Correlative Training and Recurrent Network Automata for Speech Recognition

Roberto GEMELLO, Dario ALBESANO, Franco MANA
CSELT - Centro Studi e Laboratori Telecomunicazioni
via G. Reiss Romoli, 274 - 10148 Torino - Italia
Tel.: +39-11-2286224 Fax: +39-11-2286207 email: gemello@cselt.stet.it

Abstract

Discriminative training is one of the more distinctive features of Multilayer Perceptron networks when used as classifiers. Although, when dealing with overlapping classes, it may be useful to smooth this feature not compelling the MLP to discriminate where it is impossible. This can be done adaptively, without any prior information about the classes by introducing a straightforward modification of backprop we have named Correlative Training. This new MLP feature has proved to be very useful when training the hybrid Recurrent Network Automata model for speech recognition.

1. Introduction

Neural Networks (NN) have, in the last years, found their place among speech recognition technologies as powerful adaptive non-linear classifiers [Morgan,1991].

As, at present, NN are not yet able to manage well time modelling, they are mainly employed in integration with Hidden Markov Models (HMM). This approach has been investigated by several research teams: [Franzini90,91] have introduced the Connectionist Viterbi Training to enhance HMM based connected digit recognition; [Austin92] has described Segmental Neural Networks for phonetic modelling; [Bourlard93] has proposed connectionist probability estimation to significantly improve a HMM based continuous speech recognition system.

Our contribution to this line was the introduction of Recurrent Network Automata (*) (RNA) [Albesano92,93] which integrates recurrent NN with HMM word modelling, showing the advantages which can be obtained by exploiting the joint contextual information of feedback hidden units and time delayed input.

Discriminative training is the main feature of Multi-layer Percepron (MLP) neural networks, when trained as classifiers. In the field of speech recognition this feature of MLP classifiers is particularly interesting as it represents an alternative to the Maximum Likelihood (ML) training usually employed for HMMs. In fact, while ML training characterises each word model separately, discriminative training tries to find the best separation between them.

However, in some cases, a pure discriminative training can be not advisable: when there are some quite overlapping classes the best policy seems to be smoothing the discriminative training for the correlated classes while maintaining it for the uncorrelated ones. Of course, overlapping is a continue property, and hard decisions are not well suited. To overcome this problem, that we have encountered while training the MLP component of RNAs for word recognition, we conceived and experimented a modification to standard discriminative training that we called *Correlative Training*.

Although it was studied to solve a specific problem, Correlative Training can be applied in all those cases where you are looking for a trade-off between discriminative and characterizing training. Furthermore, it results into a very simple modification to backprop equations.

2. The Recurrent Network Automata model

The RNA recognition model [Albesano92] is a hybrid HMM-NN model devoted to recognise sequential patterns. Each class is described in terms of a left-to-right automaton (with self loops) as in HMM, and the emission probability of the automata states are estimated by a Simple Recurrent Network [Elman88]. The transition probabilities among states are not considered. The RNA has an input window that comprises some contiguous frames of the sequence, one hidden layer with a self-feedback, and an output level where the activation of each unit estimates the probability of the input window to belong to an automaton state.

The hidden neuron dynamics is given by the equation:

$$y_i(t) = F(\Sigma_j w_{ij} x_j(t) + \Sigma_k w_{ik} y_k(t-1))$$

(*) patent pending

0-7803-1901-X/94 $4.00 ©1994 IEEE

where y_i is the activation of a hidden neuron, x_j is an input unit and F is the standard logistic function. The hidden neurons are also called *state neurons* because thanks to the self-feedback they can encode a contextual information about the sequence which is being recognised. The output neurons follow the standard MLP dynamics.

3. Speech Modelling with Recurrent Network Automata

The RNA model was principally conceived for speech recognition, and in particular for modelling words for isolated or connected word recognition with a small vocabulary. In RNA time modelling takes place in two ways: first, by an external modelling, through the HMM like time warping ability of the dynamic programming applied to left-to-right automata corresponding to words; second, by the internal modelling of the recurrent network. In fact, the memory capability of the recurrent network allows to give the states a contextual information, inside the word automaton, and to give a more stable evolution of emission likelihoods, inside the state [Albesano *et al.*, 1992].

The architecture of RNA has many degrees of freedom: the architecture of the NN, the input window width, the number of automaton states for the different words of the vocabulary. A lot of experimental activity has been performed to optimise the architecture for the recognition of small vocabularies (10-20 words) resulting in the structure depicted in fig. 1. The input window is 3-7 frames wide, and each frame contains 26 parameters (log Energy, 12 Cepstral Coefficients, and their first derivatives). The first hidden layer is divided into three feature detectors blocks, one for the central frame, and two for the left and right context. Each block is in its turn divided into four sub-blocks devoted to keep into account the four types of different input parameters. It was empirically found that this a priori structure is generally better than a fully connected layer. The second hidden layer has a fully connected recurrence, like in Elman's nets (the double arrow means a copy of activation values). The neurons of this layer have a twofold function: first, they represent, together with the first hidden layer neurons, a *space transform* between the input parameters and some self-organised internal features, corresponding to acoustic/phonetic characteristics (e.g. silence, stationary sounds, transitions, specific phonemes). Besides, they encode a *state information* related to the temporal context the current input is inserted in, as described in [Albesano *et al.*, 1993]. The output layer estimates the emission probabilities of the states of the word automata, and is virtually divided in several parts, each one corresponding to an automaton.

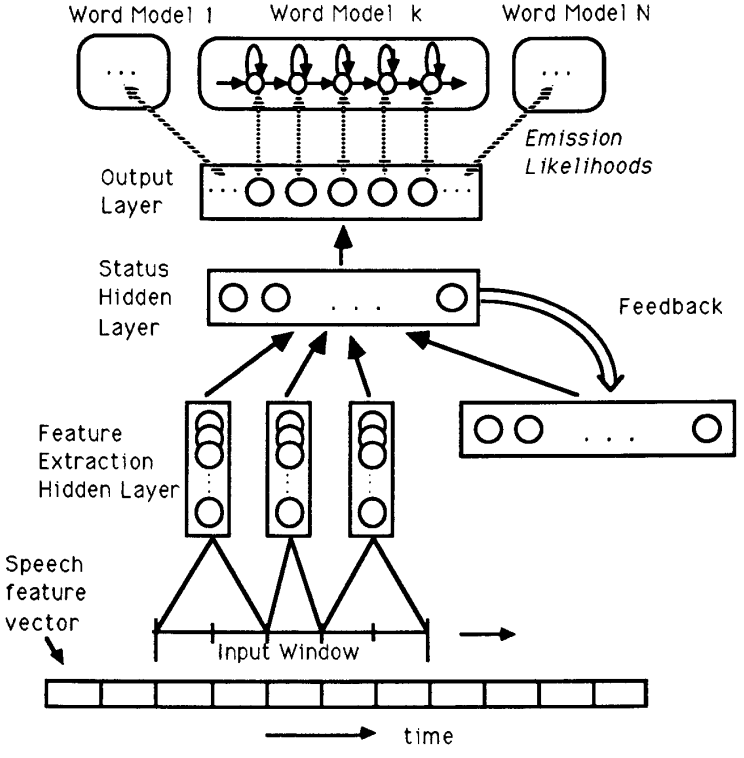

Figure 1: Architecture of a Recurrent Network Automata devoted to word recognition

Typical dimensions for a RNA devoted to recognise the ten Italian digits are:

- 7 frame input window;
- first hidden layer: central block with 24 units (divided in 2+10+2+10 for the four types of parameters), context blocks with 36 units (3+15+3+15).
- second hidden layer: 70 units, with fully connected recurrence.
- output layer: 63 units, corresponding to 63 automata states, pertaining to the 10 word automata, and divided proportionally to the average word length.

4. Correlative Training

4.1 Basic training algorithm

Recurrent Network Automata are trained using an iterative procedure as follows:

Initialisation:

- initialise the RNA with small random weights;
- create the first segmentation by segmenting the training utterances uniformly.

Iterations:

- load the present segmentation;
- train the RNA some epochs to implement the automata which approximates that segmentation;
- obtain a new segmentation by applying the dynamic programming to each utterance in the training set to re-evaluate the transition points proposed by the RNA;
- update the present segmentation by using a function of itself and of the new segmentation: present_segm = F(present_segm, new_segm);
  e.g. $F(s1, s2) = \alpha s1 + (1-\alpha)s2$, with α starting from 1.0 and decreasing during the training.

The input to the RNA is a window sliding on the speech frames, including a central frame and some left and right context frames. The targets are generated according to the present segmentation, putting 1.0 for the active state of the right automaton and 0.0 otherwise. All the automata are trained into a unique net, so performing a discriminative training. The NN basic learning algorithm is the back-propagation.

4.2 Correlative Backpropagation

The training outlined in the previous section is a discriminative training, as usual for MLP networks for classifications. With this kind of training, if two classes C1 and C2 happen to be superimposed, that is the patterns of C1 and the patterns of C2 are not separable, the estimated probability for a generic pattern of C1 or C2 will be only determined by the a priori probability of the class. In fact, as MLP nets used as classifiers were proved to estimate the posterior probability of classes [Bourlard90,93], we have:

$$p(C|X) = \frac{p(X|C)p(C)}{p(X)} \quad \text{with } C = C1, C2$$

but in this case $p(X|C1) = p(X|C2)$, as the two classes C1 and C2 are superimposed, $p(X)$ is a constant, and the only different factors are the a priori probabilities $p(C1)$ and $p(C2)$. Obviously that fact is unacceptable in a recognition system and has to be corrected.

In speech recognition with entire word models, this situation may happen in some cases and with different degrees. For example, as the Italian digits SEI and SETTE (six and seven) begins in the same way, it is foreseeable that the first 2-3 states of the two automata will be associated to overlapping sets of patterns.

To face that cases it is advisable to modify the standard discriminative training in such a way that discriminable classes are trained with a discriminative training, while for non-discriminable classes the discriminative training is adaptively smoothed. That is what we call *Correlative Training*.

The problem of similar states in word automata was already encountered by Smyth [Smyth, 1992] in the recognition of the English alphabet. He started with 26 words and three output units per word, resulting in 78 distinct states, a significant number of these states corresponding to very similar sounds (i.e. the final units of the E-set and A-set). He managed the problem of units correlation by identifying during training (from the confusion matrix) the most confusable states, and clustering them. That method works, and has the advantage of reducing the number of output units. However, it is intrinsically discrete, as two states are clustered or are not. We think that in many cases this hard

decision is not advisable, and propose, with Correlative Training a continuous way of letting the natural correlations among output states to emerge.

Correlative training has been developed for speech recognition, but its scope is beyond this application, including the cases in which it is convenient to adaptively tune the discriminative training.

Briefly, Correlative Training consists in changing the definition of the target in function of the correlation of the outputs of the considered unit and of the expected corrected unit.

Let us name:

o_k the output of output unit k;
t_k the target of output unit k;
h the index of the output unit with $t_h = 1.0$

We redefine the target of a generic output unit k as:

$$t_k(o_k, o_h) = \begin{cases} t_h & \text{if} \quad h = k \\ o_k o_h & \text{if} \quad h \neq k \end{cases}$$

where $o_k o_h$ has been chosen as a measure of the correlation of the output of the considered unit k and the output of the unit h the target of which is 1.0 .

This change in the definition of the target leads to a variation in the $\frac{\partial E}{\partial o_j}$ term of backpropagation for output units, that becomes

$$\frac{\partial E}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2} \sum_k (t_k(o_k, o_h) - o_k)^2$$

and deriving we obtain:

$$\frac{\partial E}{\partial o_j} = (t_j(o_j, o_h) - o_j)\left(\frac{\partial (t_j(o_j, o_h))}{\partial o_j} - 1\right) = \begin{cases} -(t_j - o_j) & \text{if} \quad j = h \\ o_j(o_h - 1)^2 & \text{if} \quad j \neq h \end{cases}$$

This change in backpropagation results in an adaptive smoothing of discriminative training. In fact if two output units o_m and o_n correspond to intrinsically similar acoustic phenomena they will try to respond with a high value on the same input patterns. The correlation substitutes the zero target for the "wrong" unit with the correlation of the two outputs $o_m o_n$, so allowing gradually the two units to respond both with higher and higher values on that input patterns. On the contrary, if two output units o_m and o_n respond with different values (one high and the other low) the product $o_m o_n$,will be low (~0.0) and the correlative training will behave as a regular discriminative training.

5. Recognition Experiments

We used RNA trained with correlative training to face a difficult real problem, i.e. the speaker independent recognition of the digits over the public telephone network. This problem has been already faced in our labs by using Continuous Density Hidden Markov Models (CDHMM) [Canavesio91], so we already have a large training database and some state of the art results to compare with. Preliminary results obtained using RNA and a comparison with HMMs were reported in [Albesano, 1993].

The speech database we used was collected on the Italian public telephone network, each time using a different switching circuit. It is suited for speaker independent training as about 1,000 people evenly distributed between male and female voices contributed to it. The pre-processing technique consists of a Mel-based spectral analysis followed by a Discrete Cosine Transform [Gemello90] to obtain Cepstral coefficients. Together with the cepstral coefficients, the value of the logarithm of the total energy of each frame is retained as it provides some information about distinguishing the voiced parts of the speech input from the unvoiced ones.

A RNA network was trained on a training set containing about 500 repetition for each digit. Several architectures were experimented, with and without feedback. The feedback always improved

the performances, both in recognition percentage and in the average distance between the correct model and the second one. Presently, the best RNA architecture has an input window of 7 frames, each containing 26 features (12 Cepstral, Energy, 12 Cepstral derivatives, Energy derivative), a first hidden layer with 96 units, windowed both on time and on features, as described in Fig. 1, a second hidden layer of 70 units, a feedback layer of 70 units and one output layer of 63 units (one for each state of the automata modelling the ten digits). The obtained results are exposed in Table 1.

Architecture	Training	Test
w=3, 2 hidden layers + feedback (Dec. '92)	98.7	98.4
w=7, 2 hidden layer + feedback , as in Fig. 1 (Dec. '93)	99.7	98.7
w=7, 2 hidden layer + feedback, 1 state for word	99.1	97.0

Table 1. - Recognition results of RNA trained with Correlative Training

Correlative training substantially improved the convergence of RNAs and the recognition results, and was used in all the experiment reported here.

The first result with a input window of 3 frames, (98.4) was already reported in [Albesano, 1993] . Using a window of 7 frames the result was enhanced to 98.7.

In the first two experiments the word automata are modelled with a number of states ranging from 5 for short words to 9 for long words. The last experiment, on the contrary, makes use of just one state to model each word: in this way no external modelling is provided, and all the time modelling is made internally by the recurrent net. The obtained results (97.0) are inferior, but not so bad, and outline the possibility, for small vocabularies, of using directly a recurrent network with one output for each word, so avoiding the dynamic programming decoding of automata.

6. Conclusion

Correlative Training has been described as a simple way to soften in an adaptive way the strength of discriminative training for classes that cannot be completely put apart, without compromising its power on separable classes. The correlations which naturally emerge between classes during training are considered, allowing them to gradually arise without disturbing the training process.

Correlative training was applied to the training of RNA for word recognition, allowing an effective training and performances which get over those of continuous HMMs on small vocabulary word recognition.

References

[Albesano92] D. Albesano, R. Gemello and F. Mana, "Word Recognition with Recurrent Network Automata", in *Proc. IJCNN 92*, Baltimore, June 1992, pp. 308-313.

[Albesano93] D. Albesano, R. Gemello and F. Mana, "Recurrent Network Automata for Speech Recognition", in *Proc. WCNN 93*, Portland, July 1993, vol. III, pp. 16-19.

[Austin92] S. Austin, G. Zavaliagkost, J. Makhoul, and R. Schwartz, "Speech Recognition using Segmental Neural Nets", in *Proc. ICASSP*, 1992, pp. 625-628.

[Bourlard90] H. Bourlard, C.J. Wellekens, "Links Between Markov Models and Multilayer Perceptrons", in IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 12, pp. 1167-1178.

[Bourlard93] H. Bourlard, N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach,* Kluwer Academic Publishers, 1993.

[Elman88] J.L. Elman, "Finding Structure in Time", CRL Technical Report #8801, University of California, San Diego, 1988.

[Canavesio91] F. Canavesio, L. Fissore, M. Oreglia, R. Ruscitti "HMM modeling in the public telephone network environment: experiments and results", in *Proc. EUROSPEECH 91*, Genova, September 1991, pp. 731-734.

[Franzini90] M.A. Franzini, K.F. Lee and A. Waibel, "Connectionist Viterbi Training: A new hybrid method for continuous speech recognition", in *Proc. ICASSP*, Albuquerque,NM, April 1990, pp.425-428.

[Franzini91] M.A. Franzini, A. Waibel and K.F. Lee , "Continuous Speech Recognition with the Connectionist Viterbi Training Procedure: a summary of recent work", in *Proc. IJCNN* , Singapore, 1991, pp.1855-1860.

[Haffner91] P. Haffner, M. Franzini, A. Waibel, "Integrating Time Alignment and Neural Networks for High performance Continuous Speech Recognition", in *Proc. ICASSP*, 1991, pp. 105-108.

[Li92] K.P. Li, J.A. Naylor, and M.L. Rossen, " A Whole Recurrent Neural Network for Keyword Spotting", in *Proc. ICASSP*, 1992, pp. 81-84.

[Morgan91] D.P. Morgan and C.L. Scofield, *Neural Networks and Speech Processing,* Kluwer Academic Publishers, 1991.

[Smyth92] S.G. Smyth, "Segmental sub-word unit classification using a multilayer perceptron", in R. Linggard, D.J. Myers, C. Nightingale editors *Neural Networks for Vision, Speech and Natural Language,* Chapman & Hall, 1992.