USING GAMMA FILTERS TO MODEL TEMPORAL DEPENDENCIES IN SPEECH

Steve Renals and Mike Hochberg
Cambridge University Engineering Department
Cambridge CB2 1PZ, UK
{sjr,mmh}@eng.cam.ac.uk

ABSTRACT

Hybrid systems which use connectionist networks to estimate the output probabilities of a hidden Markov model represent time both at the network level and the Markov chain level. In this paper we discuss modelling time in connectionist networks, and introduce local recurrences in a feed-forward network in the form of an adaptive gamma filter. Using the Resource Management (RM) database, we have performed continuous speech recognition experiments comparing a gamma filtered input representation to a delay line. We have also performed speaker adaptation experiments using the speaker-dependent RM database. Our results have not indicated that gamma filters offer an appreciable modelling advantage on this task. However, the baseline speaker adaptation experiments have indicated that supervised adaptation over 100 sentences reduced the word error by an average of 40%.

1. INTRODUCTION

Hybrid connectionist/hidden Markov model (HMM) systems model time at two levels, although these levels are not necessarily at different time scales. As is usual in HMM systems, a Markov process is used to specify durational, lexical and grammatical constraints on the model. However, a second level of temporal modelling is provided by a connectionist network which is used to estimate posterior probabilities of states of the Markov process conditioned on the acoustic data. Two basic types of network that have been used in this framework are *recurrent* networks [1] and *feed-forward* networks [2].

Recurrent networks (RNs)—-which may be regarded as nonlinear IIR filters—can theoretically model dynamics of arbitrary complexity. This is limited, however, by the availability of sufficiently powerful training algorithms to learn long-term dependencies. Feed-forward networks incorporating delay lines—which may be regarded as FIR filters—account for a finite amount of recent acoustic context. Clearly, the recurrent network is a more powerful temporal model than a feed-forward network. However, comparative results using the 1,000 word ARPA Resource Management (RM) continuous speech recognition task indicate that the two models result in similar word recognition performance [3].

In this paper we discuss the temporal properties of the connectionist models under investigation using the twin notions of *depth* and *resolution*. Following Principe et al. [4] we use a convolution model to generalise the delay line and discuss the gamma filter architecture to model input time dependence. We apply this architecture

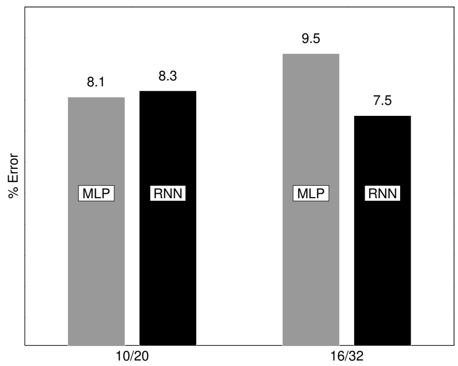

Figure 1. Comparison of acoustic front ends using an MLP and a recurrent network on the Resource Management task combined test sets (Feb 89, Oct 89, Feb 91, Sep 92). The front ends computed MFCCs at higher resolution (10/20) or lower resolution (16/32) frame rates.

in our feed-forward connectionist/HMM speech recognition system, comparing performance with the delay lined network on the 1,000 word Resource Management (RM) task.

Because the gamma filter is used at the input to the feed-forward system, we have speculated that the gamma filter approach may be well suited to speaker adaptation [5]. In section 3.2 we report on a set of experiments to test this hypothesis.

2. REPRESENTING TEMPORAL DEPENDENCE

2.1. Depth and Resolution

Following Principe et al. [4], we may characterise the time dependence displayed by a particular model in terms of *depth* and *resolution*. Loosely speaking, the depth tells us how far back in time a model is able to look[1], and the resolution tells us how accurately the past to a given depth may be reconstructed. The baseline models that we currently use are very different in terms of these characteristics.

2.2. Experimental

Previous experiments on the speaker independent RM database have indicated that the tradeoff between depth

[1]In the language of section 2.4, the depth may be expressed as the mean duration, relative to the target, of the last kernel in a filter that is convolved with the input.

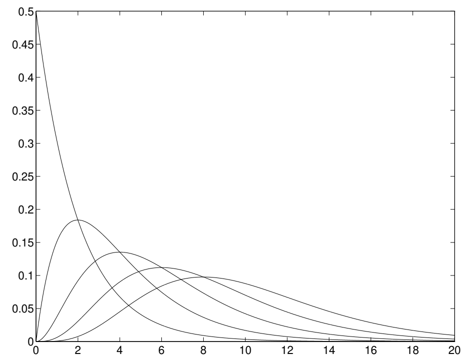

Figure 2. Gamma filter kernels for $\mu = 0.5$.

and resolution is important. We compared the performance of a multi-layer perceptron (MLP) and a RN [3], using front ends of different resolution. The two front ends computed mel-frequency cepstral coefficients (MFCCs): one with a 20ms Hamming window and a 10ms frame step (referred to as 10/20), the other with a 32ms Hamming window and a 16ms frame step (referred to as 16/32). *A priori*, we expected the higher resolution frame rate (10/20) to produce a higher performance recogniser because rapid speech events would be more accurately modelled. While this was the case for the MLP, the RN showed better results using the lower resolution front end (16/32) (see figure 1). For the higher resolution front-end, both models require a greater depth (in frames) for the same context (in milliseconds). In these experiments the network architectures were constant so increasing the resolution of the front end results in a loss of depth.

In the case of the MLP we were able to explicitly set the memory depth. Previous experiments had determined that a memory depth of 7 frames (together with a target delayed by 3 frames) was adequate for problems relating to this database. In the case of the RN, memory depth is not determined directly, but results from the interaction between the network architecture (*i.e.*, number of state units) and the training process (in this case, back-propagation through time). We hypothesise that the RN failed to make use of the higher resolution front end because it did not adapt to the required depth.

2.3. Convolution Model

A general way of representing time dependence uses a *convolution model*, in which the input x is convolved with a set of filter kernels parameterised by $\mathbf{w}$, i.e.,

$$y(t) = \sum_{n=0}^{t} w(t-n)x(n). \tag{1}$$

Note that in this general, unconstrained form the number of parameters increases linearly with filter depth. If the filter kernels are constrained to be delta functions, then we have a delay line:

$$w(t) = \sum_{k} w_k \delta(t - t_k)\ . \tag{2}$$

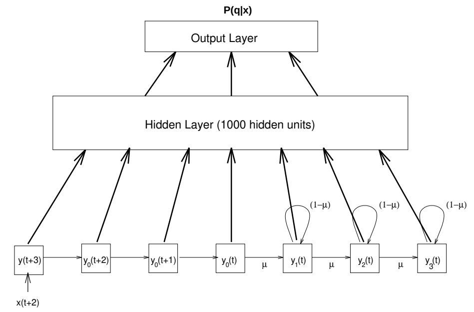

Figure 3. Gamma memory applied to the network input. The simple gamma memory does not incorporate any information about the future, unless the target is delayed. Here there is an explicit delay line to incorporate some future context.

This choice of kernel function also saves in computation when computing the convolution.

However, a less constrained set of filter kernels could be adapted to give the depth and resolution that best describes the training set. One suggested kernel function for this task is a Gaussian [6]:

$$w(t) = \sum_{k} w_k \mathcal{N}(t - t_k; \mu, \Sigma) \tag{3}$$

The disadvantage of this choice is that the convolution between the input data and kernel functions must be explicitly computed at each timestep.

2.4. Gamma Filters

Principe et al. [4] suggested using the gamma density function (or its discrete time analog, the negative binomial):

$$w(t) = \sum_{k} w_k \frac{\mu^k}{(k-1)!} t^{k-1} e^{-\mu t}\ . \tag{4}$$

These kernel functions are illustrated in fig. 2. This *gamma filter* model is attractive since it allows adaptive depth and resolution (controlled by the number of filter taps and the parameter μ) and may be updated recursively, without explicitly computing the convolution at each time:

$$y_k(t) = (1-\mu)y_k(t-1) + \mu y_{k-1}(t-1)\ . \tag{5}$$

A gamma filter may be represented as a locally recurrent network (see fig. 3). Although this is much more constrained than a fully recurrent network, there are simple stability conditions ($0 < \mu < 2$), and such local recurrences may capture the necessary time dependence in some problems.

3. EXPERIMENTS

Experiments were carried out using the RM database. The acoustic data was preprocessed using a 12th order perceptual linear prediction (PLP) analysis to produce an energy coefficient plus 12 PLP cepstral coefficients for each frame of data. A 20ms Hamming window was

Input Repn.	Depth	Test Set	Ins.%	Del.%	Sub.%	Error%
Delay Line	4.0	Oct 89	0.6	1.4	4.1	6.1
		Feb 91	1.3	1.0	4.1	6.4
		Sep 92	2.3	2.3	7.1	11.8
Gamma Filter	9.7	Oct 89	0.9	1.5	4.6	7.0
		Feb 91	0.9	1.0	4.5	6.4
		Sep 92	2.1	2.4	6.8	11.3

Table 1. Speaker-independent RM results using delay lined and gamma filtered inputs. Both systems used 3 frames of future context and a 4 tap delay line or gamma filter. Three test sets were used, each containing 300 sentences, 30 spoken by each of ten new speakers. The depth of the gamma filter was estimated as the ratio of filter order to average filter parameter, $K/\overline{\mu}$.

used with a 10ms frame step. The temporal derivatives of each of these features was also estimated (using a linear regression over ± 3 adjacent frames) giving a total of 26 features per frame.

The networks we employed were MLPs, with 1000 hidden units and 68 output units (one per phone). The input representations used were a delay lined representation with ± 3 frames of context and a four tap gamma filter per input feature, with an additional three frames of future context implemented as a delay line (see fig. 3). This gamma filter architecture was chosen following earlier experiments on the TIMIT database [5]. The Markov process used single state phone models with a minimum duration constraint, a word-pair grammar, and a Viterbi decoder was used for recognition. The gamma filter coefficients (one for each channel of the feature vector) were initialised to 1.0 (equivalent to a delay line); when training, these coefficients were not adapted during the first iteration through the data. The feed-forward weights were trained using back-propagation and the gamma filter coefficients were trained in a forward in time back-propagation procedure equivalent to real-time recurrent learning [7]. An important detail is that the gradient step size was substantially lower (by a factor of 10) for the gamma filter parameters compared with the feed-forward weights. This was necessary to prevent the gamma filter parameters from becoming unstable.

3.1. Speaker Independent Recognition

In the first experiment, the adaptive gamma filtered input was compared with a delay lined input using the same number of filter taps (and the same amount of future context). The networks were trained using the RM speaker independent training set (3990 sentences) and using the the Feb 1989 test set as a validation/development set. Tests were then carried out using the Oct 89, Feb 91 and Sep 92 test sets. Results are given in table 1.

When training the locally recurrent gamma filter network, the frames within a sentence must be presented sequentially. The delay lined network was also trained in this fashion, to allow better comparison of results, although previous work has shown that stochastic training (random presentation of frames) results in improved performance and reduced training times [8].

These experiments do not indicate that the gamma filter approach offers any modelling advantage when applied to this task; indeed the performance of a stochasticly trained MLP would be 10–20% better than that of the sequentially trained MLP used in this study. However, training the locally recurrent coefficients of the gamma filter is not a trivial process, and it may be that better results would be produced by a more adequate training process.

3.2. Speaker Adaptation

The previous section indicated that the gamma filter architecture did not offer a consistent improvement over the delay line. However, we have hypothesised that the gamma filter input representation might be a succinct way of capturing speaker characteristic information. Consequently, we have carried out some speaker adaptation experiments.

In these experiments a baseline network used above was adapted using data obtained from the speaker-dependent portion RM database. Data from all twelve speakers were used. The 100 sentence development set for each speaker was used for supervised adaptation (80 training, 20 cross-validation), and testing was done on a separate 100 sentence evaluation set. Four sets of adaptation experiments were carried out. In one set, acting as a baseline adaptation, all the weights of the delay line network were adapted using the supervised adaptation set. Two analogous adaptations using the gamma filter net were carried out adapting all the weights in one case and all the weights excluding the gamma filter coefficients in the other. However the principal experiment involved adapting the gamma filter coefficients alone, on the assumption that this process might encode speaker-dependent information such as rate. The results are shown in table 2. Again, it should be noted that the delay lined network was trained sequentially, so the recognition performance is poorer than that of a stochasticly trained MLP.

We speculated that adaptation of the gamma filter coefficients alone, would be a robust, efficient approach, since only 26 parameters are being updated. However, this adaptation requires backpropping through the MLP weights, and thus 2/3 of the computation of adapting all the parameters. Additionally the error signal is attenuated and less well estimated at the input layer, compared with the output layer. Indeed, adapting the gamma filter coefficients alone had very little effect. This implies that a simple speaker adaptation scheme based on gamma filters is unlikely to work well in practice. However, adapting all the parameters of the baseline networks was effective resulting in an average error decrease of 40% over the twelve speakers—and there was an improvement in performance for all speakers.

Cross-validation proved to be important in adaptation. In the case of the delay-lined networks, from three to twelve adaptation iterations were required. The number was chosen using the performance on the cross-validation set. There was no obvious relationship between the number of adaptation iterations required and

Speaker	Word Error %						
	Baseline		Adapted				
	Delay	Gamma	Delay (all)	Gamma (μ)	Gamma (all)	Gamma (all bar μ)
bef0_3	7.6	7.7	5.6	7.8	6.0	6.1
cmr0_2	11.1	10.4	5.6	11.9	6.5	6.6
das1_2	7.2	7.4	2.9	7.4	3.4	3.2
dms0_4	5.9	6.5	4.9	6.5	4.2	4.1
dtb0_3	5.7	6.6	5.5	6.5	6.0	6.0
dtd0_5	11.0	10.3	5.6	10.1	5.9	5.9
ers0_7	5.7	7.7	5.1	7.6	6.5	6.5
hxs0_6	12.4	11.2	5.3	10.9	4.9	4.7
jws0_4	5.5	5.9	2.9	5.9	2.8	2.7
pgh0_1	5.0	4.8	3.5	4.7	3.3	3.6
rkm0_5	16.5	17.6	6.8	17.5	7.2	6.8
tab0_7	4.7	5.0	3.6	4.9	3.9	3.6
average	8.2	8.4	4.9	8.5	5.0	5.0

Table 2. Speaker adaptation results using the 12 RM speaker dependent speakers, starting from a trained speaker independent network. Each speaker was adapted using a 100 sentence development set, and then tested using a 100 sentence evaluation set.

the resultant improvement in performance. It may be helpful to regard this adaptation approach as a regularisation, where the prior on the weights is given by the speaker independent weight matrix, and the influence of the prior is controlled by the cross-validation set.

4. CONCLUSION

Several conclusions may be drawn from this study:

1. Adaptive gamma filters have not given a modelling advantage in terms of recognition performance on this continuous speech recognition task.
2. The speculation that gamma filters may succinctly capture speaker characteristics has not been borne out by speaker adaptation experiments.
3. Failure of the gamma filter approach is not necessarily an architectural problem, but may be related to the training algorithm and learning dynamics.
4. Supervised speaker adaptation performed by adapting all the weights in a feed-forward network is effective, reducing the word error rate by an average of 40% on the RM task, when using an adaptation set of 100 sentences.

Finally, we note that a possible application of the gamma filter approach is the enhancement of acoustic representations under noisy conditions, since the gamma filter effectively implements a low-pass ($\mu < 1$) or high-pass ($\mu > 1$) filter.

ACKNOWLEDGEMENTS

Steve Renals was supported by a SERC postdoctoral fellowship. This work was supported by ESPRIT BRA 6487, WERNICKE.

REFERENCES

[1] A. J. Robinson. The application of recurrent nets to phone probability estimation. *IEEE Transactions on Neural Networks*, 5:298–305, 1994.

[2] S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco. Connectionist probability estimators in HMM speech recognition. *IEEE Transactions on Speech and Audio Processing*, 2:161–175, 1994.

[3] A. J. Robinson, L. Almeida, J.-M. Boite, H. Bourlard, F. Fallside, M. Hochberg, D. Kershaw, P. Kohn, Y. Konig, N. Morgan, J. P. Neto, S. Renals, M. Saerens, and C. Wooters. A neural network based, speaker independent, large vocabulary, continuous speech recognition system: the WERNICKE project. In *Proceedings European Conference on Speech Communication and Technology*, pages 1941–1944, Berlin, 1993.

[4] J. C. Principe, B. de Vries, and P. G. de Oliveira. The gamma filter—a new class of adaptive IIR filters with restricted feedback. *IEEE Transactions on Signal Processing*, 41:649–656, 1993.

[5] S. Renals, M. Hochberg, and T. Robinson. Learning temporal dependencies in large-scale connectionist speech recognition. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 1051–1058. Morgan Kaufmann, San Francisco, 1994.

[6] U. Bodenhausen and A. Waibel. The Tempo 2 algorithm: Adjusting time delays by supervised learning. In R. P. Lippmann, J. E. Moody, and D. S. Touretzky, editors, *Advances in Neural Information Processing Systems*, volume 3, pages 155–161. Morgan Kaufmann, San Mateo CA, 1991.

[7] R. J. Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1:270–280, 1989.

[8] S. Renals, N. Morgan, M. Cohen, and H. Franco. Connectionist probability estimation in the DECIPHER speech recognition system. In *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 601–604, San Francisco, 1992.