RECURRENT NEURAL NETWORK PREDICTORS FOR EEG SIGNAL COMPRESSION

F. Bartolini, V. Cappellini, S. Nerozzi

Dipartimento di Ingegneria Elettronica,
Università di Firenze,
via S. Marta 3, 50139, Firenze, Italy,
barto@cosimo.ing.unifi.it

A. Mecocci

Dipartimento di Elettronica,
Università di Pavia,
via Abbiategrasso, 27100, Pavia, Italy

ABSTRACT

The progress of digital electroencephalography gave rise to the problem of EEG data recording. In this paper a DPCM scheme for EEG signal compression is discussed. In particular the performance of a class of predictors based on recurrent neural networks is presented. The training strategy is accurately described and the results of a comparison with some other classical linear and static neural predictors are given. The proposed recurrent neural predictor demonstrates to be competitive with the others in offering good performance at a very low computational cost.

1. INTRODUCTION

Digital electroencephalography (EEG) produces a large amount of data that has to be recorded for future clinical uses. EEG signals have to be picked up in many pairs of points of the scalp and for a long time (at least 20 minutes). The huge quantity of resulting digital data could be reduced by means of signal compression techniques.

In [1] e vector quantization approach is proposed, it offers high compression rates and is suitable for the classification of typical graphoelements but does not behave well when high frequency components are predominant in the signal. In [2] a zero order predictor is used, the prediction error is adaptively quantized. The quantizer is designed to minimize the distortion by assuming a given probability distribution for the error signal, which is periodically scaled to fit into the quantizer range. Medics usually require data compression be lossless: a DPCM approach, which permits the control of the peak reconstruction error, seems then to be suitable for the EEG compression (by imposing a null peak error the method is lossless). In [3] the results of the comparison among different predictors for a DPCM scheme are presented. Linear adaptive and non–adaptive predictors as well as feed–forward neural networks are tested.

In a DPCM scheme (see Figure 1) the present signal sample is predicted based on the past ones; the prediction error is coded by entropic techniques (Huffmann and Arithmetic coding) [4]. The results of comparing the performance of predictors based on dynamic neural networks [5, 6] with that obtained with predictors based on classical linear adaptive schemes (ADPCM) and on static feedforward neural networks [3] are exposed. The compression ratio which can be achieved by coding the resulting prediction error is also reported.

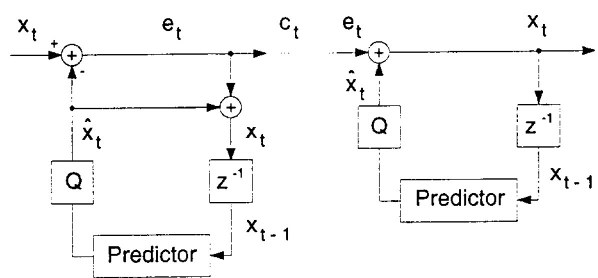

Figure 1: Outline of the DPCM lossless compression-decompression scheme.

2. RECURRENT NEURAL NETWORKS

A dynamic neural network is constituted by a unique layer of nodes: all the nodes have a feedback with unitary delay connected to all the inputs. The set of the output values at a given instant is called *state* of the network and it enhances the set of inputs used to com-

0-7803-2431-5/95 $4.00 © 1995 IEEE

pute the outputs at the successive instant. A subset of the nodes is the actual output of the network, the rest of the nodes only contributes to the state. It is possible to demonstrate that feedforward neural networks are a particular case of these dynamic networks. The presence of the feedback makes dynamic networks quite suitable for time series processing [7]. For training the dynamic neural predictors the Real Time Recurrent Learning—*RTRL*—rule proposed by Williams and Zipser [5, 6, 8] is used. It belongs to the class of the supervised learning algorithms. The training examples are constituted by segments of the input signal of length T and by the corresponding desired outputs. This sequence of input-output pairs is called *RUN*. In our case the *RUN* is made up of a segment of EEG signal. The *RTRL* algorithm search for the set of neural weights minimizing the Mean Square Error—*MSE*—over every run. The number of products required by the algorithm increases as the fourth power of the number of nodes: the training time sensitively grows with network dimensions [5].

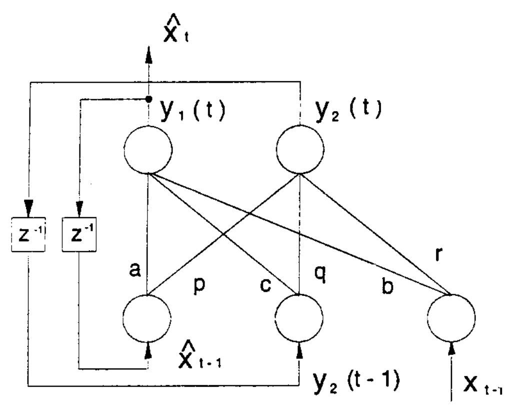

Figure 2: Predictor based on a 2 nodes dynamic neural network.

In Figure 2 a predictor based on a dynamic neural network is depicted, its input–output rule, producing the prediction value, is:

$$\hat{x}_t = f[a\hat{x}_{t-1} + bx_{t-1} + cy_2(t-1)] \tag{1}$$

and by simply rearranging the arguments of the function $f[\,]$

$$\hat{x}_t = f[\alpha x_{t-1} + \beta e_{t-1} + cf[\gamma x_{t-2} + \delta e_{t-2} + qy_2(t-2)]]. \tag{2}$$

In this last formulation it is evident the direct dependency of the predicted present sample on the previous true sample and on the previous prediction error value; there is also the dependency on the same quantities with higher delays in a secondary activation function. The y_2 node causes the memory of the predictor to be theoretically infinite. The use of the previous prediction error values in the prediction formula corresponds to model the signal as a Nonlinear Auto Regressive Moving Average—*NARMA*—process: a non linear model is very suitable given the nature of the EEG signal.

3. THE TRAINING STRATEGY

The training strategy must be accurately designed: if a *RUN* for each EEG graphoelement (alpha rythms, epileptic activity, artifacts, etc.) is built the network is constrained to learn many drastically different models of signal, presented one after the other, and it is unable to memorize a model suitable for all the graphoelements. For this reason a unique *RUN* including all the characteristic graphoelements is created.

Various experimental tests have been performed for finding the best *cocktail* of graphoelements to be chosen for composing the most representative *RUN*. They demonstrated that networks trained with a large number of graphoelements perform best, being able to generalize their behaviour by also predicting efficiently EEG segments never learned. Nevertheless it has to be observed that some particular graphoelements, such as for example the muscular artifacts, do not help learning, on the contrary their presence in the *RUN* seems to confuse the network, worsing its behaviour on the other tracts of the signal too. These graphoelements, that differ from the others both in their statistic and in their range, must then not be included in the training set.

An attempt was also made to enhance the supervision by forcing the network to reconstruct the previous sample or to predict the future one. In the former case the moving average component in the primary activation function is increased. In the latter case the network is prepared, at the present step, to predict the future sample whose estimation will be already present in the state vector at the next step: only a refinement of this estimation should then be required. These two enhancements did not produce the expected results.

Networks with many nodes or many inputs learn very fast to predict segments of alpha rythms (which are very correlated) but reduce their generalization capabilities over the other typical graphoelements.

4. EXPERIMENTAL RESULTS

The parameter used to evaluate the performance of the predictors the variance of the prediction error is evaluated. Networks of different size have been tested; those with a few nodes and a few inputs result to be much more effective: in particular the network with 1 node and 2 inputs and that one with 2 nodes and 1 input were the best.

By comparing the prediction performance of these best dynamic networks with the performance of a static neural predictor and of classical adaptive linear predictors [3] it resulted that dynamic networks highly outperform static networks and some adaptive linear predictors, but are less efficient then the best of the adaptive linear systems (the one using the gradient algorithm for coefficients updating). In Figure 3 the ratio between the variance of the original signals and that of the prediction error obtained with some different predictors is depicted. The values are reported separately for the group of signals from which the training set was extracted (**Training**) and for the rest of signal tracks (**Testing**).

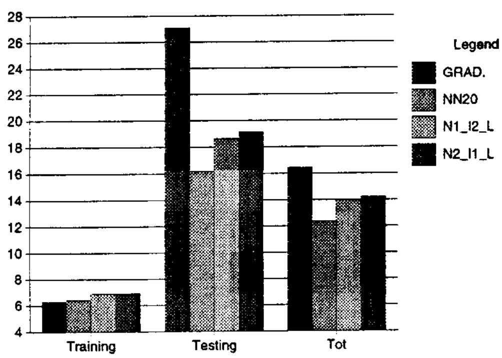

Figure 3: Ratio between the variance of the original signal and that of the prediction error for some predictors.

From the point of view of computational complexity the dynamic networks are much simpler than the static ones and the network with 1 node and 2 inputs result less demanding than all the adaptive predictors. Neural networks require the computation of a non–linear function which can be achieved by means of look–up tables. In Figure 4 a comparison of complexity—in term of sums, products and non–lineriities— among the above mentioned predictors is graphically represented.

The prediction error coding has been done by variable length codes: in particular the Huffman and the

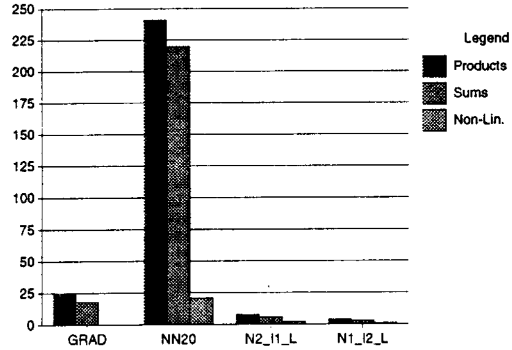

Figure 4: Evaluation of computational complexity of the predictors.

Arithmetic code have been tested [4]. For both the algorithms static and adaptive versions have been implemented, for the Arithmetic code source models with memory have also been used.

In Figure 5 the percentages of the occupancy of the compressed signals with respect to the original ones are reported for the case of Huffman coding: **ORIG.** is the original signal directly coded with the Huffman code, **GRAD.** is the compressed signal obtained with the adaptive predictor, **N2_I1_L** and **N1_I2_L** are the signals obtained with the dynamic networks with respectively 2 nodes/1 input and 1 node/2 inputs. The adaptive technique performs better than the neural predictors being able to track the variations in the statistic of the signal with higher accuracy.

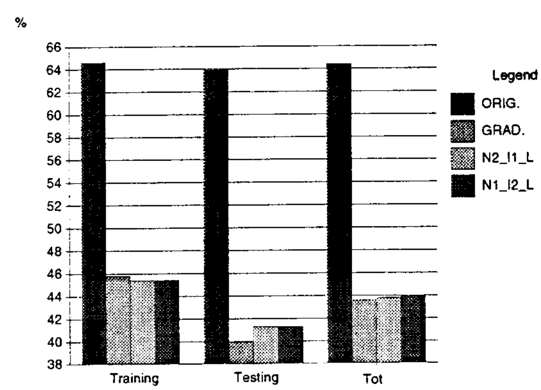

Figure 5: Percentage of occupancy of compressed signals with respect to the original one: **adaptive Huffman** coding.

In Figure 6 the best experimental results are reported: they are obtained by using an Arithmetic code which models the signal as a memory 1 process. It is worth noting that in this case the neural predictors give better performance than the adaptive one: this is caused by the fact that a residual correlation remains in the prediction error signal obtained with the neural networks, this residual correlation is well modeled by this Arithmetic coder permitting it to work at its best: a powerful cooperation between the predictor and the coder is then obtained in this case.

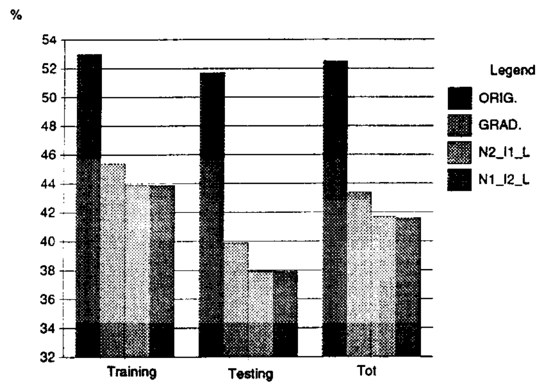

Figure 6: Percentage of occupancy of compressed signals with respect to the original one: **adaptive order 1 Arithmetic** coding.

In conclusion experimental results suggest that simple dynamic networks (the simplest one has only 1 processing node) produce good results in predicting EEG signals and can lead to the highest compression ratio in cooperation with a suitable coding algorithm.

5. REFERENCES

[1] S. Dimolitas, P.F. Lister, "Waveform preserving encoders for the compression of the EEG," *Signal Processing*, vol. 10, 1986, pp. 439–454.

[2] H. Hinrichs, "EEG data compression with source coding techniques," *Journal Biomedical Engineering*, vol. 13, September 1991, pp. 417–423.

[3] S. Bargiotti, F. Bartolini, V. Cappellini and A. Mecocci, "EEG signal compression based on ADPCM and neural network predictors," *Proceedings 14me Colloque GRETSI,* September 13-16, 1993, Juan Les Pins, France, pp. 1287-1290.

[4] M. Nelson, **The data compression book**, M & T Books, Redwood City, A, USA, 1991.

[5] D.R. Hush and B.G. Horne, "Progress in supervised neural networks (what's new since Lippmann?)," *IEEE Signal Processing Magazine,* January 1993, pp. 8-39.

[6] R. Hecht-Nielsen, **Neurocomputing**, *Addison-Wesley Inc.,* Menlo Park, MA, USA, 1991.

[7] J. Connor and L. Atlas, "Recurrent neural networks and time series prediction," *Proceedings of International Joint Conference on Neural Networks* July 8-12, 1991, Seattle, WA, USA, pp. 301-306

[8] T. Catfolis, "A method for improving the real-time recurrent learning algorithm," *Neural Networks,* vol. 6, 1993, pp. 807-821.