Volume: 1 Issue: 1

Dynamic Model Representation in the Locality Frequent Neural Networks

[1]Jitendra Joshi, [2]Ritu Nagwani, [3]Renu Deswal, [4]Priyanka Sharma

[1]Ph.D. Scholar, Jayoti Vidhyapeeth Women's University, Jaipur, E-mail: lect.jitendra29@gmail.com

[2]B.Tech. Scholar, Jayoti Vidhyapeeth Women's University, Jaipur, E-mail: ritunagwani09@gmail.com

[3]B.Tech. Scholar, Jayoti Vidhyapeeth Women's University, Jaipur, E-mail: renudeswal28@gmail.com

[4]B.Tech. Scholar,Jayoti Vidhyapeeth Women's University, Jaipur,

E-mail: 02priyanka10sharma1993@gmail.com

Abstract

Dynamic model presentation is system identification and mathematical calculation of neural network. Dynamic system using neural networks involve the dynamic differential equation into each of the neural network processing elements to create a new type of neuron called a dynamic neuron. The dynamic model represents architecture of frequent neural network in different paradigm's model.

Index Terms: *Dynamic Model, FNN, BMLP, TDL, LRNN, LF-MLN, AF-MLP, OF-MLP*

1. Introduction

Dynamic model presentation is mathematical calculation of neural network and dynamic recurrent neural networks have shown themselves to be really useful for temporal processing, particularly for digital signal processing (DSP) and temporal pattern recognition of neurons. Dynamic system using neural networks involve the dynamic differential equation into each of the neural network processing elements to create a new type of neuron called a dynamic neuron [1]. Two main methods exist to provide a static neural network with dynamic behavior: the insertion of a buffer somewhere in the network, i.e., implementing an explicit memory of the past inputs, or the use of feedback.

The first kind of dynamic network is a buffered multilayer perceptron (BMLP), in which tapped delay lines (TDL's) of the inputs are used. The latter approach brings us to the so called locally recurrent neural networks (LRNNs) or local feedback multilayer networks (LF-MLN) that will be the architectures we will be concentrated on [2].

Different architectures arise depending on The first architecture is the IIR-MLP proposed by Back and Tsoi. The second architecture is the activation feedback multilayer perceptron network (AF-MLP). The third structure is the Output Feedback multilayer perceptron network (OF-MLP) [3].

2. LOCALITY FREQUENT NEURAL NETWORKS

We are here interested in neural network architectures that are able to learn temporal features in data for time series prediction. In other words, we intend to deal with the problem of Time Series Processing: field of statistics regarding the analysis of data characterized by both spatial and temporal dimensions. The FFNN is frequently used in time series prediction. FFNN, however, has the major limitation that it can only learn an input and output mapping which is static. Thus it can be used to perform a nonlinear prediction of a stationary time series [4]. A time series is said to be stationary, when its statistics do not change with time. In many real world problems, however, the time when certain feature in the data appears, contains important information. More specifically, the interpretation of a feature in data may depend strongly on the earlier features and the time they appeared. A common example of this phenomenon is speech. A good solution is to let time have an effect on the networks response. This can be achieved when the network has dynamic properties such that it will respond to temporal sequences [5]. Dynamic recurrent neural networks have shown themselves to be really useful for temporal processing, particularly for digital signal processing (DSP) and temporal pattern recognition. Two main methods exist to provide a static neural network with dynamic behavior: the insertion of a buffer somewhere in the network, i.e., implementing an explicit memory of the past inputs, or the use of feedback [6].

In both approaches, an arbitrary input $x[t]$ may influence a future output $y[t + h]$, so that $\partial y[t + h]\ \partial x[t]$ is not equal to zero for some h. In the case of asymptotic stability, this derivative goes to zero when h goes to infinity. The value of h for which that derivative becomes negligible is called temporal depth, while the number of adaptable parameters divided by the temporal depth is called temporal resolution.

3. BUFFERED MULTILAYER PERCEPTRON WITH INPUT BUFFER

The first kind of dynamic network is a buffered multilayer perceptron (BMLP), in which tapped delay lines (TDL's) of the inputs are used. The buffer can be applied at the network inputs only, keeping the network internally static, as in buffered MLP (Figure 1):

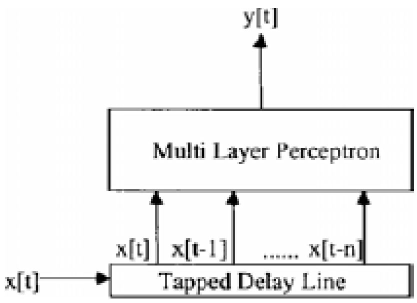

Figure 1: Buffered multilayer perceptron with input buffer.

or at the input of each neuron, as in MLP with Finite Impulse Response filter synapses [FIR-MLP](Figure 2), sometimes called time-delay neural network (TDNN), and in adaptive time-delay neural networks. It can be shown that BMLP and FIR-MLP are theoretically equivalent, since internal buffers can be implemented as an external one [8].

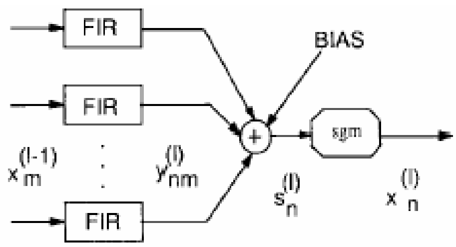

Figure 2: Buffered multilayer perceptron with input buffer.

The problem with implementing FIR-MLPs as buffered MLPs is that first layers sub networks must be replicated (with shared weights) and so the complexity is much higher than considering the buffer internal. Therefore, buffered MLP and FIR-MLP are different architectures with regard to areal implementation. The main disadvantage of the buffer approach is the limited past history horizon thereby preventing modeling of arbitrary long time dependencies between inputs and desired outputs [9]. It is also difficult to set the length of the buffer, given a certain application moreover to have sufficient temporal depth, a long buffer, i.e., a large number of input weights, could be required, usually with a decrease in generalization performance and an increase in the overall computational complexity.

In other words, the buffer approach with no feedback has the maximum temporal resolution, at the cost of a low temporal depth. To adaptively balance temporal depth with temporal resolution, another buffer type, called gamma memory, can be adopted, for which the delay operator, used in conventional TDLs, is replaced by a single pole discrete time filter [10]. Gamma memory is a dispersive delay line with dispersion regulated by an adaptable parameter.

In addition to these advantages of temporal depth and temporal resolution characteristics, it is known that neural networks with feedback have useful dynamic modeling behavior. Feedback has been implemented for the first time with the introduction of the so called fully recurrent neural networks (RNN); they are formed by a single layer of neurons fully interconnected with each other, or several such layers [10]. Such RNNs, however, exhibit some well known disadvantages: a large structural complexity (that is too many weights) and a slow and difficult training. As a matter of fact, they are very general architectures which can model a large class of dynamical systems. Nevertheless, on specific problems, simpler dynamic neural networks, which make use of available prior knowledge, can be better. Many efforts have been made with the aim of introducing temporal dynamics into the multilayer perceptron neural model. These efforts have paid in terms of less complex architectures and easier training, with respect to the RNNs. The major difference among the methods developed for the purpose lies in how feedback is included in the network [11].

4. Externally Structure of MLP Network

As in the Narendra–Parthasarathy MLP, also known as NARX network, where TDLs are also used for the outputs that are then brought back to the input of the network (Figure 3). Another example is the Elman's network (Figure 4).

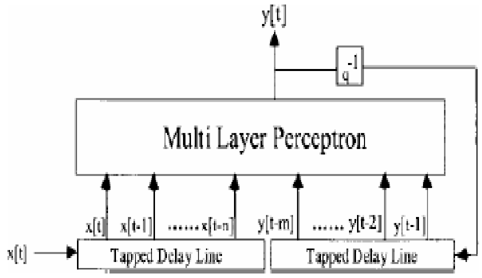

Figure 3: (a) Narendra–Parthasarathy MLP

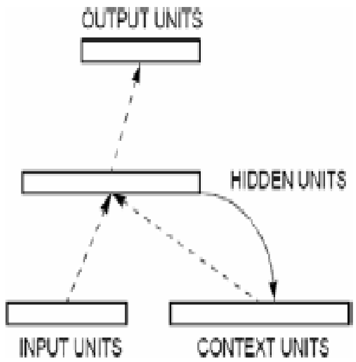

Figure 4: Elman's network.

5. INTERNALLY: INSIDE EACH NEURON

The latter approach brings us to the so called locally recurrent neural networks (LRNNs) or local feedback multilayer networks (LF-MLN) that will be the architectures we will be concentrated on. In these structures, classical infinite impulse response (IIR) linear filters, also called autoregressive moving average (ARMA) models, are used, either directly, or with some modifications [12]. Different architectures arise depending on how the ARMA model is included in the network. The first architecture is the IIR-MLP proposed by Back and Tsoi, where static synapses are substituted by conventional IIR adaptive filters (Figure 5).

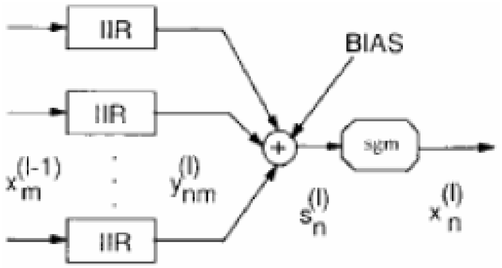

Figure 5: IIR-MLP neuron structure

6. AF-MLP NEURON STRUCTURE

The second architecture is the activation feedback multilayer perceptron network (AF-MLP). The output of the neuron summing node is filtered by an autoregressive (AR) adaptive filter (all poles transfer function), before feeding the activation function. In the most general case, the synapses are FIR adaptive filters. The AF-MLP is a particular case of the IIR-MLP, when all the synaptic transfer functions of the same neuron have the same denominator.

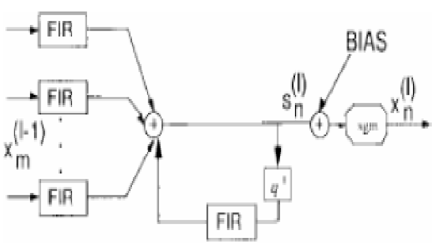

Figure 6: AF-MLP neuron structure

7. OF-MLP NEURON STRUCTURE

The third structure is the Output Feedback multilayer perceptron network (OF-MLP), in which the IIR filter is modified in order to let the feedback loop pass through the nonlinearity, i.e., the onetime step delayed output of the neuron is filtered by a FIR filter whose output is added to the inputs contributions, providing the activation. Again, in the general model the synapses can be FIR filters (Figure 7).

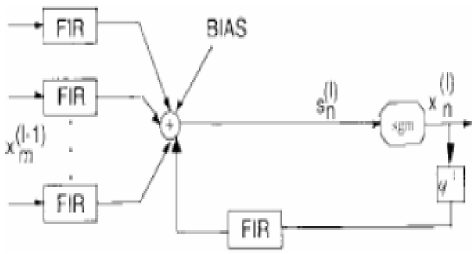

Figure 7: OF-MLP neuron structure

Elman's network is based on the introduction of context units to include memory in a network, substituting the spatial metaphor of the external buffer with the recurrent context approach [13]. Context units are dynamic recurrent neurons placed in the first layer to process the input signals, while the following layers are supposed to be static. This architectural constraint has been chosen basically to simplify the learning phase.

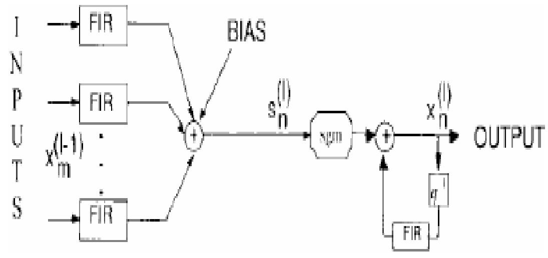

Figure 8: Auto-regressive MLP

At last, there is architecture (Figure 8) that's again a multilayer network, where each neuron has FIR filter synapses and an AR filter after the activation function (ARMLP). It is easy to see that this network is a particular case of the IIR-MLP, followed by linear all-pole filters [14, 15].

The major advantages of LFNNs with respect to buffered MLPs or fully recurrent networks can be summarized as follows:

1) Well-known neuron interconnection topology, i.e., the efficient and hierarchic multilayer;

2) Small number of neurons required for a given problem, due to the use of powerful dynamic neuron models;

3) Generalization of the popular FIR-MLP (or TDNN) to the infinite memory case;

4) Pre-wired forgetting behavior, needed in applications such as DSP, system identification and control;

5) Simpler training than fully recurrent networks;

6) Many training algorithms could be derived from filter theory and recursive identification.

8. CONCLUSION

Dynamic model presentation is mathematical calculation of neural network and dynamic recurrent neural networks have shown themselves to be really useful for temporal processing, particularly for digital signal processing (DSP) and temporal pattern recognition of neurons. A good solution is to let time have an effect on the networks response. This can be achieved when the network has dynamic properties such that it will respond to temporal sequences.

9. REFERENCE

[1] G. C. Goodwin and R. L. Payne, Dynamic System Identification, Academic Press, New York, 1977.

[2] K. Hornik, M. Stinchcombe and H. White, Multilayer Feed-forward Networks are Universal Approximators", Neural Networks Vol. 2. 1989. pp. 359-366.

[3] G. Cybenko, Approximation by Superposition of Sigmoidal Functions,

Mathematical Control Signals Systems, Vol. 2. pp. 303-314, 1989.

[3] K. I. Funahashi, On the Approximate Realization of Continuous Mappings by Neural Networks", Neural Networks, Vol. 2. No. 3. pp. 1989. 183-192.

[4] M. Leshno, V. Y. Lin, A. Pinkus and S. Schocken, Multilayer Feed-forward Networks With a Nonpolynomial Activation Function Can Approximate Any Function, Neural Networks, Vol. 6. 1993. pp. 861-67

[5] J. S. Albus, A New Approach to Manipulator Control: The Cerebellar Model Articulation Controller (CMAC),Transaction of the ASME, Sep. 1975. pp. 220-227.

[6] Y. H. Pao, Adaptive Pattern Recognition and Neural Networks, Addison-Wesley, Reading, Mass., 1989, pp. 197-222.

[7] D. F. Specht, Polynomial Neural Networks, Neural Networks, Vol.3. No. 1 pp. 1990. pp. 109-118,

[8] J. Park and I. W. Sandberg, Approximation and Radial-Basis-Function Networks, Neural Computation, Vol 5. No. 2. 1993. pp. 305-316.

[9] S. Haykin, Neural Networks. A comprehensive foundation, Second Edition, Prentice Hall, N. J.1999.

[10] M. H. Hassoun, Fundamentals of Artificial Neural Networks, MIT Press, Cambridge, MA. 1995.

[11] M. Brown and C.Harris, Neurofuzzy Adaptive Modelling and Control, Prentice Hall, New York, 1994.

[12] G. Horváth and T. Szabó, CMAC Neural Network with Improved Generalization Property for System Modelling, Proc. of the IEEE Instrumenation and Measurement Conference, Anchorage, 2002.

[13] T. Szabóand G. Horváth, CMAC and its Extensions for Efficient System Modelling and Diagnosis, Intnl. Journal of Applied Mathematics and Computer Science, Vol. 9. No. 3, pp.571-598, 1999.

[14] J. Hertz, A. Krogh and R. G. Palmer, Introduction to the Theory of Neural Computation, Addison-Wesley Publishing Co. 1991.

[15] K. S. Narendra and K. Pathasarathy, Identification and Control of Dynamical Systems Using Neural Networks, IEEE Trans. Neural Networks, Vol. 1. 1990.

[16] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson and A. Juditsky: "Non-linear black-box modeling in system identification: a unified overview", Automatica, 31:1691-1724, 1995.

[17] E. A. Wan, Temporal Backpropagation for FIR Neural Networks, Proc. of the 1990 IJCNN, Vol. I. pp. 575-580.

[18] R. J. Williams and D. Zipser, A Learning Algorithm for Continually Running Fully

Recurrent Neural Networks, Neural Computation, Vol. 1. 1989. pp. 270-280.

[19] N. Murata, S. Yoshizawa and Shun-Ichi Amari, Network Information Criterion - Determining the Number of Hidden Units for an Artificial Neural Network Model, IEEE Trans. on Neural Networks, Vol. 5. No. 6. Pp. 865-871.

[20] X. He and H. Asada, A New Method for Identifying Orders of Input-Output Models for Nonlinear Dynamic Systems, Proc. of the American Control Conference, 1993. San Francisco, CA. USA. pp. 2520-2523.

[21] S. Saarinen, B. Bramley and G. Cybenko, Ill-conditioning in Neural Network Training Problems, SIAM Journal for Scientific and Statistical Computing, 1991.

[22] G. Horváth, B. Pataki and Gy. Strausz, Black box Modeling of a Complex Industrial Process, Proc. of the 1999 IEEE Conference and Workshop on Engineering of Computer Based Systems, Nashville, TN, USA. 1999. pp. 60-66.

[23] J. Van Gorp, J. Schoukens and R. Pintelon, Learning Neural Networks with Noisy Inputs Using the Errors-In-Variables Approach, IEEE Trans. on Neural Networks, Vol. 11. No.2. pp. 402-414. 2000.