1025
Model 8	0.0571	−0.1088	0.0926
Model 9	–	–	–

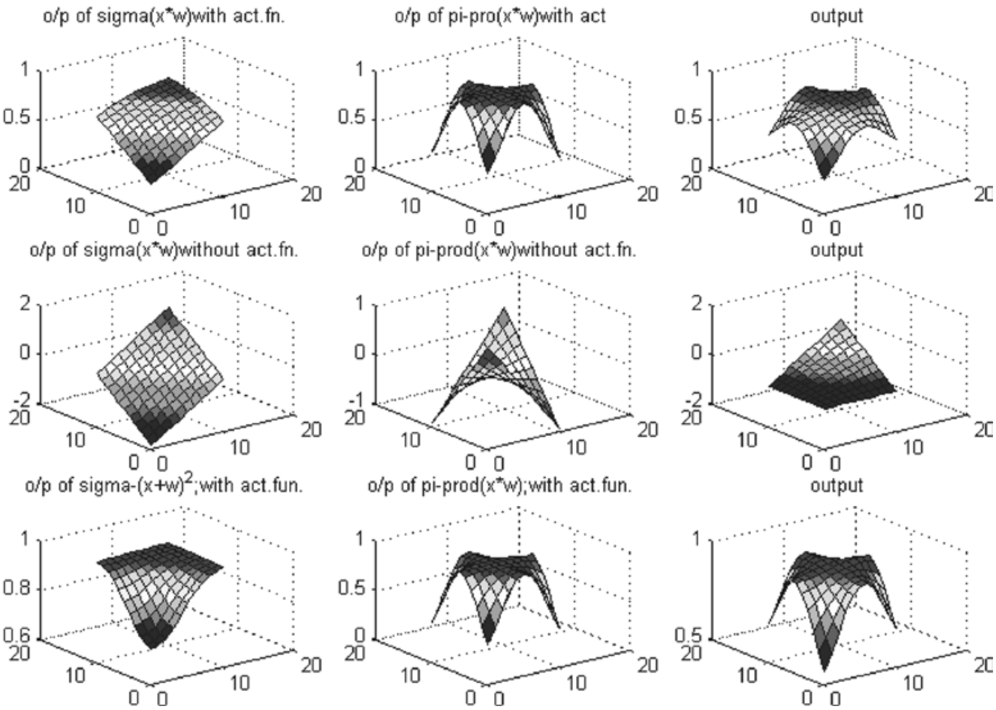


Fig. 9. Three Dimensional surfaces of neuron models-1, 2 and 3

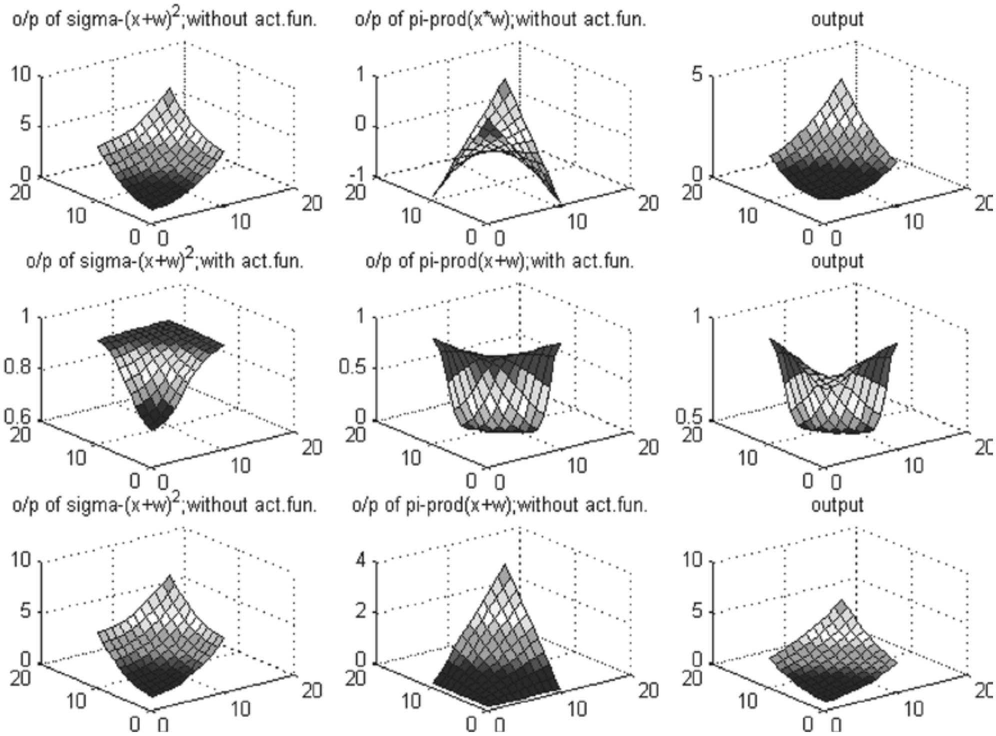


Fig. 10. Three Dimensional surfaces of outputs neuron models-4, 5 and 6

To compare the testing performance of generalized neural network models the actual and predicted output are plotted as shown in Figs. 2–8 and the RMS error, minimum and maximum errors for forecasting are given in Table 3. From this table, it is very clear that the neuron model-2 gives the least rms error while testing i.e. 0.0504. Models-3 and 5 also give the comparable rms error, which is 0.0566 and 0.0572.

The results obtained for short-term load forecasting problem from the generalized neural network using different neuron models have been compared with the actual results and it is found that the improved Generalized Neural Network Model-2 is most efficient in training and accurate in prediction. Models-3 and 5 also provides efficient results in terms of both training and testing.

The three-dimensional surfaces are plotted for the different models of improved generalized neuron model. 3-D Surface of output of summation ($\sum$) part of a models, surface of output of product (Π) part of the models and finally the surface of the complete output of the General-

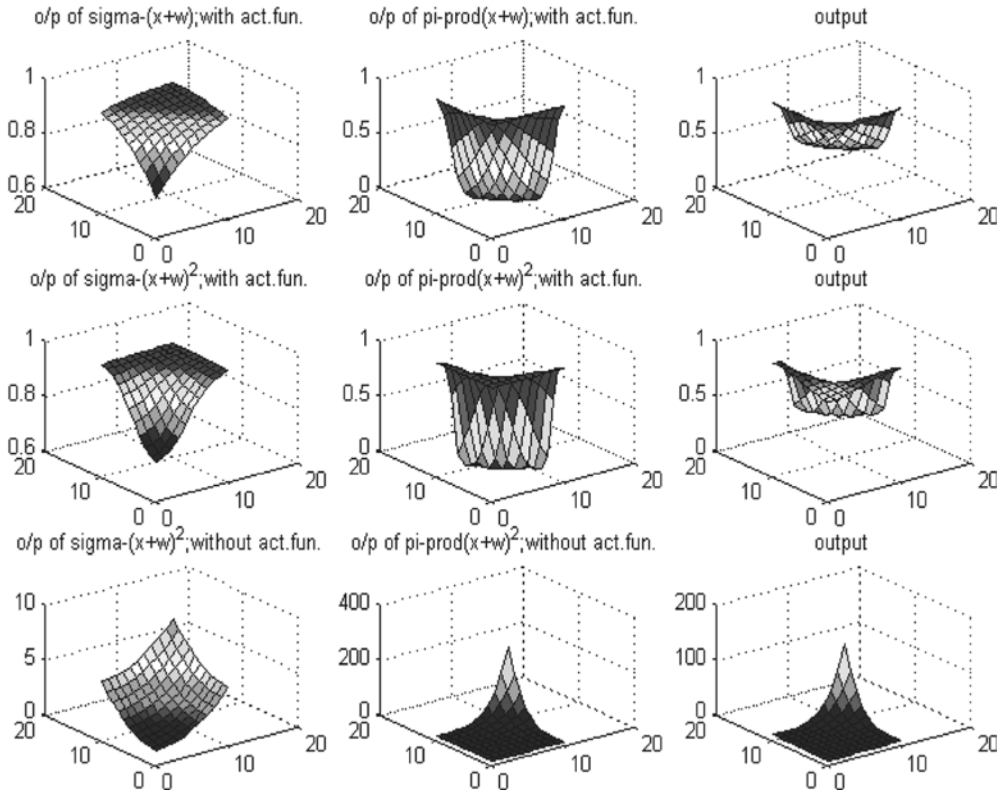


Fig. 11. Three Dimensional surfaces of output of neuron models-7, 8 and 9

ized Neuron Models are shown in Figs. 9–11. From these three-dimensional surfaces of output of different models it is very clear that the model-3 and model-5 can cope up the non- separable problems very easily and efficiently without involving the network complexity. Conventional single layer artificial neural networks cannot cope up with non-separable problems but the multiplayer network can do that, but it requires large training time and network complexity is also not less.

The output surfaces obtained for model-7 and model-8, can also cope up with the non-linearities, but the surfaces are not in the complete range 0–1. Therefore, training time for the model may be large.

6 Conclusion

The improved generalized neuron model has been developed in the present work to overcome the problems of ANN. For this purpose various modifications in the generalized neuron have been discussed. The improved generalized neuron models have both summation ($\sum$) and product (π) as aggregation function with and without activation function.

The three-dimensional surfaces are plotted for the different models of improved generalized neuron models to determine the ability of the networks to cope up with the non-linearities present in the systems.

Electrical Short-Term Load Forecasting Problem had been formulated in the framework of improved generalized neural network models using back propagation learning algorithm. Forecasting results as the predicted load are obtained for the developed models and the following observations are made:

1. In terms of training, the model-2 provides the least training error (0.0268) as shown in the Table 2. Models 3, 5 and 8 are also giving good results.
2. For short-term load forecasting problem improved generalized neuron model-2 gives minimum RMS testing error (i.e. 0.0504). Models 3, 5 and 8 also good from testing point of view as shown in Table 3.
3. From the three-dimensional surfaces of output of different improved generalized neuron models it is very clear that the model-3 and model-5 can cope up the non-separable problems very easily and efficiently without involving the network complexity.
4. The output surface of model-5 may be a closed surface very easily, which can cope up with the non-linear separable problems easily. While conventional artificial neural network can do that with multi-layers and large training time i.e. with complex network structure.

References

1. **Chaturvedi DK, Satsangi PS, Kalra PK** (1997) Short term load forecasting using generalized neural network (GNN) approach. Journal of The Institute of Engineers (India) **78:** 83–87
2. **Rumelhart DE, Hilton GE, Willams RJ** (1986) Learning internal representation by error propagation. Parallel Distributed Processing, Vol **1** 318–362 Cambridge: MIT Press
3. **Rajurkar KP, Nissen JL** (1985) Data-dependent system approach to short-term load forecasting. IEEE Trans. on Systems, Man & Cybernetics, Vol. SMC-15, No. 4
4. **Choudhary SD, Kalra PK, Srivastava SC, Kumar DMV** (1993) Short-Term Load Forecasting using Artificial Neural Network. ESAP

5. **Hwang J-N, Moon S** (1991) Temporal difference method for multi-step prediction: application to power load forecasting, Proceedings of the First International Forum on Applications of Neural Networks to Power Systems. pp. 41–45, IEEE, NY, USA
6. **Dillon TS, Sestito S, Leung S** (1991) An adaptive neural network approach in load forecasting in a power system, Proceedings of the First International Forum on Applications of Neural Networks to Power Systems. pp. 17–21, IEEE New York, NY, USA
7. **Guangxi L, Manli X** (1993) Changing topological artificial neural network for load forecasting. Automation of Electric Power Systems, **17**(11): 28–33
8. **Matsumoto T, Kitamura S, Ueki Y, Matsui T** (1993) Short-term load forecasting by artificial neural networks using individual and collective data of preceding years, ANNPS '93. Proceedings of the Second International Forum on Applications of Neural Networks to Power Systems. pp. 245–50, IEEE New York, NY, USA
9. **Papalexopoulos AD, Hao S, Peng T-M** (1993) Short-term system load forecasting using an artificial neural network, ANNPS '93. Proceedings of the Second International Forum on Applications of Neural Networks to Power Systems. pp. 239–44, IEEE New York, NY, USA
10. **Papalexopoulos AD, Shangyou H, Peng T-M** (1994) An implementation of a neural network based load forecasting model for the EMS. IEEE Transactions on Power Systems. **9**(4): 1956–62
11. **Lee KY, Choi TI, Ku CC, Park JH** (1993) Short-term load forecasting using diagonal recurrent neural network, ANNPS '93. Proceedings of the Second International Forum on Applications of Neural Networks to Power Systems. pp. 227–32, IEEE News York, NY, USA
12. **Ishibashi K, Komura T, Ueki Y, Nakanishi Y, Matsui T** (1992) Short-term load forecasting using an artificial neural network, ICARCV '92. Second International Conference on Automation, Robotics and Computer Vision. pp. INV 11.1/1–5 vol. 3, Nanyang Technol. Univ Singapore
13. **Azzam-ul-Asar JR, McDonald JR, Khan MI** (1992) A specification of neural network applications in the load forecasting problem, First IEEE Conference on Control Applications. pp. 577–82 Vol. l, IEEE New York, NY, USA
14. **Peng TM, Hubele NF, Karady GG** (1993) An adaptive neural network approach to one-week ahead load forecasting, IEEE Transactions on Power Systems. **8**(3): 1195–203
15. **Hippert HS, Pedreira CE, Souza RC** (2001) Neural networks for short-term load forecasting: a review and evaluation. IEEE Transaction on Power System **16**(1): 44–55