SMOOTHED LOCAL ADAPTATION OF CONNECTIONIST SYSTEMS

Steve Waterhouse *Dan Kershaw* *Tony Robinson*

Cambridge University Engineering Department, Cambridge CB2 1PZ, England.
Tel: [+44] 1223 332754 Fax: [+44] 1223 332662
email: {srw1001, djk, ajr}@eng.cam.ac.uk

ABSTRACT

ABBOT is the hybrid connectionist hidden Markov model (HMM) large vocabulary continuous speech recognition system developed at Cambridge University Engineering Department. ABBOT makes effective use of the linear input network (LIN) adaptation technique to achieve speaker and channel adaptation. Although the LIN is effective at adapting to new speakers or a new environment (e.g. a different microphone), the transform is global over the input space. In this paper we describe a technique by which the transform may be made locally linear over different regions of the input space. The local linear transforms are combined by an additional network using a non-linear transform. This scheme falls naturally into the mixtures of experts framework.

1. INTRODUCTION

The ABBOT system uses a recurrent neural network (RNN) to estimate the tied-state acoustic observation likelihoods in an HMM framework. The network's output $\mathbf{y}(t)$ is a vector whose elements represent estimates of the posterior probability of each of the phone classes given the input parameterised speech $\mathbf{u}(t)$. The posterior probabilities are mapped to scaled likelihoods for use in the HMM decoding process.

The linear input network has proved to be a successful method for the adaptation of connectionist systems to a new speaker [3]. This technique has recently been successfully extended to the adaptation of a recurrent network to an unknown microphone. Although the LIN scheme has proved successful at reducing the word error rate (WER) within both speaker and environmental adaptation, the method is suboptimal since it is a global transform of the input.

Local adaptation has proved successful in HMM adaptation [2] in which the local regions are selected according to sets of similar phone classes. The selection of local regions based on similar phone classes is not possible with the recurrent network, because the RNN is a dynamical system requiring a continuous input stream. In this paper we use separate adaptation LINs and attempt to learn the regions of data in which they should specialise. The outputs of the LIN-RNN combinations are then combined non-linearly to achieve a global transform that is locally linear.

The rest of the paper is structured as follows. The linear input network is introduced in Section 2, including the unsupervised block adaptation technique used in the rest of this paper. Section 3 presents the mixture of experts technique and how it is extended to the mixture of linear input networks (MLIN) architecture. Section 4 analyses the impact that the MLIN has on the adaptation process for connectionist systems.

2. THE LINEAR INPUT NETWORK

A linear mapping is created to transform the acoustic vector. During recognition, this transformed vector is fed as input to the speaker independent RNN. This is shown in Figure 1.

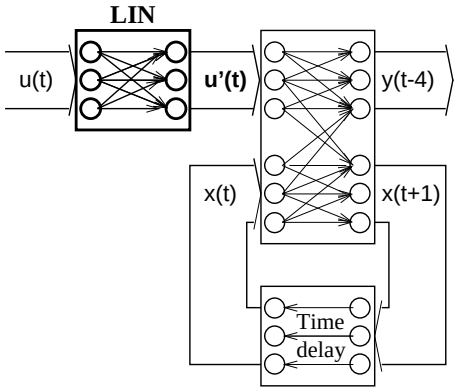

Figure 1: The linear input network "bolts on" to the recurrent network, performing a linear transformation of the acoustic feature space.

To train the LIN for a new speaker, the LIN's weights are initialised to an identity matrix; this guarantees that the initial starting point is the speaker independent model. The input is propagated forward to the output layer of the RNN. At this point the error is back-propagated through the RNN. Note that the RNN weights are kept fixed, and only the

LIN's weights are updated. This process can be likened to a global maximum likelihood linear regression (MLLR) [2], except that in MLLR the HMM parameters are adjusted. Even with few parameters (182), the LIN performs a very effective transformation of the input space. On the ARPA Resource Management speaker dependent corpus for supervised speaker adaptation, the LIN achieved a 23.8% reduction in word error rate when using 100 adaptation sentences [3].

For the 1995 ARPA Hub 3 multiple unknown microphone (MUM) evaluations [4], unsupervised block adaptation is performed over each known speaker session, to adapt to unknown speakers, noise and channel conditions over the session. A Viterbi alignment using the current model is carried out on decoded utterance hypotheses of a session, to label the acoustic frames. The LIN is now trained in a supervised fashion. Decoding is then performed with the *adapted* recurrent network. This process is iterated until there is no change in the session hypotheses from one adaptation pass to the next. The adaptation process typically converges after 2 or 3 iterations.

For the multiple unknown microphones task (H3-P0), the word error rate is reduced by 18.1%, while for the clean speech task (H3-C0) the reduction is 11.2% [1].

3. MLIN : mixtures of linear input networks for adaptation

The mixture of experts architecture consists of a set of "experts" which perform local function approximation. The expert outputs $y_i(t)$ are combined with the outputs $g_i(t)$ of a "gate" to form the overall output

$$\mathbf{y}(t) = \sum_i g_i(t)\mathbf{y}_i(t). \qquad (1)$$

In the case of classification, the experts compute vectors of class conditional probabilities [7]. In Figure 2, we show the mixture of linear input networks (MLIN) architecture. Each expert consists of a LIN and a recurrent network. The gate consists of a single layer network with softmax activation function which computes the conditional probability of selecting each expert given the current input $\mathbf{u}(t)$. During training within the EM framework, the recurrent network weights (shown shaded) are kept fixed and only the LIN weights are adapted by weighting the back-propagated error term with the posterior probability of selecting each expert given the current input and phone label. This posterior probability is given by

$$h_i(t) = g_i(t).\mathbf{y}_i^*(t) \Big/ \sum_j g_j(t).\mathbf{y}_j^*(t) \qquad (2)$$

where $\mathbf{y}_i^*(t)$ is the probability of generating the correct class from expert i. Other than this difference, training of the LIN-RNN expert combination proceeds as usual.

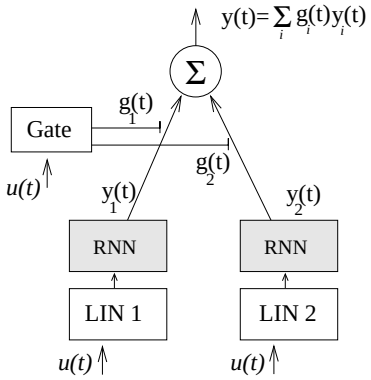

Figure 2: A mixture of 2 linear input networks (LIN) for adapting the recurrent network (RNN).

The gate is trained to predict which expert is the best one to use at each time. Its targets are thus the posterior probabilities $h_i(t)$.

Before training, the weights of the LINs in each expert are initialised to diagonal matrices with random weights between 0.95 and 1.05. The gate weight matrix is initialised using random values between -0.1 and 0.1. This randomisation ensures symmetry breaking and prevents the experts all learning the same transform.

Optimisation of the experts and gate is done via the scheme described in [6], with individual weight rate terms for each network. It is important to note that the MLIN does not increase the number of parameters significantly, since the RNN is effectively tied across the different experts. The computational cost of training is increased slightly however, due to the need to back-propagate the error terms through each copy of the RNN.

4. RESULTS

This section reports decoding results for the ARPA 1995 H3 multiple unknown microphones (MUM) Task. Decoding is performed by the ABBOT decoder, NOWAY [5], which uses a modified stack decoding algorithm, with a pruning strategy that is well matched to the hybrid connectionist-HMM approach. The subsection of the H3 task chosen for this paper was H3:C0, which is the 1995 H3 MUM unlimited vocabulary test recorded using a Sennheiser microphone. This data set contains 20 speakers with 15 sentences each.

The 1995 ABBOT system used the 60,000 word vocabulary and standard trigram generated by CMU throughout the evaluation. The pronunciation lexicon was derived primarily from a lexicon supplied by LIMSI-CNRS and expanded to cover the 60,000 word vocabulary [1].

In this section the method used for unsupervised block adaptation is the same method as described in Section 2. Results are reported for various numbers of experts and numbers of passes of adaptation at both the frame and word level.

4.1. Expert Specialisation

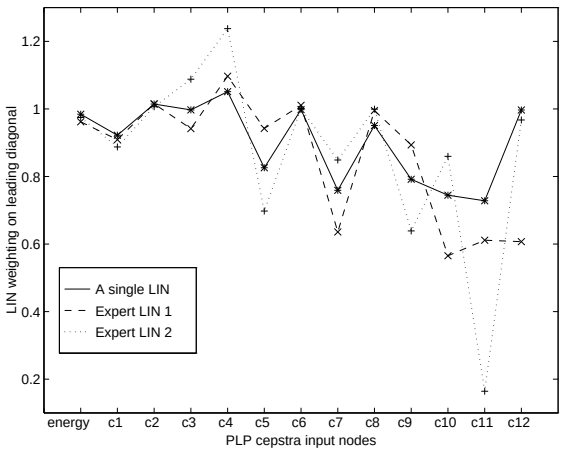

Figure 3: Comparison of the leading diagonal of the Expert LIN's weight matrices and the single LIN weight matrix.

Figure 3 shows the leading diagonals of a pair of expert LINs and the single LIN weight matrices. Although the weight matrices contain off diagonal terms also, these are not as dominant as the diagonal terms. As can be seen, the experts differ both from one another and from the single LIN, especially in the higher order cepstra.

4.2. Phoneme frame error rates

For the first pass through the adaptation process, we align the utterances using the baseline recurrent network. After training the LIN on this data we re-align the utterances using the LIN-RNN combination. Subsequent re-alignments are done using the current model after training in an iterative process. Table 1 shows the average phoneme accuracy at the frame level with respect to the current hypothesised transcription over the whole test set . All the models after 1 pass of the adaptation are trained from the data aligned using the LIN. It is clear that the MLIN performs better at this level and is learning a more accurate adaptation of the model to the hypothesised transcription.

Model	# Passes	Frame Error Rate
RNN	0	22.8 %
LIN	1	20.1 %
LIN	2	19.9 %
MLIN_2	1	19.2 %
MLIN_2	2	18.9 %
MLIN_4	2	18.2 %
MLIN_4	3	18.0 %

Table 1: Phoneme error rate at the frame level relative to the hypothesised transcriptions on the H3 task using a single LIN, MLIN with 2 and 4 experts (MLIN_2, MLIN_4).

4.3. Word error rates

We now turn to the word error rates using the LIN and MLIN adaptation techniques. Table 2 shows the error rates using different models and different numbers of adaptation passes. As can be seen, whilst the MLIN performs better at the frame level with respect to the hypothesised transcriptions, this does not translate to a significant improvement in terms of word error rates.

Model	# Passes	Sub^n.	Del^n.	Ins^n.	WER
RNN	N/A	13.1	2.9	2.1	18.1
LIN	1	11.7	2.7	2.1	16.5
LIN	2	11.2	2.6	2.1	15.9
MLIN_2	1	11.7	2.7	2.1	16.5
MLIN_2	2	11.4	2.6	2.1	16.1
MLIN_4	2	11.1	2.6	2.1	15.8
MLIN_4	3	10.9	2.6	2.1	15.7

Table 2: Decoding performance on the H3 task using a single LIN, MLIN with 2 experts (MLIN_2) and MLIN with 4 experts (MLIN_4).

This effect can be likened to a game of Chinese whispers. If the final person in the chain (analogous to the LIN) listens harder then the accuracy of their transcription may be improved, but this may still be the wrong transcription they are learning. By improving the transform used in adaptation we cause significant improvements in error rate for some speakers but also degradations in others.

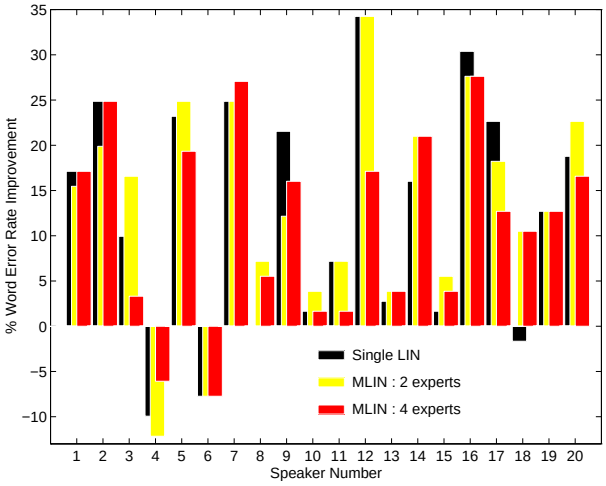

Figure 4: Improvement of word error rate per speaker using a single LIN, a mixture of 2 LINs, and a mixture of 4 LINs, after a second pass of adaptation.

This effect is demonstrated in Figure 4 which shows the improvement or degradation per speaker after different adaptation schemes and summarised in Table 3.

	Average	Average
Model	Improvement	Degradation
LIN	13.1 %	4.3 %
MLIN_2	13.5 %	6.9 %
MLIN_4	16.0 %	9.9 %

Table 3: Average improvement of word error rate over all speakers for various adaptation techniques with respect to the baseline system without adaptation.

The speakers for which adaptation degraded word error rates correspond to those for which the speaker independent system (RNN) gave very crude initial alignments.

5. CONCLUSIONS

In this paper we have demonstrated how the LIN speaker and channel adaptation approach can be extended in the mixture of experts framework. Although the word error rate improvements we have achieved are modest, we have demonstrated significant improvements at the frame level. As shown, the use of a more sophisticated transform results in an additional improvement for some speakers and an additional degradation for other speakers.

Unsupervised block adaptation suffers from the problem that even if we improve the frame rate on the hypothesised transcription, we may not be improving the word error rate with respect to the true transcription. By using a more complex transform, as we have done in this paper, one may actually degrade performance further on some speakers by adapting better to the wrong transcription. In future work we plan to test the performance of the MLIN approach on supervised adaptation. In addition, we plan to look further into the general unsupervised adaptation paradigm by using separate adaptation networks for each speaker and learning to combine them optimally for other speakers in the same framework as described here.

A number of uses for this architecture are possible. By initialising the expert LINs and gate with random weights and training, it is possible to discover structure in the input space as well as learn a nonlinear adaptation. Alternatively, prior knowledge may be used, such as initialising a set of expert LINs using different speakers or environments. Future work will look into using the MLIN approach on different channel conditions.

Acknowledgements

This work was partially funded by ESPRIT project 6487 WERNICKE. The authors would like to acknowledge MIT Lincoln Laboratory and CMU for providing language models and associated tools, LIMSI-CNRS and ICSI for providing the pronunciation lexicons. Thanks also to Gary Cook and Mike Hochberg for all their help and advice.

6. REFERENCES

1. D.J. Kershaw, A.J. Robinson, S.J.Renals, and M.M. Hochberg. The 1995 ABBOT LVCSR System. In *The ARPA Speech Recognition Workshop*, Arden House, Harriman, New York, February 1996.
2. C J Leggetter and P C Woodland. Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density HMMs. *Computer Speech and Language*, 9:171–186, 1995.
3. J. Neto, L. Almeida, M.M. Hochberg, C. Martins, L. Nunes, S.J. Renals, and A.J. Robinson. Speaker-Adaptation for Hybrid HMM-ANN Continuous Speech Recognition System. In *Eurospeech*, pages 2171–4, September 1995.
4. D.S. Pallett, J.G. Fiscus, W.M. Fisher, J.S. Garofolo, A.F. Martin, and M.A. Przybocki. 1995 HUB-3 NIST Multiple Microphone Corpus Benchmark Tests. In ARPA, editor, *The ARPA Speech Recognition Workshop*, Arden House, Harriman, New York, February 1996.
5. S. Renals and M. Hochberg. Efficient search using posterior phone probability estimates. In *Proc. ICASSP*, volume 1, pages 596–599, Detroit, 1995.
6. Tony Robinson, Mike Hochberg, and Steve Renals. The use of recurrent networks in continuous speech recognition. In C. H. Lee, K. K. Paliwal, and F. K. Soong, editors, *Automatic Speech and Speaker Recognition – Advanced Topics*, chapter 19. Kluwer Academic Publishers, 1995.
7. S. R. Waterhouse and A. J. Robinson. Classification using hierarchical mixtures of experts. In *IEEE Workshop on Neural Networks for Signal Processing*, pages 177–186, 1994.