Atmospheric Temperature Prediction using Support Vector Machines

Y.Radhika and M.Shashi

Abstract—**Weather prediction is a challenging task for researchers and has drawn a lot of research interest in the recent years. Literature studies have shown that machine learning techniques achieved better performance than traditional statistical methods. This paper presents an application of Support Vector Machines (SVMs) for weather prediction. Time series data of daily maximum temperature at a location is analyzed to predict the maximum temperature of the next day at that location based on the daily maximum temperatures for a span of previous n days referred to as order of the input. Performance of the system is observed over various spans of 2 to 10 days by using optimal values of the kernel function. Non linear regression method is found to be suitable to train the SVM for this application. The results are compared with Multi Layer Perceptron (MLP) trained with back-propagation algorithm and the performance of SVM is found to be consistently better.**

Keywords—**Comparative Studies, Neural Networks, Regression, Time Series, Weather Forecasting**

I. INTRODUCTION

Weather prediction is a complex process and a challenging task for researchers. It includes expertise in multiple disciplines [1], [2]. The prediction of atmospheric parameters is essential for various applications. Some of them include climate monitoring, drought detection, severe weather prediction, agriculture and production, planning in energy industry, aviation industry, communication, pollution dispersal [6] etc. Accurate prediction of weather parameters is a difficult task due to the dynamic nature of atmosphere. Various techniques like linear regression, auto regression, Multi Layer Perceptron, Radial Basis Function networks are applied to predict atmospheric parameters like temperature, wind speed, rainfall, meteorological pollution etc. [4], [5], [6], [7], [10], [11]. It was found that the non linear operator equations governing the atmospheric system are the ones who can better understand the dynamics of atmosphere. In the recent past many forecast methods have been developed using Artificial Neural Networks

(ANNs). Neural network techniques have the potential to handle complex, nonlinear problems in a better way when compared to traditional techniques. However systems developed using neural network model suffer from certain

Y.Radhika is with the Computer Science Engineering Department, GITAM University, Visakhapatnam, Andhra Pradesh INDIA. (Ph: 91-9985225454)

M.Shashi is with the Computer Science and Systems Engineering Department, Andhra University, Visakhapatnam, Andhra Pradesh INDIA. (Ph: 91-9949072880).

drawbacks like local minima, model over fitting etc. This work aims at a comparative statement on the performance of SVM and MLP while predicting the maximum temperature of a day based on the data of the previous n days.

II. SUPPORT VECTOR MACHINES

Support Vector Machine algorithm was developed by Vapnik [3] and is based on statistical learning theory. The basic idea of Support Vector Machines is to map the original data X into a feature space F with high dimensionality through a non linear mapping function and construct an optimal hyperplane in new space. SVM can be applied to both classification and regression. In the case of classification, an optimal hyperplane is found that separates the data into two classes. Whereas in the case of regression a hyperplane is to be constructed that lies close to as many points as possible.

In this paper Support Vector Regression (SVR) is used to predict the maximum temperature at a location. Support vector regression is different from conventional regression techniques because it uses Structural Risk Minimization (SRM) but not Empirical Risk Minimization (ERM) induction principle which is equivalent to minimizing an upper bound on the generalization error and not the training error. Due to this feature it is expected to perform better than conventional techniques which may suffer from possible over fitting.

Regression is the problem of estimating a function based on a given data set. Consider a data set $G = \{(x_i, d_i)\}_{i=1}^{N}$, where x_i is the input vector, d_i is the desired result and N corresponds to the size of the data set. The general form of Support Vector Regression [3], [8], [9] estimating function is

$$f(x) = (w.\phi(x)) + b$$

where w and b are the co-efficients that have to be estimated from data. $\phi(x)$ is the non linear function in feature space.

The non linear regression problem can be expressed as an optimization problem in the following way by minimizing the regularized risk function.

$$R(C) = \frac{1}{2}\|w\|^2 + C\frac{1}{N}\sum_{i=1}^{N} L_{\in}(d_i, y_i)$$

where

$$L_{\in}(d, y) = \left\{ \begin{matrix} |d-y|-\in & |d-y|\geq\in \\ 0 & others \end{matrix} \right\}$$

After introducing slack variables the risk function can be expressed in the following constrained form

$$\text{minimize } R(w, \xi^*) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*)$$

subject to

$$d_i - w\varphi(x_i) - b_i \leq \in +\xi_{i,}$$

$$(w.\varphi(x)) + b - d_i \leq \in +\xi_\iota^*$$

$$where\ \xi_i, \xi_i^* \geq 0$$

The dual form of the non linear SVR can be expressed as

$$Q(\alpha_\iota,\alpha_\iota^*) = \sum_{i=1}^{N} d_i(\alpha_i - \alpha_i^*) - \in \sum_{i=1}^{N}(\alpha_i + \alpha_i^*) - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(X_i, X_j)$$

subject to

$$\sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0$$

$$0 \leq \alpha_i, \alpha_i^* \leq C, i = 1, 2, ...n$$

α_i and α_i^* are the Lagrange multipliers that act as forces pushing the predictions towards target value d.

The computation in input space can be performed using a kernel function $K(X_i, X_j) = \varnothing(X_i).\varnothing(X_j)$ in feature space. Any function that satisfies Mercer's theorem [3] can be used as a kernel function. Some of the functions that satisfy this condition are shown in Table 1.

TABLE I: SOME OF THE KERNEL FUNCTIONS THAT SATISFY MERCER'S THEOREM.

kernel	Function	Comment
Polynomial	$[1+(X.X_i)]^p$	Power p is specified by the user
RBF	$\exp(-\frac{1}{2\sigma^2}\|X-X_i\|^2)$	is specified σ^2 The width by the user

The parameters C, $\in$ and kernel parameters are user defined. C controls the smoothness of approximating function and $\in$ determines the margin within which the error is tolerated. Finally the SVR estimating function can be expressed as

$$f(x) = \sum_{i=1}^{nsv}(\alpha_\iota - \alpha_\iota^*)K(X, X_i) + b$$

where nsv is the number of support vectors.

III. MULTI LAYER PERCEPTRON

Multi Layer Perceptron network is a class of neural networks which consists of a set of sensory units that constitute the input layer and one or more hidden layers of computation nodes and an output layer of computation nodes. A non linear activation function namely sigmoid function is widely used to generate output activation in the computation nodes. In general MLPs are trained with the back propagation algorithm to develop successful classification and regression systems.

Back propagation algorithm involves four stages
1) Initialize weights to small random values.
2) Feed forward stage- In this stage each input unit receives an input signal and transmits this signal to each of the hidden units. Each hidden unit calculates the activation function and sends its signal to each output unit. The output unit calculates the activation function to form the response of the net for the given input pattern.

3) Back propagate the errors- After comparing the computed activation with its target, the associated error for that pattern is calculated and is distributed back to all units in the previous layer.
4) Updation of weights and biases- The weights and biases are updated and finally the stopping condition may be minimization of error, number of epochs etc.

IV. DATA SET AND PREPROCESSING

The weather data of University of Cambridge for a period of five years (2003-2007) is used to build the models and the data between January and July of the year 2008 is used to test the models. The database includes readings of several weather parameters recorded at every half hour interval. The daily maximum temperature is extracted from this database and used for this work.

The real world databases are highly susceptible to noisy and missing data. The data can be preprocessed to improve the quality of data and thereby improve the prediction results. In this work data cleaning and transformation have been applied to the data. Data cleaning fills in the missing values, while data transformation improves the accuracy, speed and efficiency of the algorithms used.

The missing value for maximum temperature of a day is replaced with the mean of maximum temperature for that month while building the SVM model. The data is normalized using Z-score normalization where the values of an attribute, A, are normalized based on the mean ($\bar{A}$) and standard deviation (σ_A) of A. The normalized value V' of V can be obtained as

$$V' = (V - \bar{A})/\sigma_A$$

However, it was found that the negative values generated by Z-score normalization cannot be fed as input to the MLP model. Hence min max normalization is applied to transform the input into a range of 0 to 1 to build the MLP model.

V. .METHODOLOGY

In this work Maximum Temperature of a day is predicted based on the maximum temperature of previous n days where n is the optimal length of the span. The value of n is found by experimentation. The available data is divided into training, validation and test sets. Training set is used to build the model, validation set is used for parameter optimization and test set is used to evaluate the model. Separate models are developed using SVM and MLP trained with back propagation algorithm.

A non linear support vector regression method is used to train the SVM. A kernel function must be selected from the functions that satisfy Mercer's theorem. After applying polynomial and radial basis function (RBF) on sample data it was found that radial basis function performs better, and hence RBF is adopted in this study. The RBF kernel function requires setting of the parameter σ in addition to the regular parameters C and $\in$. As there are no general rules to determine the free parameters the optimum values are set by two stage grid search method [12]. Initially a coarse grain search (i.e. for C=1, 10, 100; $\in$ =1, 2; γ=1, 2, 3) is performed to identify the near to optimal values, and then a fine grain search (i.e. for C=0.5, 0.7, 0.9, 1; $\in$ =0.001, 0.01, 0.1, 1; γ=0.001, 0.01, 0.1, 1) is done to identify the optimum values.

Back propagation algorithm is used to develop the MLP models for different orders. A three layer neural network with

one input layer, one hidden layer and one output layer is used to develop the model. The activation function used is sigmoid function. Number of neurons used in the hidden layer are around ($2i+1$) where i is the number of inputs. Training is continued till a specified number of iterations.

VI. RESULTS AND DISCUSSION

The performance of the developed models is assessed after de normalizing the output generated by the models. Mean Square Error (MSE) is taken as a performance measure in this work.

The performance of Multi Layer Perceptron trained with back propagation algorithm and SVM for different orders in terms of MSE is shown in Fig 1. From the obtained results it has been observed that the window size does not have a significant effect on the performance of MLP trained with back propagation algorithm and SVM. However it can also be observed that irrespective of the order, SVM performs better than MLP. The Mean Square Error in the case of MLP varies from 8.07 to 10.2 based on the order, whereas it is in the range of 7.07 to 7.56 in the case of SVM. It can be seen that in the case of SVM there is no noticeable change in the performance of the system from order 5 onwards and MLP performed best for order 5. The error seems to reduce slightly for higher orders in the case of SVM but the training time also increases proportionately with the order. Hence order 5 is selected as optimal window size. The predictions based on a window size of 5 while sliding over a span of 193 days in sequence is shown in Fig 2.

TABLE 2: MEAN SQUARE ERRORS OF SVM AND MLP FOR DIFFERENT ORDERS

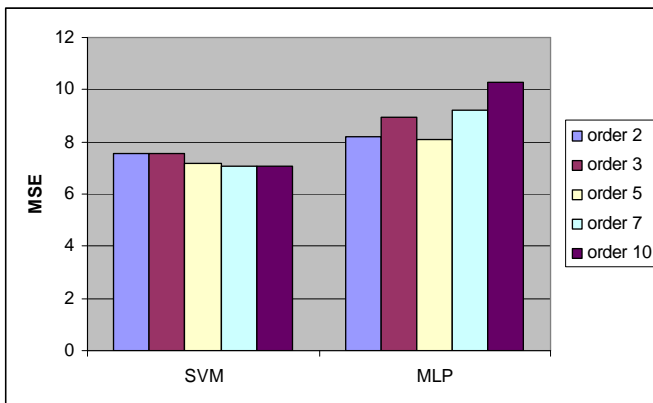| Order	2	3	5	7	10
SVM	7.56	7.52	7.15	7.092	7.07
MLP	8.21	8.95	8.07	9.19	10.26

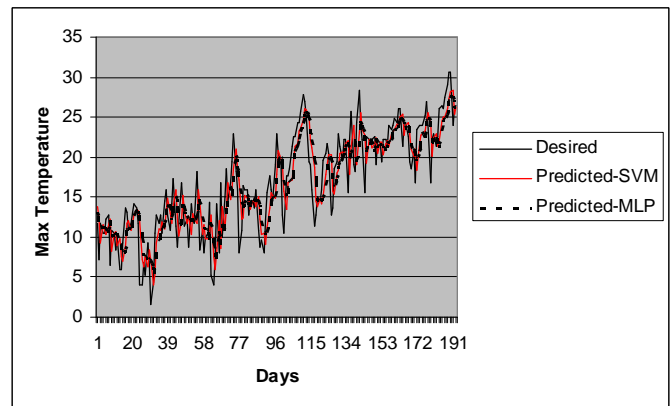Fig. 1 Comparison between MLP and SVM for different orders

Fig. 2 Performance of MLP and SVM on test data

VII. CONCLUSIONS

An application of support vector regression for atmospheric temperature prediction is presented in this paper. The performance of SVM was compared with MLP for different orders. The results obtained show that SVM performs better than MLP trained with back propagation algorithm for all orders. It was also observed that parameter selection in the case of SVM has a significant effect on the performance of the model. It can be concluded that through proper selection of the parameters, Support Vector Machines can replace some of the neural network based models for weather prediction applications.

REFERENCES

[1] Denis Riordan, and Bjarne K Hansen, "A fuzzy case-based system for weather prediction." Engineering Intelligent Systems, Vol .10, No.3. 2002

[2] Guhathakurtha, P., "Long-Range monsoon rainfall prediction of 2005 for the districts and sub-division kerala with artificial neural network." Current Science, Vol.90, No.6, 25. 2006

[3] Haykin, S., "Neural Networks- A Comprehensive Foundation." Prentice Hall. 1999.

[4] Jae H.Min., Young-chan Lee. "Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters." Expert Systems with Applications, 28, pp.603-614. 2005.

[5] Mohandes, M.A., Halawani, T.O., Rehman, S and Ahmed Hussain, A. "Support vector machines for wind speed prediction." Renewable Energy, 29, pp.939-947. 2004

[6] Pal, N.R., Srimanta Pal, Jyotirmoy Das, and Kausik Majumdar, "SOFM-MLP: A Hybrid Neural Network for Atmospheric Temperature Prediction." IEEE Transactions on Geoscience and Remote Sensing, Vol.41, No, 12, pp.2783-2791. 2003

[7] Pao-Shan Yu., Shein-sung Chen., I-Fan Chang. "Support vector regression for real- time flood stage forecasting." Journal of Hydrology, 328, pp. 704-716. 2006

[8] Ronan Collobert and Samy Benegio "SVM Torch: Support Vector Machines for Large-Scale Regression Problems." Journal of Machine Learning Research, 1, pp.143- 160. 2001

[9] Smola A.J, and Scholkopf B, "A Tutorial on support vector regression, " Neuro COLT Technical Report NC-TR-98-030, Royal Holloway College, University of London, UK. 1998

[10] Stanislaw Osowski and Konrad Garanty, "Forecasting of daily meteorological pollution using wavelets and support vector machine." Engineering Applications of Artificial Intelligence, 20, pp.745-755. 2007

[11] Wei-Zhen Lu. Wen-Jian Wang. "Potential assessment of the support vector machine method in forecasting ambient air pollutant trends." Chemosphere, 59, pp.693-701. 2005

[12] Hsu, C.-W., Chang, C-C., Lin C.-J."A Practical guide to support vector classification." Technical report, Department of Computer science and information Engineering, National Taiwan University.

Y.Radhika received her B.E (Civil Environmental Engineering) and M.Tech (Computer Science and Technology) degrees from Andhra University. She is currently an Associate Professor in the Department of Computer Science Engineering at GITAM University. She is working for her PhD in the area of Neural Networks. Her current research interest includes Data Mining, Neural Networks, Machine Learning, etc.

M.Shashi received her B.E (EEE) and M.E (Computer Engineering) degrees with distinction from Andhra University, Visakhapatnam, India. She received the best thesis award for her PhD in Artificial Intelligence from Andhra University, and AICTE Career Award as Young Teacher at the national level. She is currently a Professor in the Department of Computer Science and Systems Engineering at Andhra University. She has been publishing in national and international journals; authored a book published by Pearson Education Ltd. Her research interest includes Data Mining, Text Mining, Soft Computing, Machine Learning, etc.