Recurrent Nets for Phone Probability Estimation

Tony Robinson
Cambridge University Engineering Department
Trumpington Street, Cambridge, CB2 1PZ, UK
ajr@eng.cam.ac.uk

Abstract

This paper presents an overview of the use of recurrent networks for phone probability estimation in large vocabulary speech recognition. The current system is described and recent recognition results on the TIMIT and Resource Management tasks in multiple-speaker mode are presented. Results on the speaker-independent Resource Management task are presented for the first time.

1 Introduction

1.1 Context in speech recognition

Context is very important in speech recognition at all levels. On a short time-scale, coarticulation influences the acoustic realisation of a phoneme. On longer time scales there are many slowly varying contextual variables (e.g. the degree and spectral characteristics of background noise, channel distortion) and speaker dependent characteristics (e.g. vocal tract length, speaking rate and dialect). To attempt speech recognition at the highest possible levels of performance means making efficient use of all of the contextual information.

Current Hidden Markov Model (HMM) technology approaches the problem from two directions: top down by considering phonetic context; and bottom up by considering acoustic context. The short-term contextual influence of coarticulation is handled by creating a model for all sufficiently distinct phonetic contexts. This entails a trade off between creating enough models for adequate coverage and maintaining enough training examples per context so that each model may be sufficiently trained. Clustering and smoothing techniques can enable a reasonable trade-off to be made at the expense of model accuracy and storage requirements. The problem remains of an exponentially increasing number of potential models in the number of contextual variables which limits the applicability of this technique.

Acoustic context is handled by increasing the dimensionality of the observation vector to include some parameterisation of the neighbouring acoustic vectors. This changes the problem to one of obtaining robust probability estimation from high dimensional spaces.

1.2 Probability estimation in speech recognition

Increasing the dimensionality of the acoustic vector increases the amount of contextual information available. The simplest way to do this is to replace the single frame of parameterised speech by a vector containing several adjacent frames along with the original central frame. However, this dimensionality expansion quickly results in difficulty in obtaining good models of the data. For example, Gaussian distributions of acoustic parameters are often assumed for each class, but for an n parameter set of acoustic vectors, $O(n^2)$ parameters in the covariance matrix must be estimated. This can be reduced by assuming subsets of the acoustics vectors are independent (block diagonal covariance matrix), or all acoustic parameters are independent (diagonal covariance matrix), however, this clearly limits the modelling power available.

Careful choice of the method used to increase the information content of the acoustic vector is clearly important. Empirically it has been shown that first (and second) order derivatives taken over a window length of a few frames are reasonable choice for the parameterisation of acoustic context and yield substantial improvements in speech recognition accuracy [2]. As a result this parameterisation has been widely adopted by the speech recognition community.

Difference coefficients are a simple linear function of the acoustic vectors lying within a rectangular window. Automatic optimisation of the linear function may be achieved using linear discriminative analysis and this has also been shown to yield increased recognition performance [3].

However, long term contextual information such as the speaker dependence of the acoustic realisation of phonemes will not be adequately modelled by a linear transformation to a small subspace. Methods are needed that can capture high order correlations over long time periods.

1.3 Hybrid connectionist / Markov model systems

Use of Multi-Layer Perceptrons (MLPs) allows a large window of parameterised speech to be used directly for the estimation of phone class probabilities [5]. Indeed, it can be seen that any linear transformation may be built into the first layer of a MLP by modifying the weights before the non-linearity. The use of multiple layers allows the independence restrictions to be relaxed so enabling high order correlations to be exploited. Experiments with phone recognition [8], and word recognition using context independent phone models [6] report that connectionist probability estimators yield better results than the equivalent HMM based on mixtures of Gaussian likelihoods.

There are two extremes in approaches to building hybrid connectionist/HMM systems. At one end, a standard HMM can be considered as a connectionist model with as many layers as there are frames of speech allocated to the model. Performing gradient ascent in the log likelihood of the model gives standard Maximum Likelihood trained models (e.g. [1]). At the other extreme the phone class probability estimators are trained independently of the HMM transition probabilities. This is similar to Viterbi training of HMMs in that only the most probable state sequence is used to train the emission probabilities from a state and has the advantage that discriminative training can be used (e.g. [5]). There are several intermediate positions in which gradient descent techniques can be used for discriminative training of HMMs and posterior state occupancy probabilities can be used as targets for connectionist training.

There are also a variety of architectures worth considering for the form of connectionist probability estimators. The simplest employs a standard three layer MLP structure. Whilst this has been shown to give good results [5], at best the number of parameters to estimate varies linearly with the temporal extent of acoustic information considered. Weight sharing gives better scaling properties at the expense of imposing restrictions on the diversity of the computations performed [10]. Along with the non-connectionist probability estimation methods, these techniques are restricted to a finite length window on the acoustic data.

1.4 Recurrent nets for phone probability estimation

The incorporation of feedback within a MLP gives a method of efficiently incorporating long term context in much the same way as an IIR filter can be more efficient than a FIR filter in terms of storage and computational requirements. Duplication of resources is avoided by processing one frame of speech at a time in the context of an internal state as opposed to applying nearly the same operation to each frame in a larger window. Feedback also gives a longer context window, so it is possible that uncertain evidence can be accumulated over many time frames in order to build up an accurate representation of the long term contextual variables.

The rest of this paper will give describe such a recurrent net used to estimate phone class probabilities for incorporation with a Markov model word recognition system. Results are presented at both the phone and word levels, along with a discussion of the work that still needs to be done.

2 System overview

2.1 Preprocessor

The preprocessor used in this system is fairly conventional. Both databases discussed in this paper are sampled at 16kHz. A Hamming window of width 256 samples is applied to the speech waveform every 16ms. From this window the following features are extracted: The log power; an estimate of the fundamental frequency and degree of voicing from the position and amplitude of the highest peak in the autocorrelation function; and a normalised power spectrum from an FFT grouped into 20 mel scale bins.

After the preprocessor, all channels are normalised and scaled to fit into a byte using a monotonically increasing function such that every value is eqiprobable. On presentation to the network, these values are expanded into a Gaussian distribution with zero mean and unit variance. This normalisaton was done to reduce the storage requirements of the preprocessed database.

2.2 Recurrent net

The recurrent net is a MLP where part of the output is fed back to the input after a time delay of one frame. This feedback forms an internal state in which context information may be stored. This is illustrated in figure 1.

There are 23 inputs from the preprocessor, and about 200 inputs from the state vector. There is one output for every phone in the lexicon, this translates to 61 or 49 outputs for the TIMIT and Resource Management (RM) tasks, respectively. The exact number of state units and hence the size of the network is limited by the storage and computational power of the computer used to train the network.

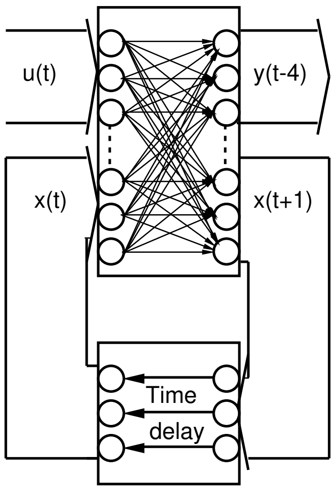

Figure 1: The recurrent network

2.2.1 Recognition

Recognition is performed by presenting the network with an external input vector from the preprocessor and an internal input vector from the state vector computed at the last time frame. These two vectors are concatenated and multiplied by the weight matrix. The resultant vector is split; the standard sigmoid squashing function is applied to every state unit, and the one-from-many softmax activation function is applied to the output units. This makes the state units independent of each other, and restricts the output units to sum to one. There is a four frame delay between presenting the acoustic evidence at the input to the net, and obtaining the phone probabilities. This is to allow some forward context and to allow non-linear processing by the state units. Increasing the delay allows more acoustic information to be used, but requires the information from the central frame to stored in the state vector for longer, resulting in competition for resources that could otherwise be used to compute more accurate non-linear transforms.

At the very start of the recognition each element of the state unit is initialised to 0.5. After several frames of silence the network settles to a steady state. During the transition period the network typically estimates closures, which is reasonable. Stability with respect to the initial state has never proved to be a problem, although the time taken for transients to decay does limit the system to training and recognition in acoustic context.

2.2.2 Training

The network is trained by "Back propagation through time" (e.g. [7, 11]) which considers the state units as hidden units of a future output. In this way the network weights can be optimised without specifying target values for the state, or even a particular form of information storage within the state.

A forced alignment with the correct word string gives a target phone class for every frame. Training is performed by performing a forward pass over a batch of 32 frames, the state unit being available from the previous time frame as in the recognition phase. The contribution to the gradient of the cost function from the frames in the window can be calculated directly by differentiation. The effect of future frames is disregarded, allowing the gradient at the state units at the end of the buffer to be set to zero. The derivatives can then be "back-propagated" through the network, starting at the last frame in the buffer which the gives derivatives for the state units used for the penultimate frame and so on back to the first frame. The assumption of disregarding future context is reasonable provided that there is sufficient evidence within the buffer for the context dependent features to be modelled. The start point of the buffer is varied on every iteration to average the effect of the boundary conditions. A local gradient is generated from the sum of the gradients over 64 buffers, or about 0.2% of the complete training set.

Parameter adaptation is done by considering a step size for every weight, and adjusting each weight by this amount in the direction of the local gradient. The step size is geometrically increased if the sign of the local gradient is in agreement with an averaged gradient, otherwise it is geometrically decreased. This training method was found to be considerably faster for this task than the others that can be found in the literature. About 64 passes through the entire training set is required for a given set of phone boundaries, and about four forced alignments and retrainings are required to move from a system trained on one task to a new task.

2.3 Markov model

The Markov model used was a simple context independent single pronunciation system with no provision for function word or cross word modelling.

Scaled emission probabilities for each state are derived from the posterior probabilities computed by the network by dividing by the prior probability of the phone occurrence [5].

2.3.1 Time domain pruning

Previously, a variable rate approach to Viterbi recognition has been proposed for this system [9]. A slight modification of this is to consider the probability that no phone transition occurs within a group of frames, or equivalently that the model remains in any one phone state for the whole segment. The product of the scaled emission probabilities, $y_i(t)$, and transition probabilities, a_i, for a specific phone, i, gives the probability of staying in that phone state. Summing the result over all phones in a segment from t to $t+T$ gives the probability of staying in any phone state for the duration, P_t^{t+T}.

$$P_t^{t+T} = \sum_i a_i^{T-1} \prod_{n=t}^{t+T} y_i(n) \qquad (1)$$

This equation can be used to recursively define a series of boundaries, $B(n)$, such that within any one segment, no value of P_t^{t+T} is less than some threshold value, $P_{\min}$. Thresholding is used as for a real-time system this segmentation should be computed with minimum delay.

$$\begin{aligned} B(0) &= 0 && (2) \\ B(n+1) &= B(n) + T \quad \text{such that} && (3) \\ & P_{B(n)}^{B(n)+T} \geq P_{\min} > P_{B(n)}^{B(n)+T+1} && \end{aligned}$$

Provided none of the phone boundaries obtained by the unconstrained Viterbi decoding are deleted, then replacing the sequence of $y_i(n)$ between two boundaries by P_t^{t+T} leaves the system unchanged. The degree of pruning can be varied by varying the threshold, $P_{\min}$.

Figure 2 presents the percentage error for the no grammar and word pair grammar cases versus the degree of pruning. The top solid curve represents previously presented results [9], and the bottom, dashed curve those according to equations 1-3. As can be seen from the plots, this technique allows for a reduction in the frame rate by a factor of about three with no extra errors in the word-pair grammar case and a slight decrease in the number of errors in the case of no grammar.

2.3.2 Minimum state durations

Along with other researchers, it has been found that imposing state duration constraints yields a significant reduction in the number of errors [9]. The minimum duration constraints are easily incorporated by rewriting every state as a sequence of states with tied emission probabilities, as in figure 3. Only one of the states has a non-zero self-loop probability, so the computational overhead is minimal. However, when a system using both time-domain pruning and minimum state durations was implemented, it was found that

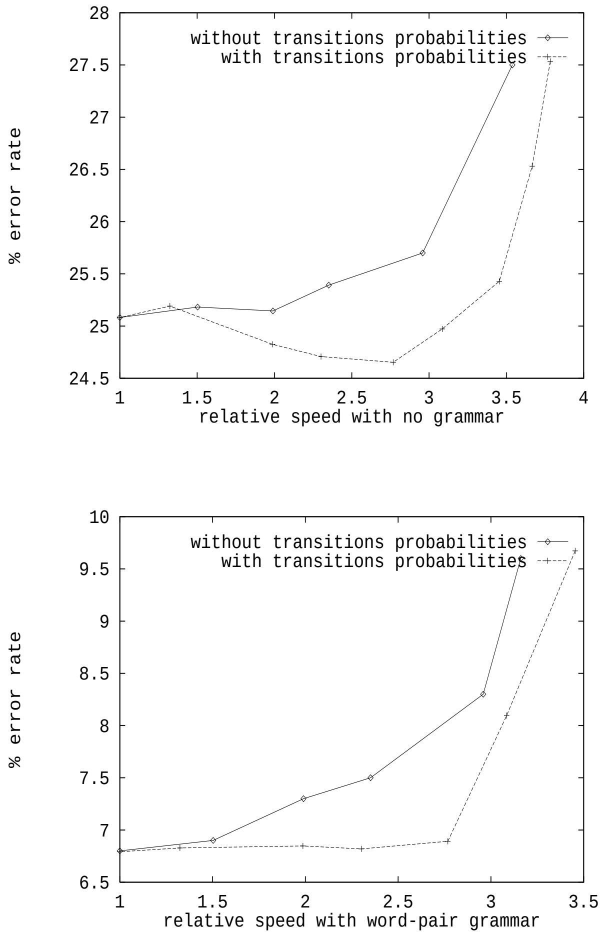

Figure 2: Time domain pruning with no grammar and with the word-pair grammar

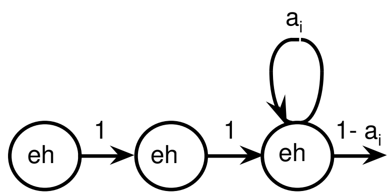

Figure 3: Lower bounds on state durations

any use of time-domain pruning increased the error rate. Presumably both schemes are crude methods for durational modelling, with minimum state durations being the more effective of the two. Results are reported without time-domain pruning, although it remains a convenient way to decrease the time taken to do Viterbi recognition (e.g for real time recognition or as a component to a fast match procedure).

3 Results

3.1 Phone recognition from the TIMIT database

The earliest work with the recurrent net architecture was performed for phone recognition from the TIMIT acoustic-phonetic continuous speech corpus. Here the Markov model is ergodic with 61 states, one per phone label. No minimum state durations have been investigated as the Markov model imposes only weak syntactic constraints in the form of a bigram grammar, and so it is likely that the acoustic evidence is dominant. The recurrent net used 192 state units and 54648 had adjustable weights.

Minor improvements have been made in the last year over [8] giving the phone recognition accuracies of table 1.

task	correct	sub^n	del^n	ins^n	errors
TIMIT	73.3%	20.8%	5.9%	3.5%	30.2%

Table 1: TIMIT phone recognition results

3.2 Multiple speaker RM task

The first word recognition results with this architecture were performed on a multiple speaker task created by training on the first five hundred utterances from each of the twelve speakers in the speaker dependent RM corpus. The remaining one hundred utterances from each of the twelve speaker in the speaker dependent training set were used for testing. Results with 160 state units were reported in [9], and improved to 5.4% errors with the word-pair grammar before presentation.

Increasing the number of state units from 160 (38456 parameters) to 192 (52056 parameters) gave the results in table 2. The number of errors is reduced by 13% for a 35% increase in the number of weights.

grammar	correct	sub^n	del^n	ins^n	errors
none	82.3%	13.4%	4.3%	2.5%	20.2%
word pair	95.9%	2.6%	1.5%	0.6%	4.7%

Table 2: RM multiple speaker results

3.3 Speaker independent RM task

The multiple-speaker system was used to bootstrap the standard 109 speaker training set for the speaker independent RM task. An increase in the physical memory on the computer used for training allowed an increase in the number of state units to 256 which corresponds to 85400 adjustable weights. Results are presented in tables 3 and 4.

task	correct	sub^n	del^n	ins^n	errors
Feb89	81.3%	15.6%	3.2%	3.2%	22.0%
Oct89	79.1%	17.4%	3.5%	3.3%	24.2%
Feb91	82.6%	14.5%	2.9%	3.7%	21.1%
Sep92	75.3%	20.8%	3.9%	4.0%	28.6%

Table 3: Speaker independent results with no grammar

task	correct	sub^n	del^n	ins^n	errors
Feb89	94.8%	4.2%	1.0%	0.9%	6.1%
Oct89	94.4%	4.5%	1.1%	0.9%	6.4%
Feb91	95.5%	3.5%	1.1%	0.7%	5.3%
Sep92	90.0%	7.7%	2.3%	1.6%	11.7%

Table 4: Speaker independent results with the word-pair grammar

4 Discussion

At the phone level, this system performs well in comparison to other systems that have been applied to this task (e.g. [4]).

However, at the word level there is a significant difference in performance between the results reported here and the best systems for this task. There are probably two main factors: the number of parameters used; and the complexity of the word modelling.

At somewhat under a million parameters, this system is considerably smaller than most other systems that are applied to this task. The number of parameters is limited due to the computational complexity of the training algorithm. Training time for the speaker independent task was one week on a 60Mflop machine, although it is hoped that this will decrease with new hardware. A better understanding of the functions computed by the state units is likely to be beneficial to training time and overall performance. At the same time, the network is trained to discriminate between classes, so it is reasonable to expect that this modelling of the class boundaries would take fewer parameters than modelling the full class distributions.

The poor word modelling is conceptually a simpler problem to address as many techniques are known which would improve the system (e.g. the use of multiple pronunciations per word and cross word modelling). Without the use of triphones or function word dependent phones, the current system is very sensitive to the accuracy of the pronunciation dictionary, and there is no scope for allowing increased acoustic variation in certain contexts, such as in function words.

The division of the system into a connectionist probability estimator and a Markov model recogniser gives additional freedom of design not found in a single HMM system. For example, it would be possible to decompose the phone set into a number of independent features which would be recombined at recognition time. Such a decomposition may allow adequate training of infrequent phones through training of individual features in other examples. This and other issues will be addressed in future work.

5 Conclusion

This paper has presented a relatively simple speech recognition system with a powerful mechanism for incorporating acoustic context. Whilst much work remains to be done at the word level, phone level results are good, and it is hoped that though increasing the size of the recurrent network and incorporating established advances in word recognition, that the phone level advantages will also been seen at the word level.

Acknowledgements

The author would like to acknowledge the UK Science and Engineering Research Council for personal support; NIST for the provision of the TIMIT and Resource Management databases and the ParSiFal project (IKBS/146) which developed the transputer array.

References

[1] J. S. Bridle and L. Dodd. An Alphanet approach to optimising input transformations for continuous speech recognition. In *Proc. ICASSP*, pages 277–280, 1991.

[2] S. Furui. Speaker independent isolated word recognition using dynamic features of speech spectrum. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, pages 52–59, 1986.

[3] R. Haeb-Umbach and H. Ney. Linear discriminant analysis for improved large vocabulary speech recognition. In *Proc. ICASSP*, volume I, pages 13–16, 1992.

[4] A. Ljolje. New developments in phone recognition using an ergodic hidden markov model. Technical memorandum TM-11222-910829-12, A T & T Bell Laboratories, 1991. Submitted to IEEE transactions on Signal Processing as: High Accuracy Phone Recognition Using Context Clustering and Quasi-triphonic Models.

[5] N. Morgan and H. Bourlard. Continuous speech recognition using multilayer perceptrons with hidden Markov models. In *Proc. ICASSP*, pages 413–416, 1990.

[6] S. Renals, N. Morgan, M. Cohen, and H. Franco. Connectionist probability estimation in the Decipher speech recognition system. In *Proc. ICASSP*, volume I, pages 601–604, 1992.

[7] A. J. Robinson. *Dynamic Error Propagation Networks*. PhD thesis, Cambridge University Engineering Department, Feb. 1989.

[8] T. Robinson. Several improvements to a recurrent error propagation network phone recognition system. Technical Report CUED/F-INFENG/TR.82, Cambridge University Engineering Department, Sept. 1991.

[9] T. Robinson. A real-time recurrent error propagation network word recognition system. In *Proc. ICASSP*, volume I, pages 617–620, 1992.

[10] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang. Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(3):328–339, Mar. 1989.

[11] P. J. Werbos. Backpropagation through time: What it does and how to do it. *Proc. IEEE*, 78(10):1150–1560, Oct. 1990.