DISCRIMINATIVE OPTIMISATION OF LARGE VOCABULARY RECOGNITION SYSTEMS

V. Valtchev *P.C. Woodland* *S.J. Young*

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

ABSTRACT

This paper describes a framework for optimising the structure and parameters of a continuous density HMM-based large vocabulary recognition system using the Maximum Mutual Information Estimation (MMIE) criterion. To reduce the computational complexity of the MMIE training algorithm, confusable segments of speech are identified and stored as word lattices of alternative utterance hypotheses. An iterative mixture splitting procedure is also employed to adjust the number of mixture components in each state during training such that the optimal balance between number of parameters and available training data is achieved. Experiments are presented on various test sets from the Wall Street Journal database using the full SI-284 training set. These show that the use of lattices makes MMIE training practicable for very complex recognition systems and large training sets. Furthermore, experimental results demonstrate that MMIE optimisation of system structure and parameters can yield useful increases in recognition accuracy.

1. INTRODUCTION

Previous research has shown that the accuracy of a speech recognition system trained using the Maximum Likelihood Estimation (MLE) criterion can often be improved further using discriminative training. In particular, Maximum Mutual Information Estimation (MMIE) [1] has been applied to small vocabulary tasks and substantial gains in performance have been reported [2, 3].

For several reasons, discriminative optimisation of HMM parameters is much more complex than the conventional MLE framework. First, the discriminative nature of the objective function inherently requires the availability of acoustically confusable segments of speech that normally constitute the errors made during recognition. Even in a small vocabulary task, the gathering of statistics about mismatched segments of speech results in a dramatic increase in computational requirements compared to the corresponding MLE case. Second, given the current state/frame alignment of the training data there are no closed form solutions for parameter estimates that maximise the objective function. Instead, some form of gradient-based optimisation must be used. Thus, whilst an MLE system can typically be trained in a few iterations, MMIE training may require considerably more.

More recently the MMIE training algorithm was applied to the HTK large vocabulary continuous recognition system [7] using the Wall Street Journal database. Using lattices to represent alternative sentence hypotheses and a modified training algorithm with improved convergence, the work in [6] demonstrated the viability of the technique in the large vocabulary task domain. Improvements in recognition performance of 5%-10% were observed. At the same time the discriminative optimisation of HMM parameters allowed the overall number of free parameters in the HMMs to be reduced without any degradation in recognition performance. However, in certain cases over-training was observed with the performance of the resulting system deteriorating on an independent test set.

Current speech recognition systems rely on mixtures of Gaussian densities to model the acoustic data. The use of such densities has several advantages, most important of which is the ability to derive an arbitrarily close approximation to the "true" distributions of the source. Thus, the art of building a good acoustic model often translates into finding the right balance between the number of free parameters in the system and the amount of training data available. To achieve good recognition performance, it is necessary that the models utilise a large number of mixture components to adequately model the acoustic variability across different samples of the same speech sound (*resolution*). However, the number of mixture components should be small enough to allow for reliable estimates of the Gaussian parameters and mixture component weights (*trainability*). In most recognition systems the optimal balance between resolution and trainability is achieved through parameter sharing. Examples include tied mixture systems where all output distributions share the same set of Gaussian components and decision tree state-clustered systems where many HMM states in the system use the same mixture distribution.

This paper extends the discriminative training framework presented in [6] to incorporate a mixture component splitting mechanism based on the MMIE criterion. The algorithm is aimed to provide improved resolution in output distribution where confusions occur and where the amount of training data available allows for reliable estimation of model parameters. In the remaining sections of the paper, the basic framework is first described and then a number of experiments using the HTK tied-state LVCSR system [7] and the Wall Street Journal database are presented.

2. MMI ESTIMATION OF HMM PARAMETERS

For R training observations $\{\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_r, \ldots \mathcal{O}_R\}$ the MMIE objective function is given by

$$\mathcal{F}(\lambda) = \sum_{r=1}^{R} \log \frac{P_\lambda(\mathcal{O}_r|\mathcal{M}_r)P(w_r)}{\sum_{\hat{w}} P_\lambda(\mathcal{O}_r|\mathcal{M}_{\hat{w}})P(\hat{w})} \quad (1)$$

The denominator term in the above equation can be replaced by $P_\lambda(\mathcal{O}|\mathcal{M}_{gen})$ where $\mathcal{M}_{gen}$ is the model used during recognition. The calculation of statistics from $\mathcal{M}_{gen}$ is computationally involved and depends on the size of the recognition lexicon, the grammar and any contextual constraints. In many practical situations, for example where cross-word context dependent models are used in conjunction with a long span language model, the explicit construction of a complete model for $\mathcal{M}_{gen}$ is intractable.

The following formulae derived in [3] are used to re-estimate the mean ($\mu_{j,m}$) and variance ($\sigma^2_{j,m}$) parameters of Gaussian mixture component m at state j of the HMM

$$\hat{\mu}_{j,m} = \frac{\left\{\theta_{j,m}(\mathcal{O}) - \theta^{gen}_{j,m}(\mathcal{O})\right\} + D\mu_{j,m}}{\left\{\psi_{j,m} - \psi^{gen}_{j,m}\right\} + D} \quad (2)$$

$$\hat{\sigma}^2_{j,m} = \frac{\left\{\theta_{j,m}(\mathcal{O}^2) - \theta^{gen}_{j,m}(\mathcal{O}^2)\right\} + D(\sigma^2_{j,m} + \mu^2_{j,m})}{\left\{\psi_{j,m} - \psi^{gen}_{j,m}\right\} + D} - \hat{\mu}^2_{j,m} \quad (3)$$

In the above equations, $\theta_{j,m}(x)$ represents the sum of all x weighted by the probability of occupying component m of state j and $\psi_{j,m}$ represents the corresponding occupation count. Similarly, the mixture weight parameters $c_{j,m}$ can be re-estimated according to

$$\hat{c}_{j,m} = \frac{c_{j,m}\left\{\frac{\partial}{\partial c_{j,m}}\mathcal{F}(\lambda) + C\right\}}{\sum_{\hat{m}} c_{j,\hat{m}}\left\{\frac{\partial}{\partial c_{j,\hat{m}}}\mathcal{F}(\lambda) + C\right\}} \quad (4)$$

Suitable normalisation of the derivatives is also used in order to remove emphasis from small-valued parameters [3]. The constant D is set to be just large enough to ensure that all variances remain positive. The constant C is chosen such that all parameter derivatives are positive. Experimentally, these selection criteria have been shown to give relatively smooth and fast convergence [2, 3]. In addition, past experience [6] has shown that setting the value of D on per-model basis results in further speedup in convergence ($\times 2$).

3. WORD LATTICES

A word lattice forms a compact representation of many different sentence hypotheses and hence provides an efficient representation of the confusion data needed for discriminative training. In the HTK system [7], a lattice (figure 1) consists of a set of nodes that correspond to particular instants in time, and arcs connecting these nodes to represent possible word hypotheses. Associated with each arc is an acoustic score (log likelihood) and a language model score.

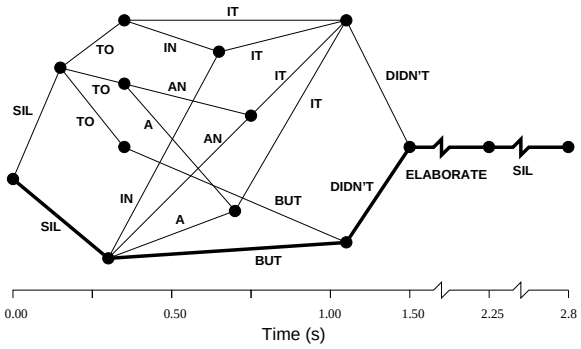

Figure 1: An example lattice. The path shown in bold corresponds to the correct transcription of the utterance.

Lattices are generated as a by-product of the recognition process. Assuming that the lattice coverage does not change during training, the use of each lattice as a constraining word graph in computing the objective function (equation 1) forms the basis of the training algorithm. For each utterance in the data set, a pair of *numerator* and *denominator* lattices is generated. The numerator lattice is produced by aligning the acoustic data against a network of HMMs, possibly including pronunciation variations, built according to the "correct" transcription. The denominator lattice corresponds to running an unconstrained recognition pass. In both cases an appropriate N-gram language model is used.

4. SPLITTING OF MIXTURE COMPONENTS

Most speech recognition systems employ a fixed number of mixture components per output distribution. The number is determined empirically, by starting off with few mixture components per state and gradually increasing this number until the performance of the system on a validation test set no longer improves. Alternatively, the number of mixture components in the output distributions can vary across different HMM states based on the amount of training data available. Even in the latter case, distributions with a large number of mixture components may learn to model intricate parts of the "interior" regions of the underlying distributions, thus not directly aiding improved discrimination.

The splitting of mixture components according to the MMIE criterion was proposed in [4]. The algorithm uses the derivatives of the objective function with respect to the mixture component weights to decide if increased resolution is needed. The parameter derivatives are given by

$$\frac{\partial}{\partial c_{j,m}}\mathcal{F}(\lambda) = \sum_{r=1}^{R}\left\{\frac{1}{P_\lambda(\mathcal{O}_r|\mathcal{M}_r)}\frac{\partial P_\lambda(\mathcal{O}_r|\mathcal{M}_r)}{\partial c_{j,m}} - \frac{1}{P_\lambda(\mathcal{O}_r|\mathcal{M}_{gen})}\frac{\partial P_\lambda(\mathcal{O}_r|\mathcal{M}_{gen})}{\partial c_{j,m}}\right\} \quad (5)$$

The first term on the RHS of equation (5) is the occupancy count for mixture component m in state j of the model, produced by aligning the training utterances against the model sequence corresponding

to the correct transcription (numerator lattice). Similarly, the second term is the occupancy count from aligning the training data against the recognition model which in our case is conveniently defined by the corresponding denominator lattice. All paths in the numerator lattice built from the correct transcription will also also exist in the denominator lattice. However, these paths will compete against all confusable word sequences (see figure 1). Hence, a positive $\frac{\partial}{\partial c_{j,m}}\mathcal{F}(\lambda)$ indicates that the component occupancy accumulator was often updated during the alignment of the data against the correct transcription. At the same time, the equivalent path in the denominator lattice contributed very little due to its relatively low likelihood compared to other competing paths. This clearly constitutes a discrimination problem which can be corrected by increasing the resolution in such mixture components. The splitting algorithm can be summarised as follows

1. Compute the mixture component occupancy counts according to equation 5;
2. Re-estimate the parameters of the mixture distributions. using equations (2), (3) and (4);
3. Split mixture components with the top n largest positive occupancy counts. This is accomplished by cloning the re-estimated distribution and perturbing the resulting Gaussians by ± 0.2 standard deviations respectively.

5. MMIE FRAMEWORK

The rational nature of the MMIE objective function suggests a logical way of structuring the training process. The discriminative training framework is shown in figure 2. The top left and right branches of the diagram show the calculation of statistics for the numerator/denominator parts of the MMIE objective function respectively. For each training utterance, the numerator or denominator lattice is loaded into the recogniser and reduced to a word graph. Recognition is performed using the current HMM set and the language model scores from the word graph. A new output lattice is then produced containing the original language model scores and new acoustic scores. This is followed by the computation of forward (α) and backward (β) probabilities for each node in the lattice. In a post-processing step, the two sets of statistics are combined to calculate new parameter values according to equations (2), (3) and (4). This is optionally followed by an up-mixing procedure whereby occupancy statistics are used to split selected mixture components in order to enhance modelling resolution.

6. LATTICE GENERATION

Lattices were generated for the Wall Street Journal SI-284 training set using the HTK LVCSR system. The system uses a time-synchronous one-pass decoder that is implemented using a dynamically built tree-structured recognition network. This approach allows the integration of cross-word context-dependent acoustic models and an N-gram language model directly within the search [5].

Each frame of speech is represented by a 39 dimensional feature vector that consists of 12 mel frequency cepstral coefficients, normalised log energy and the first and second differentials of these

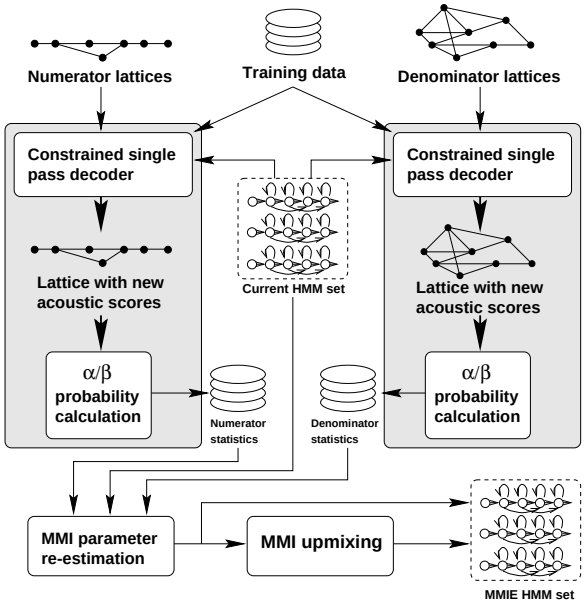

Figure 2: Discriminative training framework incorporating an optional mixture splitting stage.

values. The state clustering algorithm uses decision trees built for every monophone HMM state to determine equivalence classes between sets of triphone contexts.

For the lattice generation on the training data, a 65k word list was created by adding the words occurring in the training set to our standard WSJ recognition lexicon [7]. A corresponding bigram back-off language model was then constructed to accommodate the SI-284 training set which contains utterances with both verbalised and non-verbalised punctuation. The language model contained 4.2 million bigrams estimated from the **nab94** text corpus of 227 million words.

The recognition system (HMM-1) consists of gender independent decision tree state-clustered cross-word triphone HMMs built using the 1993 LIMSI WSJ Lexicon and phone set. There are 6399 unique states with 12 mixture components per state. The system was trained for 4 iterations of MLE on the SI-284 training set of 36,441 utterances. A relatively narrow pruning beam was used which resulted in an average lattice density figure of 15 for the denominator set. Details of this system and its performance on various WSJ test sets is given in [7].

7. LVCSR EXPERIMENTS

Recognition experiments were performed on the 1994 ARPA Hub-1 development and evaluation test sets, and on the 1995 European SQALE project American English evaluation test set. Details of these test sets are given in table 1. The HTK 1994 recognition vocabulary of 65k words was used in conjunction with a bigram back-off language model estimated from the **nab94** text corpus of 227 million words. The presented results were produced by re-scoring recognition lattices originally computed using the HMM-1 system with 12 mixture components per state.

set	task	utterances	speakers	OOV
H1-dev	94 Hub-1	310	20	0.31
H1-eval	94 Hub-1	316	20	0.65
si_et	SQALE AE	200	20	0.39

Table 1: Recognition test sets used in MMIE experiments.

System	Equiv.	si_et	H1-dev	H1-eval
HMM-1(4)	2 mix	14.85	13.94	15.16
HMM-1(16)	4 mix	13.65	13.44	14.04

Table 3: MMIE results for the variable mixture component per state versions of the HMM-1 system and a bigram language model.

7.1. Baseline MMIE Experiments

In these experiments, variants of the HMM-1 system with 1, 2, 4 and 12 mixture components per state were optimised for 4 iterations of MMIE. In all cases, the models were originally trained for 4 iterations using the MLE criterion on the SI-284 data set. Table 2 gives the performance of these systems in terms of % word error rate listed in increasing order of complexity. The results demonstrate that MMIE has provided a worthwhile improvement in the performance of all systems, with larger performance gains observed on the smaller model sets.

Mix	si_et		H1-dev		H1-eval	
	MLE	MMIE	MLE	MMIE	MLE	MMIE
1	19.12	16.46	18.22	15.24	20.44	18.05
2	17.36	15.70	15.58	14.31	17.44	15.96
4	15.26	14.38	14.98	14.06	15.32	14.54
12	12.60	11.90	12.75	12.49	13.43	12.64

Table 2: Baseline MMIE results on the various test sets using the HMM-1 system and a bigram language model.

7.2. MMI Up-mixing Experiments

In these experiments, the single mixture component HMM-1 system was up-mixed in stages to produce two variable mixture variants[1] HMM-1(4) and HMM-1(16) with number of parameters equivalent to the baseline 2 and 4 mixture component systems respectively. The training patterns for these systems were HMM-1(4): **tu-t-tu-t-t** and HMM-1(16): **tu-t-tu-t-tu-t-tu-t-t** where **t** denotes a MMIE pass and **tu** denotes a MMIE + up-mixing pass. Table 3 gives the performance of these systems on the WSJ test sets. On all test sets the performance of the HMM-1(4) system is consistently better than the MMIE-trained equivalent[2] with 2 mixture components per state. Similarly, the HMM-1(16) system has provided improved recognition performance of 3%-5% and 8%-10% when compared to the equivalent 4 mixture component per state MMIE and MLE trained systems respectively.

8. CONCLUSIONS

This paper has described an implementation of the MMIE discriminative training and mixture splitting algorithm based on the use of lattices to compactly represent confusable segments of data. It has been demonstrated that this approach makes it feasible to apply MMIE training to very large HMM-based recognition systems. At the same time, the MMIE algorithm has been shown to provide improvements in recognition performance of up to 16% in the cases tested. In addition, the splitting of mixture components based on the MMI criterion can yield further gains in recognition accuracy and/or provide an effective mechanism for constructing compact and "parameter-efficient" HMM sets.

[1]The number in brackets indicates the maximum number of mixture components per state.

[2]Equivalent in the sense of using equal number of parameters.

ACKNOWLEDGEMENTS

This work is in part supported by an EPSRC/MOD research grant GR/J10204. Additional computational resources were provided by the ARPA CAIP computing facility. Julian Odell gave valuable assistance with the lattice software.

9. REFERENCES

1. L.R. Bahl, P.F. Brown, P.V. de Souza, and R.L. Mercer. Maximum Mutual Information Estimation of Hidden Markov Model Parameters for Speech Recognition. In *Proc. ICASSP'86*. IEEE, 1986.
2. S. Kapadia, V. Valtchev, and S.J. Young. MMI Training for Continuous Phoneme Recognition on the TIMIT Database. In *Proc. ICASSP'93*, volume 2, pages 491–494, Minneapolis, April 1993. IEEE.
3. Y. Normandin. *Hidden Markov Models, Maximum Mutual Information Estimation, and the Speech Recognition Problem*. PhD thesis, Department of Electrical Engineering, McGill University, Montreal, March 1991.
4. Y. Normandin. Optimal Splitting of HMM Gaussian Mixture Components with MMIE Training. In *Proc. ICASSP'95*, volume 1, pages 449–452, Detroit, May 1995. IEEE.
5. J.J. Odell, V. Valtchev, P.C. Woodland, and S.J. Young. A One Pass Decoder Design For Large Vocabulary Recognition. In *Proc. ARPA Human Language Technology Workshop*, pages 405–410. Morgan Kaufmann, March 1994.
6. V.Valtchev, J.J. Odell, P.C. Woodland, and S.J. Young. Lattice-Based Discriminative Training for Large Vocabulary Speech Recognition. In *Proc. ICASSP'96*, volume 2, pages 605–608, Atlanta, May 1996. IEEE.
7. P.C. Woodland, C.J. Leggetter, J.J. Odell, V. Valtchev, and S.J. Young. The 1994 HTK Large Vocabulary Speech Recognition System. In *Proc. ICASSP'95*, volume 1, pages 73–76, Detroit, May 1995. IEEE.