Neural Network Models for Air Quality Prediction: A Comparative Study

S V Barai [A1], A K Dikshit [A2], Sameer Sharma[A3]

[A1]Department of Civil Engineering, Indian Institute of Technology Kharagpur 721 30, India,
[A2]Center for Environmental Science and Engineering, Indian Institute of Technology Bombay, 400 076, India,
[A3]TCE Consulting Engineers Ltd., Mumbai 400 009, India

Abstract: The present paper aims to find neural network based air quality predictors, which can work with limited number of data sets and are robust enough to handle data with noise and errors. A number of available variations of neural network models such as Recurrent Network Model (RNM), Change Point detection Model with RNM (CPDM), Sequential Network Construction Model (SNCM), and Self Organizing Feature Maps (SOFM) are implemented for predicting air quality. Developed models are applied to simulate and forecast based on the long-term (annual) and short-term (daily) data. The models, in general, could predict air quality patterns with modest accuracy. However, SOFM model performed extremely well in comparison to other models for predicting long-term (annual) data as well as short-term (daily) data.

Key Words: Air Quality, Change Point Detection, Recurrent Neural Networks, Self Organizing Feature Maps

1 Introduction

Air pollutants exert a wide range of impacts on biological, physical, and economic systems. Their effects on human health are of particular concern. The decrease in respiratory efficiency and impaired capability to transport oxygen through the blood caused by a high concentration of air pollutants may be hazardous to those having pre-existing respiratory and coronary artery disease (Rao and Rao, 2000). Consequently, it has become a vital task to accurately keep track of the variation of ambient air pollution levels in urban areas.

Natural phenomena are mostly a time series with some degree of randomness. Pollutants in the atmosphere may disperse or concentrate during varied time periods. Previous studies (Giorgio and Piero, 1996) have indicated that the data of ambient air quality are stochastic time series, thereby making it possible to make a short-term forecast on the basis of historical data. Though models may be imperfect, they are the best tool for use in all aspect of air quality planning where prediction is a major component such as for emission control (Melas et al., 2000), accidental release of pollutant, land-use planning, traffic planning (Hadjiiski and Hopke, 2000), planning of measurement programs (Rao and Rao, 2000), analyses of measurements/ trends and episode forecasting (Melas et al., 2000).

Within the class of statistical methods until now, either the time-series methods, which do not use meteorological inputs, or regression and similar methods, which are mostly based on multivariate linear relationship between meteorological conditions and ambient air pollution concentrations, were commonly used. However, when applying the conventional time-series models to the ambient air pollution forecast, the pollutant level variations are generally not simple autoregressive or moving average models. Analyst must employ statistical graphs of the autocorrelation function and partial autocorrelation function to identify an appropriate time-series model (Chakraborty et al., 1992). In the model identification stage, the resulting model quality frequently relies on individual experience and knowledge of time-series statistic. Furthermore, a time-series model may not be applicable for varied periods of data. A model applicable in one period may require manual adjusting of its model parameters to meet the data characteristics in other time periods. These complexities make applying a time series model to regular air quality forecast an inefficient task. In other words, though the statistical methods do provide reasonable results, these are essentially incapable of capturing complexity and non-linearity of pollution-weather relationships.

The neural networks (Principe and Kuo, 1995) have emerged out to be more flexible, less assumption dependent and adaptive methodology in environment related areas such as rainfall runoff modeling, stream flow forecasting (Thirumalaiah and Deo, 1998), ground water modeling, water management policy, precipitation forecasting , hydrologic forecasting and reservoir operation (Thirumalaiah and Deo, 2000), lake and reservoir modeling (ASCE, 2000a, 2000b), remote sensing and GIS related activities, real time control of water treatment plants, water quality and air quality management (Boznar et al., 1993), adsorbent beds design (Basheer and Najjar, 1996), and hazardous waste management.

The present study investigates the advantage of using neural networks for forecasting the air pollution. The aim is to find better air quality predictors, which can work with low number of data sets and should be robust enough to handle data with noise and errors. The objectives of the study are as follows:

- To implement various available variations of neural network models for predicting air quality
- To collect suitable data sets for multiple air quality parameters - one containing yearly average pollutant concentrations at a specific location and other containing daily average pollutant concentrations record for significantly long duration.
- To conduct exhaustive simulations using above-developed models with yearly data and hourly data to assess the relative advantage of each model in prediction
- To perform comparative study to identify suitable air quality prediction model(s) for yearly (long-term) data and that for daily (short-term) data

2 Implementation of Various Neural Networks Models

This section provides details about selected a few neural networks models applied for air quality prediction.

2.1 Recurrent Network Model (RNM)

For a neural network to be dynamic, it must be given memory. Memory may be divided into "short-term" and "long-term" memory depending upon the retention time. Long-term memory is built into a neural network through supervised learning, whereby the information content of the training data set is stored in the synaptic weights of the networks (Haykin, 2000). However, if the task at hand has a temporal dimension, we need some form of short-term memory to make the network dynamic. The static network accounts for non-linearity and the memory accounts for time. Short-term memory can be implemented in continuous time or in discrete time. Such networks typically use a variant of back-propagation for training. Essentially, there are three ways that a "memory" can be introduced into static neural networks (Connor et al., 1994; Parlos et al., 2000). These are (in increasing order of complexity and capability):

- Tapped Delay Lines Models**:** In these models, the network has past inputs explicitly available (through a tapped delay line) to determine its response at a given point in time. Thus, the temporal pattern is converted to a spatial, which can then be learned through, say, classic back propagation (Haykin, 2000).

- Context Model or Partial Recurrent Models: These models retain the past output of nodes instead of retaining the past raw inputs. For example, the output of the hidden layer neurons of a feed forward network can be used as inputs to the network along with true inputs. These “network derived” inputs are also called context inputs. When the interconnections carrying the context inputs are fixed, classical back propagation can be used for training the network.
- Fully Recurrent Models: These models employ full feedback and interconnection between all nodes (Haykin, 2000). Algorithms to train fully recurrent models are significantly more complex in terms of time and storage requirements.

For the present study, the partial recurrent model has been used.

2.2 Change Point Detection Model (CPDM)

In general, air quality parameters are controlled by various other factors such as emission rate from vehicle and industries etc. This emission change with the introduction or removal of new vehicles, industries and change in atmospheric condition etc. Therefore, we can conjecture that the movement of air quality parameters has a series of change points (Kyong and Han, 2000), which occur because of these changes. The proposed model consists of three stages. The first stage is to detect successive change points in the air quality patterns over a number of years called the *change point detection (CPD) stage*. The second stage is to forecast the change-point group with say back propagation referred as the *change-point-assisted group detection (CPGD) stage*. The final stage is to forecast the output with say back propagation and it is referred as the *output forecasting neural network (OFNN) stage*. The back propagation model is used as a classification tool in CPGD and as a forecasting tool in OFNN. This model obtains intervals divided by change points in the training phase, identifies them as a change point groups in the training phase, and forecast to which group each sample is assigned in the testing phase. In this a series of change point will be detected by the Pettitt test, a nonparametric change-point detection method, as nonparametric statistical property is a suitable match for neural network model that is a kind of nonparametric method (Pettitt, 1979). For the present study, partial recurrent network model has been used in place of classical back propagation model.

2.3 Sequential Network Construction Model (SNCM)

This model introduces an application of the Sequential Network Construction (SCN) to select the size of several popular neural network predictor architectures for various benchmark-training sets. The specific architecture considered here consists of a Finite Impulse Response (FIR) network and the partial recurrent Elman network for adding context units to the output layer (Back and Tsoi, 1991). This model considers an enhancement of a FIR network in which only the weights having relevant time delays are utilized. Bias-variance trade off in relation to the prediction risk estimation by means of Nonlinear Cross Validation is discussed elsewhere (Tomasz and Zurada, 1977).

2.4 Self Organizing Feature Maps (SOFM) Model

These networks are based on competitive learning i.e. the output neurons of the network compete among themselves to be activated or fired, with the result that only one output neuron or one neuron per group is on at any one time (Haykin, 2000). An output neuron that wins the competition is called a winning neuron. One way of inducing a winner-takes-all competition among the output neurons is to use inhibitory connections (negative feedback paths) between them.

In a self organizing map, the neurons are placed at the nodes of a lattice that is usually one- or two-dimensional. The neurons become selectively tuned to various input patterns or a classes of input patterns in the course of a competitive learning. The locations of the neurons so tuned become ordered with respect to each other in such way that a meaningful coordinate system for different input features is created over the lattice. Hence neurons in the lattice are indicative of intrinsic statistical features contained in the input patterns. The spatial location of an output neuron in a topographic map corresponds to a particular domain or feature of data drawn from input space (Kohonen, 1990).

The principal goal of the self organizing map is to transform an incoming signal pattern of arbitrary dimension into a one- or two-dimensional discrete map, and to perform this transformation adaptively in a topologically ordered fashion. Each neuron in the lattice is fully connected to all the source nodes in the input layer. This network represent a feed forward structure with a single computational layer consisting of neurons arranged in rows and columns. The algorithm responsible for the formation of the self organizing map proceeds first by initializing the synaptic weights in the network. This can be done by assigning them small values picked from a random number generator. Once the network has been properly initialized, there are three essential process involved in the formation of the self organizing map, explained elsewhere (Principe and Wang, 1993; Haykin, 2000).

All above-mentioned models were implemented in MATLAB software (Math works, 2000) using Neural Networks Tool box (Demuth and Beale, 1992). More details can be found elsewhere (Sharma, 2002).

3 Data Collection and Properties

Most air quality data are obtained from air quality monitoring stations directly or through remote sensing instruments. Also, an existing model or laboratory experiment can be used to generate data patterns for specific applications. Again there appears to be no fixed method for determining the number of input-output data pairs that will be required. To ensure a good approximation, the number of data pairs used for training should be equal to or greater than the number of parameters (weights) in the network (ASCE, 2000a). For the present study, two types of data sets - one for annual average pollutant concentrations and another for daily average pollutant concentrations have been arranged to apply the model for long-term as well as short-term predictions. The details on data collection, database construction etc. have been discussed in following sections.

3.1 Case Study 1

Data for 115 counties of California State in United States of America has been collected from US EPA website (www.epa.gov). This data is annual average data for 15 years from 1985 to 1999 for seven parameters namely VOC (volatile organic carbon), NO_X (oxides of nitrogen), CO (carbon monoxide), SO_2 (sulphur dioxide), PM10 (particulate matter with size less than 10 microns), PM2.5 (particulate matter with size less than 2.5 microns) and NH_3 (ammonia). All concentrations are in micrograms per cubic meter.

3.2 Case Study 2

The data for three parameters namely RPMA (Respiratory Particulate Matter Average), SO_2 (sulphur dioxide) and NO_2 (nitrogen dioxide) is collected for Delhi State at nine locations. These data are daily average concentrations for last two years from 3/7/2000 to 20/8/2001. This data set has been collected from Tata Energy Research Institute web site www.teri.in. However, only the data for Ashram Chowk has been used for carrying out simulation studies. The dataset size had 110 patterns.

3.3 Statistical Properties of Data Sets

The statistical properties of the data sets are very important for data analysis and pre-processing. They are mean, standard deviation, variance, hypothesis tests etc. Here some properties namely mean, standard deviation, median of the data of two case studies are give below in Tables1 and 2, respectively.

Table. 1. Statistical properties of data set – Case Study 1

Parameter	Property		
	Mean	Std. Dev.	Median
VOC	69.91	23.09	66.73
NO_X	3662.1	1762.4	2912
CO	675.69	249.43	539.13
SO_2	551.66	451.46	273.38
PM10	150.07	118.63	78.39
PM2.5	33.21	25.71	40.15
NH3	13.34	16.98	0.08

Table. 2. Statistical properties of data set - Case Study 2

Parameter	Property		
	Mean	Std. Dev.	Median
RPMA	172.83	119.88	141
SO_2	9.59	3.7	9
NO_2	77.18	31.59	72

3.4 Preprocessing

The data is processed before used as input to the network. Data can be rescaled/normalized/standardized according to the requirement and properties of the data sets (Warren, 2002). *Rescaling* a vector means to add or subtract a constant and then multiply or divide by a constant. *Normalizing* a vector most often means dividing by a norm of the vector, for example, to make the Euclidean length of the vector equal to one. In the neural network literature, normalizing also often refers to rescaling by the minimum and range of the vector to make all the elements lie between 0 and 1. *Standardizing* a vector means subtracting a measure of location and dividing by a measure of scale. For example, if the vector contains random values with a Gaussian distribution, one might subtract the mean and divide by the standard deviation, thereby obtaining a 'standard normal' random variable with mean 0 and standard deviation 1.

The data is normalized in three ways. In first method, the whole data set is divided by the maximum value of that parameter. In second method, a normalized value is calculated $N = (w_{max} - w_{present})/(w_{max} - w_{min})$ and then this N is used to renormalize the output value. In third method, all values are mapped between 0 and 1.The maximum value is put equal to one and minimum value to zero and all intermediate values are mapped between 0 and 1.

3.5 Selection of Input and Output Variables

The goal of neural networks is to generalize a relationship of the form

$$Y^n = f(X^m) \tag{1}$$

where X^n is an n-dimensional input vector consisting of variables $x_1, \ldots\ldots x_i, \ldots\ldots x_n$; and Y^m is an m-dimensional output vector consisting of resulting variable of interest $y_1 \ldots\ldots y_j, \ldots\ldots y_m$.

The selection of an appropriate input vector that will allow neural networks to successfully map to the desired output vector is not a trivial task. Unlike physically based models, the set of variables that influence the system are not known a priori. A neural network should not be considered mere a black box, a firm understanding of the system under consideration is an important aspect.

In air quality, the value of x_i can be causal variables such as wind speed, downwind distance, crosswind distance, ambient air temperature, relative humidity, atmospheric stability etc. The values of y_i can be air quality parameters such as ambient concentration of ozone, nitrogen oxides, carbon monoxide, particulate matter etc.

3.6 Presenting the Input to Network

Data can be presented to the network in two ways. In first way, the input given to the network is the year or day number and the normalized values of the parameter for which the network is being trained are kept as the target values of the network. This technique has been applied to RNM and

CPDM models. In second way, the input to the network is the normalized value of the parameter that is being modeled. A number of the past values of the parameter that is being modeled may also be given as additional input to the network. The number of past values to be given as input varies from the model to model. In SNCM, and SOFM, the second technique has been used with normalized values (I_t) lying between 0.1 and 0.9 as follows:

$$_t = 0.5 - (W_{mean} - W_t) / (W_{max} - W_{min})*0.8 \quad (2)$$

where

W_{max} = maximum value in that pattern.

W_{min} = minimum value in that pattern.

$W_{mean} = (W_{max} + W_{min}) / 2$.

W_t = value of any element in that pattern.

After modeling, the normalized output predictions (W_{pr}) produced by model are reverse normalized using:

$$O_t = W_{mean} - (0.5 - W_{pr}) / 0.8*(W_{max} - W_{min}) \quad (3)$$

where

O_t = Predicted value produced by the model.

3.7 Performance Evaluation of Models for Case Studies

In the Section 2, the background of specific neural network models proposed for present study was discussed. The following four models have been implemented for carrying out simulations to forecast air quality.

- RNM: This is Recurrent Network model and Elaman Networks are used for the simulation.
- CPDM_RNM: This model uses Change point detection technique and Recurrent Elaman Networks for simulation.
- SNCM: This model utilizes the Recurrent Networks for simulation in a different way. It selects the network Architecture for time series modeling by adding neurons one by one and training the network again and again.
- SOFM: This model exploits the properties of Self-organizing Feature Maps and utilized for non-linear time series modeling.

3.8 Network Error Calculation

The performance of various models for different air quality parameters was estimated in terms of mean percentage error (PE) defined as follows over a specified number of iterations or for achieving the goal of specified sum squared error (SSE) by the network, whichever reach earlier:

$$PE = (target - output) / target * 100 \quad (4)$$

The model having minimum PE is expected to be the best model for forecasting.

4 Results and Discussion

This section provides the numerical experimentation carried out using various neural networks for two datasets.

4.1 Results of Case Study 1

The parameters of various models used for modeling annual time series data are presented in Tables 3 to 6. The model performance was evaluated using following approache:

Training on nine data points of time series and predicting remaining part of time series six data points. The performance of various models for different air quality parameters was estimated in terms of mean percentage error (PE). The model having minimum PE is expected to be the best model for forecasting. The performance of various models have been compared in Table 7 and discussed in forthcoming section.

Table 3. The RNM Parameters (Case Study 1)

Air-quality Parameter	VOC	NOx/CO/ SO_2/PM10	PM2.5/NH_3
Model Parameters↓			
NN Architecture	1-6-6-1	1-5-5-1	1-5-5-1
Activation Function	Tansig/purelin	Tansig/purelin	Tansig/purelin
Learning Rate	0.03	0.03	0.04
No. of Epochs	5000	5000	5000
SSE	1.0e-5.	1.0e-5.	1.0e-5.
Input parameters	Yr. No	Yr. No	Yr. No
Output Parameters	VOC	NOx/ CO/ SO_2/PM10	PM2.5//NH_3

Table 4. The CPDM_RNM Parameters (Case Study 1)

Air-quality Parameter	VOC	NOx/ CO/ SO_2/PM10	PM2.5/NH_3
Model Parameters↓			
NN Architecture	1-6-6-1	1-5-5-1	1-5-5-1
Activation Function	Tansig/purelin	Tansig/purelin	Tansig/purelin
Learning Rate	0.03	0.03	0.04
No. of Epochs	1000	1000	1000
SSE	1.0e-6.	1.0e-6.	1.0e-7.
Input parameters	Yr. No	Yr. No	Yr. No
Output Parameters	VOC	NOx/ CO/ SO_2/PM10	PM2.5/ NH_3

Table 5. The SNCM Parameters (Case Study 1)

Air-quality Parameter	VOC/NOx/ CO/ SO_2/PM10/ PM2.5/NH_3
Model Parameters↓	
NN Architecture	1-3-3-1 to 1-8-8-1.
Activation Function	Tansig/purelin
Learning Rate	0.03
No. of Epochs	5000.
SSE	0.1e-7.
Input parameters	Yr. No
Output Parameters	VOC/ NOx/ CO/ SO_2/PM10/ PM2.5/NH_3

Table 6. The SOFM Model Parameters (Case Study 1)

Air-quality Parameter	VOC/NOx/ CO/ SO_2/PM10/ PM2.5//NH_3
Model Parameters↓	
NN Architecture	1-4-1
Learning Rate	1
No. of Epochs	5000.
Input parameters	VOC/NOx/ CO/ SO_2/PM10/ PM2.5//NH_3
Output Parameters	VOC/NOx/ CO/ SO_2/PM10/ PM2.5//NH_3

Table 7. Neural networks models performance (Case Study 1)

Air-quality Parameter	VOC	NOx	CO	SO_2	PM10	PM2.5	NH3
Model ↓							
RNM	201	88	81	69	36	20.5	26.9
CPDM_RNM	46.87	42.01	45.905	27.72	14.56	7.71	21.49
SNCM	43.24	47.77	42.587	45.986	47.63	36.752	41.84
SOFM	**23.81**	**16.34**	**7.806**	**15.722**	**12.73**	**7.87**	**14.73**

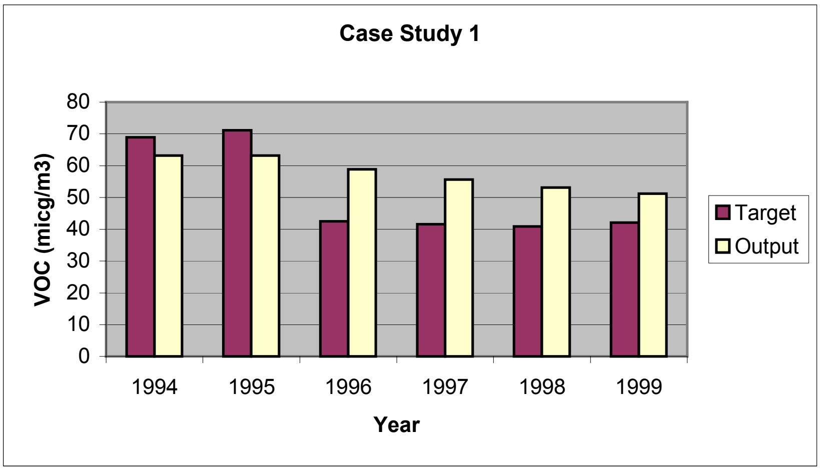

Figure 1. VOC emissions with SOFM

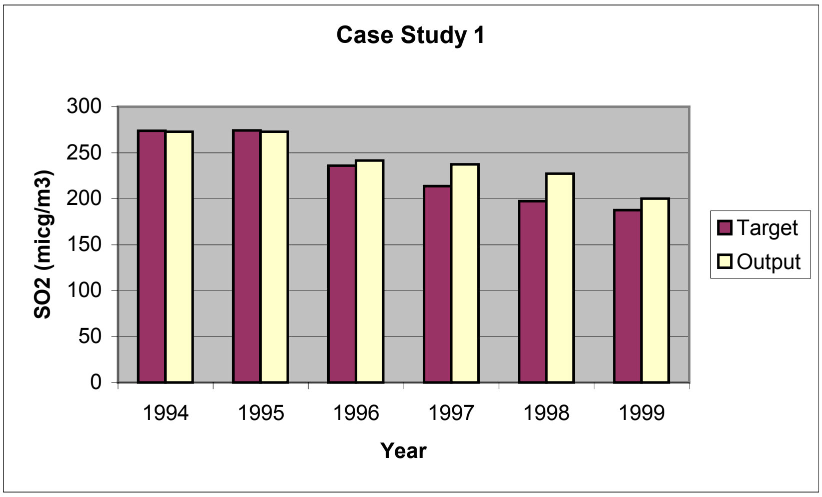

Figure 2. SO_2 emissions with SOFM

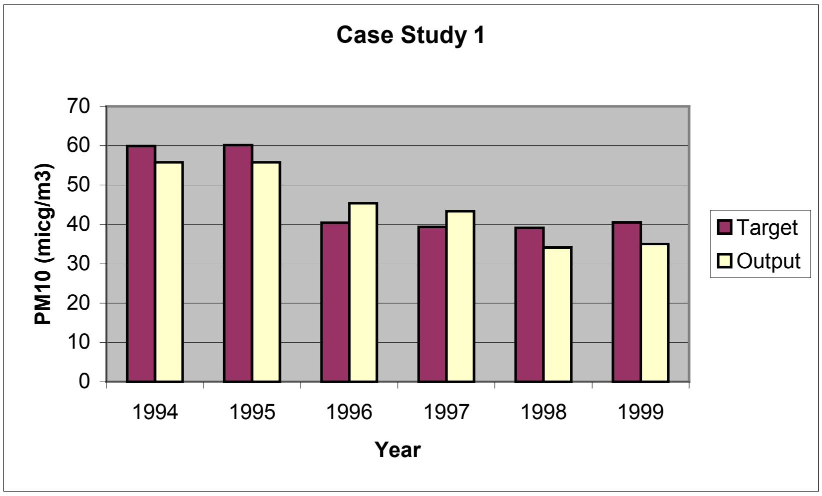

Figure 3. PM10 emissions with SOFM

4.1.1. Observations

From the Table 7, given in above section, following observations can be made.

- Time-series prediction method demonstrates the more realistic problems in the field. Here one would like to forecast about air quality parameters based on the past history. The models in general could predict with some accuracy. However, among all the models implemented, Self-organizing Feature Map (SOFM) based model has performed extremely well in comparison to other models. This performance could be attributed to the basic underlying characteristic of the algorithm. The algorithm localizes the data points of the time domain and classifies them in the cluster of similar characteristics. SOFM identifies the parameter values from the cluster when the new instance is asked to be predicted,
- The typical results for VOC emissions, CO emissions, and NO_x emissions are shown in Figures 1 to 3 for SOFM Model. The results highlight the very good performance of the model. The discrepancy observed in the model prediction can be due to the modeling of the problem. As discussed in earlier about SOFM model, the prediction of emission variables are dependent on past history of the data used for training. Measurement error in the data used for training may lead to prediction error.

- The case study demonstrated an example of an annual average emission (long-term) data prediction using various neural networks models for a very limited dataset. Models in general have performed reasonably well even with the limited historical data. It is expected that availability of more annual average emission data can improve the performance of models studied.

4.2 Results of Case Study 2

The parameters of various models for modeling daily average data are shown in Tables 8 to 11. The performance of various models was evaluated for following scenarios: Training on 80 data points of time series and predicting remaining part of time series 30 data points.

Table 8. The RNM Parameters (Case Study 2)

Air-quality Parameter	RPMA/ SO_2/NO_2
NN Architecture	1-10-10-1
Activation Function	Tansig/purelin
Learning Rate	0.04
No. of Epochs	5000
SSE	0.1e-4
Input parameters	Day No.
Output Parameters	RPMA/ SO_2 /NO_2

Table 9. The CPDM_RNM Parameters (Case Study 2)

Air-quality Parameter	RPMA/ SO_2 /NO_2
NN Architecture	1-4-4-1
Activation Function	Tansig/purelin
Learning Rate	0.03
No. of Epochs	5000
SSE	0.1e-4
Input parameters	Day No.
Output Parameters	RPMA/ SO_2 /NO_2

Table 10. The SNCM Parameters (Case Study 2)

Air-quality Parameter	RPMA/ SO_2 /NO_2
NN Architecture	1-3-3-1 to 1-8-8-1.
Activation Function	Tansig/purelin
Learning Rate	0.03
No. of Epochs	1000.
SSE	0.1e-7.
Input parameters	Day No.
Output Parameters	RPMA/ SO_2 /NO_2

Table 11. The SOFM Parameters (Case Study 2)

Air-quality Parameter	RPMA	SO_2	NO_2
NN Architecture	1-5-1	1-5-1	1-5-1
Learning Rate	1	1	1
No. of Epochs	5000.	5000.	5000.
Input parameters	RPMA	SO_2	NO_2
Output Parameters	RPMA	SO_2	NO_2

The models' performance in terms of PE has been shown in Table 12 for all three air quality parameters. The salient features of different models have been discussed and compared in next section.

Table 12. Neural networks models performance (Case Study 2)

Air-quality Parameter	RPMA	SO2	NO2
RNM	56.76	48.63	43.5
CPDM_RNM	45.36	41.83	38.9
SNCM	33.45	37.79	35.87
SOFM	**25.6**	**30.73**	**28.94**

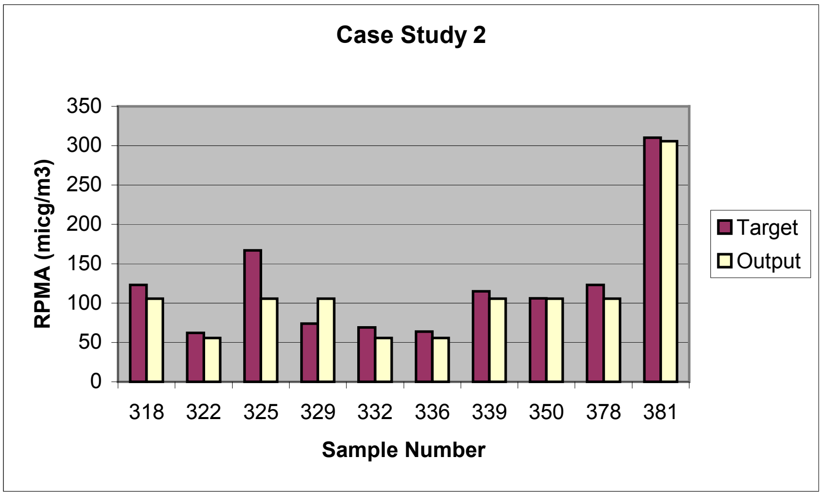

Figure 4. RPMA emissions with SOFM

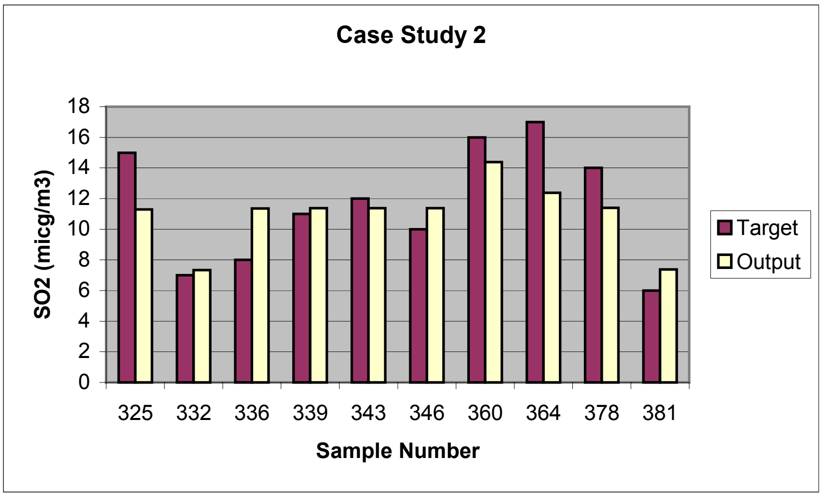

Figure 5. SO_2 emissions with SOFM

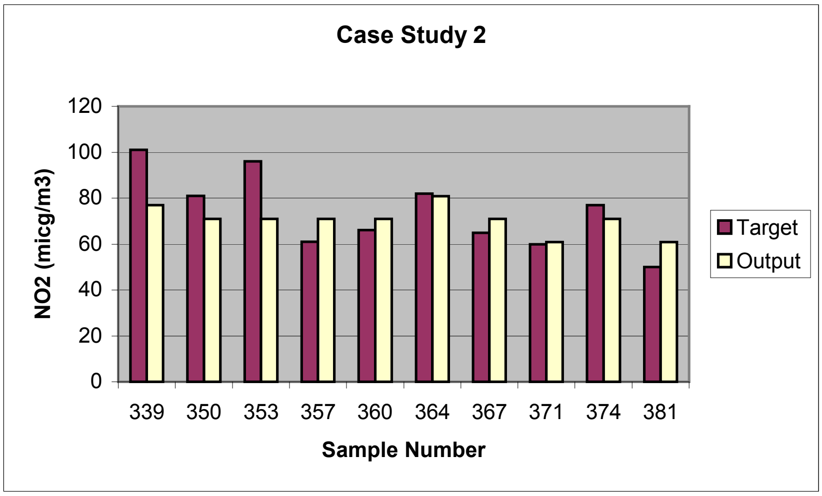

Figure 6. NO_2 emissions with SOFM

4.2.1. Observations

The following observation can be made on the basis of results of various simulations shown in Table 12.

- The models studied for this case, in general could predict with modest accuracy. However, among all the models implemented, SOFM based model has performed extremely well in comparison to other models. This may be attributed to the fact that network learns the history much more. Similar trend was observed during earlier case study too.
- As discussed above, the typical results for RPMA emissions, SO_2 emissions, and NO_2 emissions are shown in Figures 4 to 6. The results shown in Figures 4 to 6 depicts reasonably good match between model predictions with target prediction.

The case study demonstrated an example of a daily average emission data prediction using various neural networks model for a very large size dataset. Models in general have performed reasonably well even though data was showing randomness in time domain.

5 Future Projections

The predictions of studied models are based on the limited history of air quality. However, the model prediction can be improved by carrying out investigation incorporating following aspects:

- Models can have as inputs data from multiple sources, such as historical air quality measurements, meteorological data etc.
- Models can have along with emissions data, episode levels definition, historical measurements of surface and upper air meteorological data.
- Models should be able to give predictions for the following three different time windows: 1 day, 1 week and 1 month predictions.
- Models should have an easy to use interface and should be able to present the results in an understandable way to non-computer experts.
- The model parameters and architecture of models in this project were arrived with trial and error. One can arrive at optimal and better performance model after carrying out systematic studies on networks models and their parameters using optimization techniques such as Genetic Algorithm.

6 Closing Remarks

In this paper, the study was carried out on air quality forecasting using various neural network models: RNM, CPDM_RNM, SNCM and SOFM. The study was focused at preliminary investigation of single variable based time series prediction. The investigation was carried out for long-term as well as short-term air quality data set. Self-Organizing Features Maps (SOFM) used for time series prediction came up as the best tool for time series forecasting. These were found to be very useful for large training datasets. The results shown here are indications that the neural network techniques can be useful tool in the hands of practitioners of air quality management and prediction. In that case, practitioners need not know even about the development of the model. The models studied in this study are easily implemented, and they can deliver prediction in real time, unlike other modeling techniques. The models can very well easily deal with input noise and uncertainty.

References

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000a), "Artificial neural networks in hydrology. I: preliminary concepts", *J. Hydrologic Engineering*, ASCE, 5(2), 115-123.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000b), "Artificial neural networks in hydrology. II: preliminary concepts", *J. Hydrologic Engineering*, ASCE, 5(2), 124-137.

Back, A.D., and Tsoi, A.C. (1991), "FIR and IIR synapses, a new neural network architecture for time series modeling", *Neural Computation*, 3, 375-385.

Basheer, A., and Najjar, M. (1996), "Predicting dynamic response of adsorption columns with Neural Nets", *J. Computing in Civil Engineering*, ASCE, 10(1), 211-220.

Boznar, M., Lesjak, M., and Mlakar, P (1993), "A neural network-based method for short term prediction of ambient SO_2 concentrations in highly polluted industrial areas of complex terrain", *Atmos. Environ.,* 27B, 221-230.

Chakraborty, K., Mehrotra, K., Mohan, C.K., and Ranka, S. (1992), "Forecasting the behaviour of multivariate time series using neural networks", *Neural Networks*, 2, 53-58.

Connor, J.T, Martin, R.D., and Atlas, L.E. (1994), "Recurrent neural networks and robust time series prediction", *IEEE Trans. Neural Networks*, 5(2).

Demuth, H., and Beale, M. (1992), "*Neural Networks Toolbox Manual"*, Math Works Inc.

Giorgio, F., and Piero, M. (1996), "Mathematical models for planning and controlling air quality*"*, *Proceedings of IIASA Workshop*, 17.

Hadjiiski, L., and Hopke, P. (2000), "Application of artificial neural networks to modeling and prediction of ambient ozone concentrations", *J. Air and Waste Manage. Assoc*., 50, 894-901.

Haykin, S. (2000), "*Neural Networks*", Second Edition, Addison Wesely Longman.

Kohonen, T. (1990), "The self-organizing map", *Proceedings of the IEEE*, 78, 1464-1480.

Kyong, J.O., and Han, I. (2000), "Using change-point detection to support artificial neural networks for interest rate forecasting", *Expert System with Applications,* 19, 105-115.

Math Works (2000), "*MATLAB User's Guide*", Math Works Inc.

Melas, D., Kioutsioukis, I., and Ziomas, I.C. (2000),"Neural network models for predicting peak photochemical pollutant levels", *J. Air and Waste Manage. Assoc.,* 50, 495-501.

Parlos, A.G., Rais, O.T., and Atiya, A.F. (2000), "Multi-step-ahead prediction using dynamic recurrent neural networks", *Neural Networks,* 13, 765-786.

Pettitt, A.N. (1979), "A non-parametric approach to the change-point problem", *Applied Statistics*, 28(2), 126-135.

Principe, J., and Wang, L. (1993), "Non-linear time series modeling with self-organization feature maps", *Neural Networks,* 3.

Principe, J., and Kuo, J.M. (1995), “Dynamic modeling of chaotic time series with neural networks”, *Proc. of Neural Infor. Proc. Sys*., 7.

Rao, M.N., and Rao, H.V. (2000), ”*Air Pollution*”, Tata McGraw-Hill, New Delhi.

Sharma, S. (2002), "*Neuro Air Quality Predictors*", M.Tech. Thesis, IIT Kharagpur.

Thirumalaiah, K., and Deo, M.C. (1998), “River stage forecasting using artificial neural networks”, *Journal of Hydrologic Engineering*, ASCE, 3(1), 26-32.

Thirumalaiah, K., and Deo, M.C. (2000), “Hydrological forecasting using neural networks”, *Journal of Hydrologic Engineering*, ASCE, 5(2). 180-189.

Tomasz, J.C., and Zurada, J.M. (1977), "Sequential Network Construction for time series prediction", In *Proceedings of the IEEE International Joint Conference on Neural Networks,* 2034–2039, Houston, Texas, USA, June 9–12.

Warren, S. S. (2002), "*FAQ in Neural Networks*", ftp://ftp.sas.com/pub/neural/FAQ.html