A REAL–TIME RECURRENT ERROR PROPAGATION NETWORK WORD RECOGNITION SYSTEM

Tony Robinson

Cambridge University Engineering Department,
Trumpington Street, Cambridge, CB2 1PZ, England.

ABSTRACT

This paper presents a hybrid system using a connectionist model and a Markov model for the DARPA Resource Management task of large-vocabulary multiple-speaker continuous speech recognition. The connectionist model employs internal feedback for context modelling and provides phone state occupancy probabilities for a simple context independent Markov model. The system has been implemented in real-time on a workstation supported by a DSP board. The use of context independent phone models leads to the possibility of time-domain pruning and computationally efficient durational modelling, both of which are reported in the paper.

1. INTRODUCTION

In the field of very large vocabulary speech recognition it is acknowledged that sub-word units must be used as an intermediate state between the acoustic and lexical levels. A common choice for a suitable sub-word unit is the phone and the use of pronunciation dictionaries justifies this level of description. However, there is great variation in the acoustic realisation of a phone, both between speakers and for a single speaker depending on context. Some method of modelling this variation is required in order to produce high accuracy speech recognition systems.

This paper employs a recurrent connectionist model for modelling the low level mapping of the context dependent acoustic information to a context independent phonetic form. This is followed by a simple Markov model to perform the decoding to a sequence of words. This system places the computational burden at a lower level than a standard triphone based Hidden Markov Model (HMM) system (e.g. [5]).

The paper starts with a brief description of the basic hybrid system [11], and a real-time implementation of it. The use of context independent phone models leads to two unusual aspects that are explored in the next sections. Firstly, the outputs of the connectionist model are highly correlated which leads to the possibility of performing pruning in the time domain. Secondly, an analysis of the errors reveals that some phones are being recognised at their shortest possible duration, i.e. a single frame. This is quite improbable for most phones and suggests the need for a better durational model. Finally the performance of the system is evaluated and compared with other similar systems and suggestion are made for further improvements.

2. THE BASIC HYBRID SYSTEM

To date there have been two main approaches to building hybrid connectionist/HHM systems, both of which employ a connectionist model for the estimation of HMM state probabilities. The first method is similar to the forward-backward reestimation algorithm in that both the connectionist and HMM structures are simultaneously optimised to maximise the log likelihood of the observed sequence being generated by the model (e.g. [1]). The second method is that of Viterbi training where the HMM is used to make a forced alignment and so generate the target probability distributions for the connectionist model (e.g. [7]). Researchers using standard HMM techniques report that there is little difference in the performance of the two methods on large vocabulary tasks (e.g. [3]), and this work employs the second technique for simplicity.

The form of connectionist model used in this system employs internal feedback to model the context dependency in speech. This is in contrast to the more common connectionist solution of using a fixed length input window which slides over the speech, or the standard HMM approach which uses multiple states per phone to model a succession of steady state regions. Comparisons of this system with HMMs using triphones for modelling context dependency are favourable for the task of phone recognition from the TIMIT database [10]. By modelling context effects internally in the recurrent network, the output can be context independent phones, which greatly simplifies the subsequent Markov model.

The recurrent network is shown in figure 1. The input, $u(t)$, comes from the preprocessor and together with the state vector, $x(t)$, is multiplied by the weight matrix. The output is passed though a non-linear transform to yield the output vector, $y(t+1)$, and the next state vector, $x(t+1)$. The weight matrix is trained by unfolding in time using a modified gradient descent procedure.

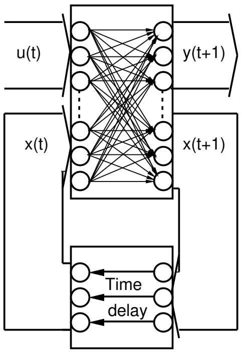


Figure 1. The Recurrent Error Propagation Network

The current preprocessor employs a power channel, 20 mel scaled power spectra channels, a channel for fundamental frequency and one for the degree of voicing. There are 160 state units, and 49 output units, one for each phone in the lexicon. Training the network as a phone classifier on the Resource Management task naturally conditions the phone probability outputs to match the distribution of phones in the database. However, a grammar that well models this database will also generate this distribution of phones, and so, following [7], the phone probabilities output by the network are divided by their prior probabilities of occurrence when doing word recognition.

The speaker dependent part of the Resource Management database [9] was used in a multiple-speaker mode. Of the 12 speakers in the database, the first 500 sentences from each speaker were used for training and the last 100 for testing. There are 894 unique words in the training data, and 392 in the test data. In the test data, 74 of the words do not occur in the training data.

An initial segmentation was achieved using a recogniser trained on the TIMIT database. This was then refined by retraining and using forced alignment to get a better set of phone boundaries. Each training cycle took 32 passes of the training data to converge. After three cycles the word error rate stabilised to the values given in table 1.

perp.	correct	ins^n	sub^n	del^n	errors
975	78.1%	3.2%	16.0%	5.9%	25.1%
60	94.6%	1.4%	3.5%	1.9%	6.8%

Table 1. Baseline results

3. A REAL-TIME IMPLEMENTATION

The system described above has been implemented on a SparcStation II with a DSP32C SBus board. The DSP performs all the processing up to the point of the estimation of the phone state occupancy probabilities, which are then parsed to the word level by the UNIX host.

Compared to triphone based large vocabulary HMM architectures, the use of context independent network outputs requires relatively little computation to calculate the phone state occupancy probabilities. Moreover, the largest task is that of the forward pass of the recurrent network, which is mainly repeated multiply and accumulate operations. As such, the bottom end is well suited to implementation on a single DSP. The network has 38,456 weights, each of which occupy 32 bits as floating point values, so the storage space required is not excessive. Use of the DSP32C digital signal processor has allowed a complete implementation of the preprocessor and network in real time with the standard 16ms frame rate.

In this paper, the real-time recogniser uses a standard Viterbi decoder without employing any pruning. A tree structured dictionary with one entry for each of the 975 unique pronunciations has 3043 unique states with no grammar. The log probabilities for each state can be accumulated in real time on an unloaded SparcStation II. Currently, backtracking is performed when more than 128ms of silence have been detected, which in practice on the Resource Management database is always at the end of a word and often at the end of a sentence.

It is expected that the speed improvements gained by pruning the search space will be more than enough to allow an optional silence at the end of each word and the use of the standard word-pair grammar. The remainder of the paper presents results for these recogniser configurations by storing the output probabilities for later parsing.

4. TIME DOMAIN PRUNING

Several authors have proposed a variable frame rate acoustic analysis for speech recognition. Often the motivation has been a reduction in the overall computation requirements, although in some cases it can also lead to improved recognition accuracy [8]. This section proposes a variable frame rate approach at the next level up from the acoustic level, that of probabilistic state segmentation.

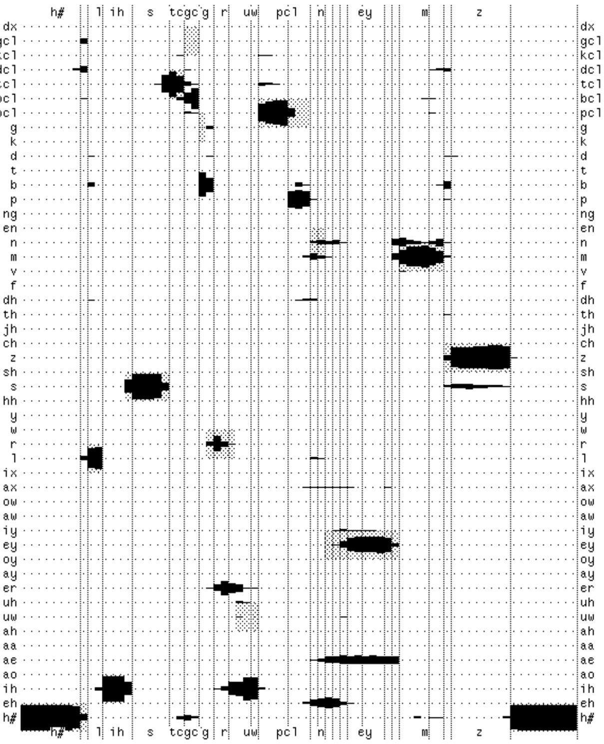


Figure 2. Example output with boundary pruning

Figure 2 shows the phone state occupancy probabilities as the output from the recurrent network for the short test sentence "list group names" (speaker `bef`, sentence `sr507`). Time is displayed along the horizontal axis and each of the 49 phones on the vertical axis. Several steady state regions can be identified where several consecutive frames have a near unity estimated state occupancy probability for one phone, and near zero probabilities for the remainder.

In a standard Viterbi search, phone boundaries are proposed at every frame. However, boundaries in the middle of the steady state periods will only exist with very low probability, and this can be used as the basis for a preliminary segmentation. To be useful to a real-time system, this segmentation should be computed with minimum delay.

If the assumption is made that observations between two proposed boundaries are independent then equation 1 can be used to calculate the probability that the model remains in any one phone state for the whole segment. The product of the state occupancy probabilities for a specific phone gives the probability of being in that phone state for the whole segment, and by summing the result over all phones gives the probability of being in any phone state for the whole segment. In practice, the original independence assumption is false, so leading to an under-estimate. Use of the self-loop probabilities may give better segmententation.

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

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, $P_{\min}$.

$$B(0) = 0 \qquad (2)$$
$$B(n+1) = B(n) + T \qquad (3)$$

$$\text{such that} \quad P_{B(n)}^{B(n)+T} \geq P_{\min} > P_{B(n)}^{B(n)+T+1}$$

Provided none of the phone boundaries obtained by the unconstrained Viterbi decoding are deleted, it is possible to combine all the frames within a segment such that the likelihood of the most probable segmentation, and hence the recognition results, are unchanged. This is accomplished by taking the product of the phone state occupancy probabilities over the segment and including the self loop probabilities for all internal transitions within the segment.

$$y_i'(n) \quad = \quad a_i^{B(n+1)-B(n)-1} \prod_{t=B(n)}^{B(n+1)-1} y_i(t) \qquad (4)$$

The degree of pruning can be varied by varying the threshold, $P_{\min}$. Table 2 and figure 3 present the percentage error for the no grammar and word pair grammar cases with the degree of pruning. For an increase in the number of errors by 5%, this technique allows for a reduction in the frame rate by a factor of 2.8 for no grammar, and a factor of 1.8 with the word-pair grammar. By counting the number of phone boundaries and the number of frames, the maximum degree of pruning without missing a phone boundary is a factor of 5.0. Whilst this is not a great advantage, it is largely independent of the speed advantages gained by the use of a standard beam search pruning technique.

$P_{\min}$	speed-up	no grammar	WP grammar
1.00	1.00	25.1%	6.8%
0.81	1.50	25.2%	6.9%
0.53	1.99	25.1%	7.3%
0.40	2.35	25.4%	7.5%
0.25	2.96	25.7%	8.3%
0.10	3.86	28.5%	14.1%

Table 2. Effect of pruning on the error rates

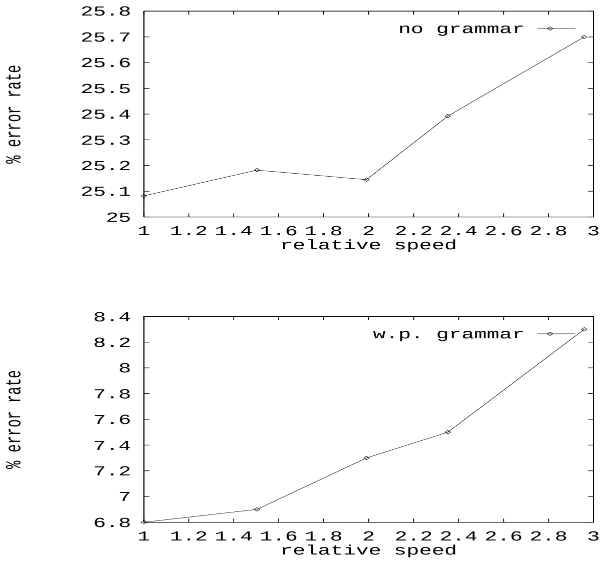


Figure 3. Time domain pruning with no grammar and with the word-pair grammar

5. MINIMUM STATE DURATIONS

It is widely acknowledged that a standard HMM provides poor durational modelling. This is especially important if a single state is to model a relatively large unit such as a phone as the exponential decay of probability of state occupancy with time is a poor match to the observed duration distribution. Proposed solutions to this problem include explicitly computing the distribution from the training set, using parameterised versions with the Poisson or Gamma distributions [6], and using minimum and maximum durations[2, 3].

Minimum duration constraints can be easily incorporated by rewriting every state as a sequence of states with tied emission probabilities. If only one of the states has a nonzero self-loop probability, as in figure 4, then the computational overhead is very little, only two extra lookup operations for every phone model. This is achieved by keeping a buffer of the accumulated log probabilities for each state in the grammar with the length of the minimum state duration. A similar buffer is kept for the state occupancy probabilities for each phone (in the same way as equation 4), and on entering a state the probabilities for a delay of the minimum phone duration are used.

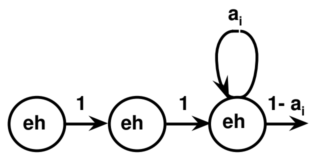


Figure 4. Lower bounds on state durations

Minimum state durations were calculated from the TIMIT database so as as to exclude no more than a given fraction of occurrences. Self-loop probabilities, a_i, are then calculated by assuming that all occurrences that fall below the minimum duration take the minimum value. Figure 5 shows the true durational distribution for an example phone [eh], the distribution of the standard system (minimum duration of 1), and the distribution when bounded by 1/32 and 1/16 of the total number of occurrences (minimum durations of 3 and 4 respectively).

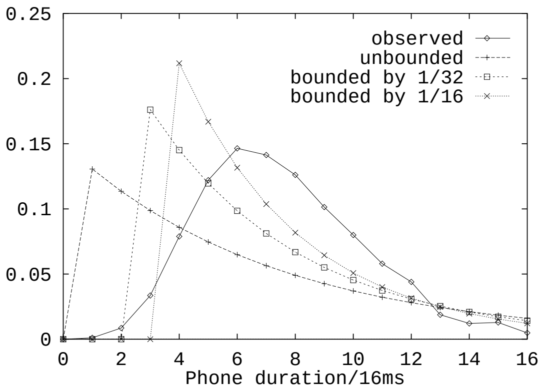
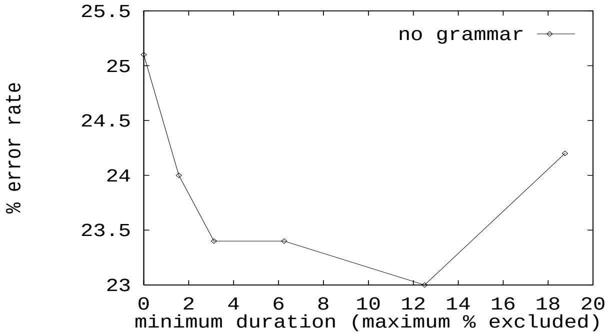


Figure 5. Bounded state durations for [eh]

Figure 6 plots the total number of errors against the threshold fraction of occurrences which controls the minimum duration. It can be seen that there is a considerable improvement in performance when minimum duration constraints are applied, from an error rate of 25.1% to 23.0% in the case of no grammar and from 6.8% to 5.7% when using a word pair grammar. The width of the minimum is relatively wide so the performance does not depend critically on the threshold chosen.



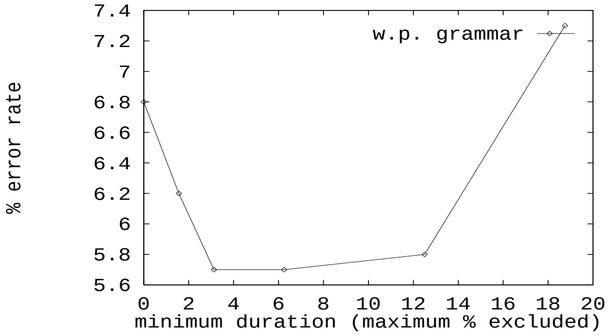


Figure 6. Time domain pruning with the word-pair grammar

6. CONCLUSION

This paper has presented a hybrid connectionist/Markov model recogniser capable of real-time recognition. Because the system uses context independent phone models, it has been shown that a limited amount of pruning is possible in the time domain. This pruning scheme always increases the error rate, while other schemes have reported a decrease in error rate [8]. The use of minimum durations for the phone models has been used to give increased performance at little computational cost. However, the best reported results in this paper still have over twice the error rate of the best systems (for example BBN report 2.6% word error rate on this task [4]).

An analysis of the remaining errors shows that the majority are caused by short words. "the" and "a" each account for 10% of all the errors made with the word-pair grammar, while "what", "of", "to", "on", "in" account for the next 10%. Unlike similar HMM based systems for this task, the current system makes no attempt to explicitly model these short words.

The use of a large number of context dependent phone models gives a certain degree of robustness to the pronunciation dictionary. This robustness is achieved by using infrequently used triphones to model minor errors in transcription, or pronunciation variation. In contrast, the use of context independent phone models with a single transcription per word is relatively brittle. However, with increasing vocabulary size, triphone contexts will necessarily become less specific to individual words, so resulting in a decrease in the transcription robustness. It is expected that the use of a multiple-pronunciation dictionary will increase the robustness to pronunciation variation.

Currently the word boundary modelling is poor, as can be seen in the example sentence "list group names" of figure 2. The generated transcription is [l ih s tcl gcl g r uw pcl n ey m z], even though the sequence [tcl gcl] is illegal and there is a clear release of the /p/. Again, this is an area for further work.

In conclusion, a reasonably simple system has been presented with an unusual method for modelling context dependency. Further work is necessary to bring this system to state-of-the-art performance.

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; the ParSiFal project (IKBS/146) which developed the transputer array; and Ariel Corporation for the SBus DSP32C board.

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] H. Gu, C. Tseng, and L. Lee. Isolated-utterance speech recognition using hidden Markov models with bounded state durations. *IEEE Transactions on Signal Processing*, 39(8):1743–1752, Aug. 1991.

[3] V. N. Gupta, M. Lennig, P. Mermelstein, P. Kenny, F. Seitz, and D. O'Shaughnessy. Using phoneme duration and energy countour information to improve large vocabulary isolated-word recognition. In *Proc. ICASSP*, pages 341–344, 1991.

[4] F. Kubala and R. Schwartz. A new paradigm for speaker-independent training. In *Proc. ICASSP*, pages 833–836, 1991.

[5] K.-F. Lee. *Automatic Speech Recognition: The Development of the SPHINX System.* Kluwer Academic Publishers, Boston, 1989.

[6] S. E. Levinson. Continuously variable duration hidden Markov models for automatic speech recognition. *Computer Speech and Language*, 1(1):29–45, Mar. 1986.

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

[8] K. M. Ponting and S. M. Peeling. The use of variable frame rate analysis in speech recognition. *Computer Speech and Language*, 5:169–179, 1991.

[9] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett. The DARPA 1000-word Resource Management database for continuous speech recognition. In *Proc. ICASSP*, pages 651–654, 1988.

[10] 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.

[11] T. Robinson and F. Fallside. A recurrent error propagation network speech recognition system. *Computer Speech and Language*, 5(3):259–274, July 1991.