LARGE SCALE DISCRIMINATIVE TRAINING FOR SPEECH RECOGNITION

P.C. Woodland & D. Povey

Cambridge University Engineering Department,
Trumpington Street, Cambridge, CB2 1PZ, UK.
{pcw,dp10006}@eng.cam.ac.uk

ABSTRACT

This paper describes, and evaluates on a large scale, the lattice based framework for discriminative training of large vocabulary speech recognition systems based on Gaussian mixture hidden Markov models (HMMs). The paper concentrates on the maximum mutual information estimation (MMIE) criterion which has been used to train HMM systems for conversational telephone speech transcription using up to 265 hours of training data. These experiments represent the largest-scale application of discriminative training techniques for speech recognition of which the authors are aware, and have led to significant reductions in word error rate for both triphone and quinphone HMMs compared to our best models trained using maximum likelihood estimation. The MMIE lattice-based implementation used; techniques for ensuring improved generalisation; and interactions with maximum likelihood based adaptation are all discussed. Furthermore several variations to the MMIE training scheme are introduced with the aim of reducing over-training.

1. INTRODUCTION

The model parameters in HMM based speech recognition systems are normally estimated using Maximum Likelihood Estimation (MLE). If speech really did have the statistics assumed by an HMM (model correctness) and an infinite training set was used, the global maximum likelihood estimate[1] is optimal in the sense that it is unbiased with minimum variance [19]. However, when estimating the parameters of HMM-based speech recognisers, training data is not unlimited and the true data source is not an HMM. In this case examples can be constructed where alternative *discriminative training* schemes such as the Maximum Mutual Information Estimation (MMIE) can provide better performance than MLE [20].

During MLE training, model parameters are adjusted to increase the likelihood of the word strings corresponding to the training utterances without taking account of the probability of other possible word strings. In contrast to MLE, discriminative training schemes take account of possible competing word hypotheses and try and reduce the probability of incorrect hypotheses (or recognition errors directly). Discriminative schemes have been widely used in small vocabulary recognition tasks, where the relatively small number of competing hypotheses makes training viable e.g. [21, 14, 28]. For large vocabulary tasks, especially on large datasets there are two main problems: generalisation to unseen data in order to increase test-set performance over MLE; and providing a viable computation framework to estimate confusable hypotheses and perform parameter estimation.

The computation problem can be ameliorated by the use of a lattice-based discriminative training framework [30] to compactly encode competing hypotheses. This has allowed investigation of the use of maximum mutual information estimation (MMIE) techniques on large vocabulary tasks and large data sets and a variation of the method described in [30] is used in the work described in this paper.

For large vocabulary tasks, it has often been held that discriminative techniques can mainly be used to produce HMMs with fewer parameters rather than increase absolute performance over MLE-based systems. The key issue here is one of *generalisation* and this is affected by the amount of training data available, the number of HMM parameters estimated, and the training scheme used.

Some discriminative training schemes, such as frame-discrimination [14, 24], try to over-generate training set confusions to improve generalisation. Similarly in the case of MMIE-based training, an increased set of training set confusions can improve generalisation. The availability of very large training sets for acoustic modelling and the computational power to exploit these has also been a primary motivation for us to carry out the current investigation of large-scale discriminative training.

The paper first introduces the MMIE training criterion and its optimisation using the Extended Baum-Welch algorithm. The use of lattices in MMIE training is then described, and the particular methods used in this paper are introduced. Sets of experiments for conversational telephone transcription are presented that show how MMIE training can be successfully applied over a range of training set sizes. The effect of methods to improve generalisation, the interaction with maximum-likelihood adaptation and variations on the basic training scheme to avoid over-training are then discussed.

[1] It should be noted that conventional HMM training schemes only find a local maximum of the likelihood function.

2. MMIE CRITERION

MLE increases the likelihood of the training data given the correct transcription of the training data: models from other classes do not participate in the parameter re-estimation. MMIE training was proposed in [1] as an alternative to MLE and maximises the mutual information between the training word sequences and the observation sequences. When the language model (LM) parameters are fixed during training (as they are in this paper and in almost all MMIE work in the literature), the MMIE criterion is equivalent to Conditional Maximum Likelihood Estimation (CMLE) proposed in [19]. CMLE increases the *a posteriori* probability of the word sequence corresponding to the training data given the training data. However the technique is still normally referred to as MMIE and we use this term in this paper.

For R training observations $\{\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_r, \ldots \mathcal{O}_R\}$ with corresponding transcriptions $\{w_r\}$, the CMLE/MMIE objective function is given by

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

where $\mathcal{M}_w$ is the composite model corresponding to the word sequence w and $P(w)$ is the probability of this sequence as determined by the language model. The summation in the denominator of (1) is taken over all possible word sequences $\hat{w}$ allowed in the task and it can be replaced by

$$p_\lambda(\mathcal{O}_r|\mathcal{M}_{\mathrm{den}}) = \sum_{\hat{w}} p_\lambda(\mathcal{O}_r|\mathcal{M}_{\hat{w}})P(\hat{w}) \qquad (2)$$

where $\mathcal{M}_{\mathrm{den}}$ encodes the full acoustic and language model used in recognition.

It should be noted that optimisation of (1) requires the maximisation of the numerator term $p_\lambda(\mathcal{O}_r|\mathcal{M}_{w_r})$, which is identical to the MLE objective function, while simultaneously minimising the denominator term $p_\lambda(\mathcal{O}|\mathcal{M}_{\mathrm{den}})$. Since the denominator includes all possible word sequences (including the correct one) the objective function has a maximum value of zero. The minimisation of the denominator might ordinarily involve doing a recognition pass on all the training data for each iteration of MMIE training. While this is viable for small vocabulary tasks, it is too computationally expensive for large vocabulary tasks when, for instance, cross-word context dependent acoustic models are used in conjunction with a long span language model. Therefore, an approximation to the denominator is required for the computational load to be feasible.

Another notable feature of the MMIE objective function is that it gives greater weight to training utterances which have a low posterior probability of the correct word sequence. This feature, further discussed in [12, 28], contrasts with the situation in MLE where all training utterances are equally weighted. While it has been argued that MMIE may give undue weight to outlier training utterances, attempts in [28] to modify the criterion to deweight training utterances far from the decision boundary, in a similar way to Minimum Classification Error (MCE) training [2], did not result in improved recognition performance.

3. EXTENDED BAUM-WELCH ALGORITHM

The MMIE objective function can be optimised by any of the standard gradient-based methods although these are either slow to converge or, if using second order information, may be impractical for very large systems. Hence in this work, we have used a version of the Extended Baum-Welch (EBW) algorithm for optimisation.

The EBW algorithm uses re-estimation formulae reminiscent of those used by the standard Baum-Welch algorithm for MLE training. It is shown in [9] that a re-estimation formula of the form

$$\hat{\lambda}_{i,j} = \frac{\lambda_{i,j}(\frac{\partial \mathcal{F}}{\partial \lambda_{i,j}} + D)}{\sum_k \lambda_{i,k}(\frac{\partial \mathcal{F}}{\partial \lambda_{i,k}} + D)} \qquad (3)$$

will converge to give a local optimum of $\mathcal{F}(\lambda)$ for a sufficiently large value of the constant D.

Mean and Variance Updates

For continuous density HMMs, such as used in this work, the formula in (3) does not lead to a closed form solution for the re-estimation of means and variances. However, using a discrete approximation to the Gaussian distribution, Normandin [22] showed that the mean of a particular dimension of the Gaussian for state j, mixture component m, μ_{jm} and the corresponding variance, σ^2_{jm} (assuming diagonal covariance matrices) can be re-estimated by

$$\hat{\mu}_{jm} = \frac{\left\{\theta^{\mathrm{num}}_{jm}(\mathcal{O}) - \theta^{\mathrm{den}}_{jm}(\mathcal{O})\right\} + D\mu_{jm}}{\left\{\gamma^{\mathrm{num}}_{jm} - \gamma^{\mathrm{den}}_{jm}\right\} + D} \qquad (4)$$

$$\hat{\sigma}^2_{jm} = \frac{\left\{\theta^{\mathrm{num}}_{jm}(\mathcal{O}^2) - \theta^{\mathrm{den}}_{jm}(\mathcal{O}^2)\right\} + D(\sigma^2_{jm} + \mu^2_{jm})}{\left\{\gamma^{\mathrm{num}}_{jm} - \gamma^{\mathrm{den}}_{jm}\right\} + D} - \hat{\mu}^2_{jm} \qquad (5)$$

In these equations, the $\theta_{j,m}(\mathcal{O})$ and $\theta_{j,m}(\mathcal{O}^2)$ are sums of data and squared data respectively, weighted by occupancy, for mixture component m of state j, and the Gaussian occupancies (summed over time) are γ_{jm}. The superscripts num and den refer to the model corresponding to the correct word sequence, and the recognition model for all word sequences, respectively.

Setting D

A key issue in using the update equations, (4) and (5), is setting the constant D. If the value set is too large then training is very slow (but stable) and if it is too small the updates may not increase the objective function on each iteration. A useful lower bound on D is the value which ensures that all variances remain positive. In [30] this lower bound constraint was shown to lead to a system of quadratic inequalities to find a suitable value of D, and in fact D was set to twice that value. Furthermore, using a single global value of D can lead to very slow convergence, and in [30] a phone-specific value of D was used.

In preliminary experiments for the work reported here, it was found that the convergence speed could be further improved if D was set on a per-Gaussian level, i.e. a Gaussian specific D_{jm} was used. It was set at the maximum of i) twice the value necessary to ensure positive variance updates for all dimensions of the Gaussian; or ii) a global constant E multiplied by the denominator occupancy γ_{jm}^{den}.

The bulk of the experiments in this paper use a value of $E = 1$. However, in Section 8, the use of other values for E are investigated: either $E = 2$ or a value termed $E = \text{halfmax}$. The latter setting is found by first computing the value of D_{jm} as twice the minimum value for positive variances for each Gaussian and then setting E to half the maximum value of $\frac{D_{jm}}{\gamma_{jm}^{\text{den}}}$ for all Gaussians. The scheme results in a way of setting E that is fairly task and HMM-set independent. When $E = \text{halfmax}$ was used for the experiments in this paper, E increased from about 2 to 6 as training progressed.

Mixture Weight & Transition Probability Updates

The originally proposed re-estimation formula for the mixture weight parameters c_{jm} follows directly from (3)

$$\hat{c}_{jm} = \frac{c_{jm}\left\{\frac{\partial \mathcal{F}}{\partial c_{jm}} + C\right\}}{\sum_{\hat{m}} c_{j\hat{m}}\left\{\frac{\partial \mathcal{F}}{\partial c_{j\hat{m}}} + C\right\}} \tag{6}$$

The constant C is chosen such that all mixture weights are positive. However, the derivative

$$\frac{\partial \mathcal{F}}{\partial c_{jm}} = \frac{1}{c_{jm}}(\gamma_{jm}^{\text{num}} - \gamma_{jm}^{\text{den}}) \tag{7}$$

is extremely sensitive to small-valued parameters. As an alternative, a more robust approximation for the derivative was suggested in [18]:

$$\frac{\partial \mathcal{F}}{\partial c_{jm}} \approx \frac{\gamma_{jm}^{\text{num}}}{\sum_{\hat{m}} \gamma_{j\hat{m}}^{\text{num}}} - \frac{\gamma_{jm}^{\text{den}}}{\sum_{\hat{m}} \gamma_{j\hat{m}}^{\text{den}}} \tag{8}$$

This method was used, for example, by [21, 30]. Unfortunately this update rule can lead to instability as training proceeds and so an alternative was sought.

The alternative mixture weight update rule suggested here is free from smoothing constants, and informal experiments have shown that normally it results in a faster increase in the overall MMIE objective function than the above approach with the derivative approximation in (8).

For a particular state j, the mixture weight update used in this paper consists of finding the mixture weights $\hat{c}_{jm}$ which maximise the following function:

$$\sum_{m=1}^{M} \gamma_{jm}^{\text{num}} \log \hat{c}_{jm} - \frac{\gamma_{jm}^{\text{den}}}{c_{jm}} \hat{c}_{jm} \tag{9}$$

subject to the sum-to-one constraint. In (9), the c_{jm} are the original weights and the γ_{jm} are the mixture component occupancies. A proof that maximising (9) will increase the objective function is given in [25] and uses the assumption that as each mixture weight is varied, the mixture component occupancies that would be obtained from forward-backward alignment will vary by a factor that is between 1 and the ratio of the new to the old mixture weights. For the purposes of the proof, the mixture weight occupancies must also be assumed to be independent of the other parameters in the HMM.

The optimisation of (9) may be performed using a generic function-optimisation routine. However, in the experiments reported here an iterative scheme was used which involves (repeatedly) taking each mixture weight in turn and finding the optimal value of that weight assuming the others' relative values are fixed while maintaining the sum-to-one constraint.

The update equation for a single row of a transition matrix is performed in the same way as the mixture weight update. Note that for both the mixture weights and transition probabilities, if the denominator occupancies are zero the update is equivalent to the standard MLE update.

It should be noted that for the decision-tree tied-state mixture Gaussian HMMs used in the experiments reported here, the effect of MMIE training on the mixture weights (and hence the mixture weight update itself) is relatively unimportant. Of course, for an HMM system using tied mixture models, the mixture weight update rule is of much greater significance.

4. LATTICE-BASED MMIE TRAINING: PREVIOUS WORK

The parameter re-estimation formulae presented in Section 3 require the generation of occupation and weighted data counts for both the numerator terms which rely on using the correct word sequence and the denominator terms which use the recognition model.

The calculation of the denominator terms directly is computationally very expensive and so approximations to the denominator have been suggested. Early work suggested using N-best lists [3] which are calculated once (from an MLE model set) to approximate the set of possible sentences during MMIE training. However, for even moderately complex tasks and long sentences only a very small number of the probable sentences will be included. An alternative is to use some type of lattice structure to represent the various likely alternatives. In [23] a looped lattice model was proposed which could include any pronunciation of a particular word at most once. The approach was evaluated using a 2000 word task with a few hours of training data.

A more sophisticated approach to the use of word lattices that fully encode sequential acoustic and language model constraints was presented in [29, 30]. The lattices used were generated by the HTK large vocabulary recognition system [31]. The HTK lattices are composed of nodes which represent the ends of words at particular points in time and the arcs that connect these represent particular word pronunciations. The denominator lattices often contain repeated

arcs/nodes to encode slightly different start/end times and different start/end context-dependent HMMs due to variant previous/following words. Lattices are generated once using an MLE HMM set, and then used repeatedly for several iterations of MMIE training. The technique also uses lattices for collecting the numerator statistics to represent the possibility of alternative pronunciations. In cases where the recogniser-generated denominator lattices did not contain the correct sequence, the denominator lattice was formed by merging the recogniser lattice with the numerator lattice.

Given word lattices for the numerator and denominator, the technique in [30] performed at each iteration a forward-backward pass at the word lattice node/arc level to generate the posterior probability of a particular lattice arc occurring. The Viterbi state-level segmentation for for each arc was found, and used with the arc posterior probability to calculate the statistics for the EBW re-estimation formulae. The method was used to train HMM sets for up to 65k word vocabulary tasks for the North American Business News corpus using cross-word triphone acoustic models, N-gram LMs, and up to 66 hours of training data.

5. IMPROVING MMIE GENERALISATION

A key issue in MMIE training (and discriminative training in general) is the generalisation performance i.e. the difference between training set and test set accuracy. While MMIE training often greatly reduces training set error from an MLE baseline, the reduction in error rate on an independent test set is normally much less, i.e., compared to MLE, the generalisation performance is poorer. Furthermore, as with all statistical modelling approaches the more complex the model the poorer the generalisation. Since fairly complex models are needed to obtain optimal performance with MLE, it can be difficult to improve these with MMIE training. Therefore it has been widely thought that the major application of discriminative training techniques to large vocabulary recognition tasks is to reduce error rates when relatively few parameters are used rather than to improve the best achievable error rates from MLE training: this paper is aimed at challenging that view.

There have been a number of approaches to try to improve generalisation performance for MMIE-type training schemes, some of which are discussed below. These methods involve trying to increase the amount of confusable data processed during training in some way. The Frame Discrimination (FD) technique, that we have previously investigated, is discussed first. In this paper we have experimented with two other techniques aimed at improving generalisation: weaker language models and acoustic model scaling.

Frame Discrimination

Frame Discrimination (FD) [14] replaces the recognition model probability in the denominator of (1) with all Gaussians in parallel.[2] FD therefore removes many constraints that make some Gaussian sequences very unlikely (phone model, lexical and LM) but provides far more "confusable" states for any particular utterance. This in turn, as would be expected, reduces training set performance compared to MMIE but improves generalisation. In [24] it was shown that the improvements obtained by FD were at least as good as those reported by MMIE using the same models and task setup in [30]. It could be argued that FD over-generalises the confusable data set by modelling confusions that will never in practice arise, and will perform more poorly for the most challenging recognition tasks with greater inherent acoustic confusability. It was reported in [32] that FD didn't improve error rates over MLE trained models for a broadcast news recognition task.

Weakened Language Models

In [27] it was shown that improved test-set performance could be obtained using a unigram LM during MMIE training, even though a bigram or trigram was used during recognition.[3] The aim is to provide more focus on the discrimination provided by the acoustic model by loosening the language model constraints. In this way, more confusable data is generated which improves generalisation. The use of a unigram LM during MMIE training is further investigated in this paper.

Acoustic Model "Scaling"

When combining the likelihoods from an HMM-based acoustic model and the LM it is usual to scale the LM log probability. This is necessary because, primarily due to invalid modelling assumptions, the HMM underestimates the probability of acoustic vector sequences leading to a very wide dynamic range of likelihood values.

An alternative to LM scaling is to multiply the acoustic model log likelihood values by the inverse of the LM scale factor (acoustic model scaling). This will produce the same effect as language model scaling when considering only a single word sequence as for Viterbi decoding.[4] However, when likelihoods from different sequences are added, such as in the forward-backward algorithm or for the denominator of (1), the effects of LM and acoustic model scaling are very different. If language model scaling is used, one particular state-sequence tends to dominate the likelihood at any point in time and hence dominates any sums using path likelihoods. However, if acoustic scaling is used, there will be several paths that have fairly similar likelihoods which make a non-negligible contribution to the summations. Therefore acoustic model scaling tends to increase the confusable data set in training by broadening the posterior distribution of state occupation γ_{jm}^{den} that is used in the EBW update equations. This increase in confusable data also leads to improved generalisation performance.

It should be noted that acoustic scaling is used for similar reasons when finding word posterior probabilities from lat-

[2] A unigram Gaussian level language model based on training set occurrences is used.

[3] Although a unigram was used in MMIE training, the confusable data was also constrained by the word lattices used which were generated with a trigram LM.

[4] The acoustic model and LM scaling effects will be identical for the Viterbi path only if all components of the acoustic model log likelihood are scaled including the contribution from transition probabilities.

tices [17, 4] which are used for either posterior decoding or confidence estimation.

6. CURRENT LATTICE-BASED TRAINING METHODS

The lattice-based training technique used in this paper is based on that in [30] but has various differences in detail. Furthermore several variants of the current scheme have been investigated.

The first step is to generate word-level lattices, normally using an MLE-trained HMM system and a bigram LM appropriate for the training set. This step is normally performed just once and for the experiments in Section 7 the word lattices were generated in about 5x Real-Time (RT).[5]

The second step is to generate *phone-marked* lattices which label each word lattice arc with a phone/model sequence and the Viterbi segmentation points. These are are found from the word lattices and a particular HMM set, which may be different to the one used to generate the original word-level lattices. In our implementation, these phone marked lattices also encode the LM probabilities used in MMIE training which again may be different to the LM used to generate the original word-level lattices. This stage typically took about 2xRT to generate triphone-marked lattices for the experiments in Section 7, although the speed of this process could be considerably increased.

Given the phone-marked lattices for the numerator and denominator of each training audio segment, two alternative implementations have been used to generate the Gaussian-level occupation probabilities and associated weighted-data statistics needed for EBW updates. The *full-search* implementation aims to perform a full forward-backward pass at the state-level constrained by the lattice. Pruning is performed by using the phone-marked lattice segmentation points extended by a short-period in each direction.[6] However in the alternative *exact-match* case, a state-level forward-backward pass for each context-dependent model instance in the lattice is performed solely between the Viterbi segmentation points for each model. In both cases, the search was also optimised as far as possible by combining redundantly repeated models which first requires the conversion to a model-level lattice. For the recognition experiments in this paper, these model-level lattices typically have an average lattice density of several hundred arcs. Different optimisations were possible in the two cases and these are discussed below.

Details of the Full-Search Implementation

For the full-search case, the model-level lattice is compacted by combining instances of the same model which occur in the same position in the same word and overlap in time. A single instance of the model is created with start/end times the minimum/maximum of the two original models. The set of arcs entering/leaving the new combined arc is set to the union of the original sets of transitions, with duplicates removed. This process of lattice reduction is repeated until no further merges are possible and decreases the average lattice density by up to an order of magnitude. A full forward-backward search on the resulting lattice is then performed, with the time information for each phone, extended by a small margin, used for pruning. The acoustic likelihood scaling is performed by directly scaling the values of the state output distribution log probability densities. Typically, the full-search method takes about 1xRT per iteration for the experiments in Section 7.

Details of the Exact-Match Implementation

The exact-match approach calculates the likelihood of each phone segment in the lattice, based on its start and end times, and then accumulates statistics for the EBW updates using the forward-backward algorithm. There are two possible advantages to this approach. Firstly, only one forward-backward pass is necessary for a given model with given start and end times, no matter how many times it appears in the lattice and hence the exact-match typically runs twice as quickly as the full-search method. Secondly, the segment-level acoustic log likelihoods can be scaled as a whole which keeps multiple parallel confusable models while retaining sharp transitions between states. However, the fact that the segmentation times in the phone-marked lattices are treated as constants across multiple iterations of MMIE training could lead to reduced accuracy.

7. MMIE EXPERIMENTS WITH HUB5 DATA

This section describes a series of MMIE-training experiments using the Cambridge University HTK (CU-HTK) system for the transcription of conversational telephone data from the Switchboard and Call Home English corpora ("Hub5" data). These experiments were performed in preparation for the NIST March 2000 Hub5 Evaluation.

The experiments investigated the effect of different training set and HMM set sizes and types; the use of acoustic likelihood scaling and unigram LMs in training and any possible interactions between MMIE training and maximum likelihood linear regression-based adaptation. All the experiments in this section used the full-search lattice-training implementation and a value of $E = 1$ to set the Gaussian-specific D for EBW updates. The effect of alternatives will be discussed in Section 8.

Basic CU-HTK Hub5 System

The CU-HTK Hub5 system is a continuous mixture density, tied-state cross-word context-dependent HMM system based on the HTK HMM Toolkit. The full system operates in multiple passes, using more complex acoustic and language models and unsupervised adaptation in later passes.

Incoming speech is parameterised into cepstral coefficients and their first and second derivatives to form a 39 dimensional vector every 10ms. Cepstral mean and variance normalisation and vocal tract length normalisation is performed for each conversation side in both training and test.

[5] All run times are measured on an Intel Pentium III running at 550MHz.
[6] Typically 50ms at both the start and end of each phone.

The HMMs are constructed using decision-tree based state-clustering [33] and both triphone and quinphone models can be used. The lexicon used in the experiments below was either a 27k vocabulary (as used in [10]) or a 54k vocabulary and the core of this dictionary is based on the LIMSI 1993 WSJ lexicon. The system uses word-based N-gram LMs estimated from an interpolation of Hub5 acoustic training transcriptions and Broadcast News texts. In the experiments reported here, trigram LMs are used unless otherwise stated.

The system operates in multiple passes. Triphone models are used in word lattice generation. The lattices are used for both later recognition passes and also during system development. Lattice rescoring was used to generate many of the results given below.

Baseline Models and Hub5 Training/Test Data

Three different training sets and three different test sets were used in the MMIE experiments. The different training sets, ranging from 18 hours to 265 hours in size were used to investigate how well the MMIE approach scales to very large training sets while still allowing many experiments to be run.

The characteristics of the three training sets are shown in Table 1. The Minitrain set, defined by BBN, used BBN-provided transcriptions, while the h5train00 sets used transcriptions based on those provided by Mississippi State University (MSU). All the training sets contain data from the Switchboard I (SWB1) corpus and the h5train00 sets also contain Call Home English (CHE) data. The h5train00sub set is a subset of h5train00 and covers all of the training speakers in the SWB1 portion of h5train00, and a subset of CHE.

Training Set	Total Time (hrs)	Conversation Sides	
		SWB1	CHE
Minitrain	18	398	–
h5train00sub	68	862	92
h5train00	265	4482	235

Table 1: Hub5 training sets used.

The test sets used were a subset of the 1997 Hub5 evaluation set, eval97sub, containing 10 conversation sides of Switchboard II (SWB2) data and 10 of CHE; the 1998 evaluation data set, eval98, containing 40 sides of SWB2 and 40 CHE sides (in total about 3 hours of data) and the March 2000 evaluation data set, eval00, which has 40 sides of SWB1 and 40 CHE sides.

Training Set	Number of Speech States	Gaussians per state	Gaussians per hour
Minitrain	3088	12	2060
Minitrain	3088	6	1030
h5train00sub	6165	12	1090
h5train00	6165	16	370

Table 2: Hub5 Triphone Model Sets

Baseline gender independent sets of triphone HMMs were created for each training set and trained using MLE. The number of clustered speech states in each triphone model set; the number of Gaussians per state; and the average number of Gaussians to be trained per hour of training data is given in Table 2. Note that there are two versions of the MLE model set for Minitrain.

Experiments with 18 Hours Training

Initially we investigated MMIE training using Minitrain with 12 Gaussian/state HMMs which were our best MLE trained models. Lattices were generated on the training set using a bigram LM. The bigram 1-best hypotheses had a 24.6% word error rate (WER) and a Lattice WER (LWER) [31] of 6.2%.

MMIE Iteration	%WER	
	Acoustic Scaling	LM Scaling
0 (MLE)	50.6	50.6
1	50.2	51.0
2	49.9	51.3
3	50.5	51.4
4	50.9	–

Table 3: 18 hour experiments with 12 mixture component models (eval97sub): comparison of acoustic model and language model scaling.

The Minitrain 12 Gaussian/state results given in Table 3 compare acoustic and language model scaling for several iterations of MMIE training. It can be seen that acoustic scaling helps avoid over-training and the best WER is after 2 iterations. The training set lattices regenerated after a single MMIE iteration gave a WER of 16.8% and a LWER of 3.2%, showing that the technique is very effective in reducing training set error. However, it was found that these regenerated lattices were no better to use in subsequent training iterations and so all further work used just the initially generated word lattices.

The advantage from MMIE training for the 12 Gaussian per state system is small and so a system with fewer Gaussians per state was investigated. As shown in Table 2 the 6 Gaussian system has approximately the same ratio of parameters to training data as our h5train00sub system.

MMIE Iteration	%WER	
	Lattice Bigram	Lattice Unigram
0 (MLE)	51.5	51.5
1	50.0	49.7
2	49.8	49.6
3	50.1	50.0
4	50.8	–

Table 4: 18 hour experiments with 6 mixture component models (eval97sub): comparison of lattice LMs.

The results from MMIE training of the 6 Gaussian/state Minitrain system (with acoustic scaling) are shown in Table 4 and again show the best performance after two MMIE

iterations. Furthermore the gain over the MLE system is 1.7% absolute if a bigram LM is used and 1.9% absolute if a unigram LM is used: the 6 Gaussian per state MMIE-trained HMM set now slightly outperforms the 12 Gaussian system. Furthermore it can be seen that using a weakened LM (unigram) improves performance a little and in fact the gain from using a unigram is greater if no acoustic scaling is performed: both acoustic scaling and the weakened LM increase the amount and diversity of confusable data.

Experiments with 68 Hours Training

The effect of using the 68 hour h5train00sub set was investigated next and tests were performed on both the eval97sub and eval98 sets. In this case the phone-marked denominator lattices had a LWER of 7.4%. The results of MMIE training are shown in Table 5.

MMIE Iteration	%WER	
	eval97sub	eval98
0 (MLE)	46.0	46.5
1	43.8	45.0
2	43.7	44.6
3	44.1	44.7

Table 5: Word error rates on eval97sub and eval98 using h5train00sub training.

Again it can be seen that the peak improvement comes after two iterations, but in this case there is an even larger reduction in error rate than was seen for the 6 Gaussian/state Minitrain experiments: 2.3% absolute on eval97sub and 1.9% absolute on eval98. The word error rate for the 1-best hypothesis from the original bigram word lattices measured on 10% of the training data was 27.4%. The MMIE models obtained after two iterations on the same portion of training data gave an error rate of 21.2%, so again MMIE provided a very sizeable reduction in training set error.

Further experiments using this same training set/baseline model set are given in Section 8.

Triphone Experiments with 265 Hours Training

The good performance on smaller training sets led us to investigate MMIE training using all the available Hub5 data: the 265 hour h5train00 set. The h5train00 set contains 267,611 segments and numerator and denominator word level lattices were created for each trained segment, and from these, phone-marked lattices were generated.

MMIE Iteration	%WER	
	eval97sub	eval98
0 (MLE)	44.4	45.6
1	42.4	43.7
1 (3xCHE)	42.0	43.5
2	41.8	42.9
2 (3xCHE)	41.9	42.7

Table 6: Word error rates when using h5train00 training with and without CHE data weighting (3xCHE).

We also experimented with data-weighting with this setup during MMIE training. The rationale for this is that while the test data sets contain equal amounts of Switchboard and CHE data, the training set is not balanced. Therefore we gave a 3x higher weighting to CHE data during training. The results of these experiments on both the eval97sub and eval98 test sets are shown in Table 6. It can be seen that without data weighting there is an improvement in WER of 2.6% absolute on eval97sub and 2.7% absolute on eval98.

Data weighting gives a further 0.2% absolute on eval98, but rather variable results on eval97sub. However if data weighting is applied during MLE training for eval97sub the MLE baseline improves by 0.7% absolute. It might be concluded that the extra weight placed on poorly recognised data by MMIE training relative to MLE reduces the need for the data weighting technique.

Quinphone Model Training

Since the CU-HTK Hub5 system also uses quinphone models, we also investigated MMIE training of these models using the full h5train00 set. The decision tree state clustering process for quinphones includes questions regarding ± 2 phone context and word-boundaries. The baseline quinphone system uses 9640 speech states and 16 Gaussians per state to give 580 Gaussians per hour of training data.

The quinphone MMIE training used triphone-generated word lattices, but, since the phone-marked lattices were regenerated for the quinphone models, it was necessary to further prune the word-lattices. The results of MMIE trained quinphones on the eval97sub set are shown in Table 7. Note that these experiments, unlike all previous ones reported here, include pronunciation probabilities.

MMIE Iteration	%WER eval97sub
0 (MLE)	42.0
1	40.4
2	39.9
3	40.1

Table 7: Quinphone MMIE results on eval97sub. Pronunciation probabilities were used.

As with the MMIE training runs discussed above, the largest WER reduction (2.1% absolute) comes after two iterations of training. While MMIE training is still working well, the reductions in error rate are not quite as large as for the triphone models. This may be because of the extra pruning required for the phone-marked lattices, or because there are rather more HMM parameters to estimate.

Interaction with MLLR

All the above results used models that were not adapted to the particular conversation side. Since model adaptation by parameter transformation using maximum likelihood linear regression (MLLR) [15, 6] is now a well-established technique, it is important to investigate if there is an interaction between the MMIE trained models and transformation parameters estimated using MLE.

To measure MLLR adaptation performance, MMIE and MLE models (both using CHE data weighting) were used in a full-decode of the test data, i.e. not rescoring lattices, with a 4-gram language model. The output from this first pass was used to estimate a global speech MLLR (block-diagonal mean and diagonal variance) transform. If enough data was available a separate transform was also estimated for silence models and the output from the respective non-adapted pass was used for adaptation supervision. The adapted models were then used for a second full-decode pass. The results of these experiments are shown in Table 8.

Adaptation	% WER eval98	
	MLE	MMIE
None	44.6	42.5
MLLR	42.1	39.9

Table 8: Effect of MLLR on MLE and MMIE trained models.

The results show that the MMIE models are 2.1% absolute better than the MLE models without MLLR, and 2.2% better with MLLR. In this case, MLLR seems to work just as well with MMIE trained models: a relatively small number of parameters are being estimated with MLLR and these global transforms keep the Gaussians in the same "configuration" as optimised by MMIE.

March 2000 CU-HTK Hub5 System

The MMIE triphone and quinphone models were included in the March 2000 CU-HTK Hub5 evaluation system [11]. Although this system incorporates numerous changes compared to that described in [10], the use of MMIE models in the system gave the greatest benefit.

Initial lattices were generated using gender independent MMIE triphone HMMs with a 54k vocabulary and a 4-gram language model. Subsequent passes through the data used MMIE triphones and quinphones as well as MLE gender-dependent soft-tied [16] triphones and quinphones. All model sets use pronunciation probabilities, iterative MLLR adaptation combined with a global full-variance transform [7]. The final system output for each model set was generated to minimise the expected word error rate via confusion networks [17]. The output of the MMIE and MLE model stages were combined via confusion network combination [5] to give the final output.

On the eval98 data, this system gives an error rate of 35.0%, and on the March 2000 evaluation data (eval00) 25.4%, which was the lowest error rate obtained in the evaluation by a statistically significant margin.[7]

8. FURTHER INVESTIGATION OF THE MMIE TRAINING SCHEME

In this section, the properties of the MMIE training scheme used in Section 7 are investigated along a number of variations. These include the effect of acoustic likelihood scaling on the number of confusable states; the use of the exact-match and full-search lattice processing methods; the effect of different values of the global constant E on optimisation and test-set performance; and a brief investigation into a modified objective function.

Increased Confusion Data by Acoustic Model Scaling

To illustrate the effect of acoustic scaling (rather than language model scaling) on the distribution of the posterior probability of state-occupation, the average number of states with a posterior probability greater than 0.01 was computed for both the full-search and the exact-match lattice search procedures. The results are shown in Table 9.

Search Type	Scaling			
	Acoustic		LM	
	num	den	num	den
Full-search	3.54	8.16	1.43	1.63
Exact-match	1.78	5.58	1.26	1.45

Table 9: Average number of states with a posterior probability of occupation greater than 0.01 with and without acoustic scaling.

As expected, acoustic likelihood scaling significantly broadens the posterior probability distribution. It is also noteworthy that the exact-match procedure reduces the number of confusable states quite markedly since models are not computed outside the lattice arc Viterbi segmentation points.

Objective Function Optimisation and Generalisation

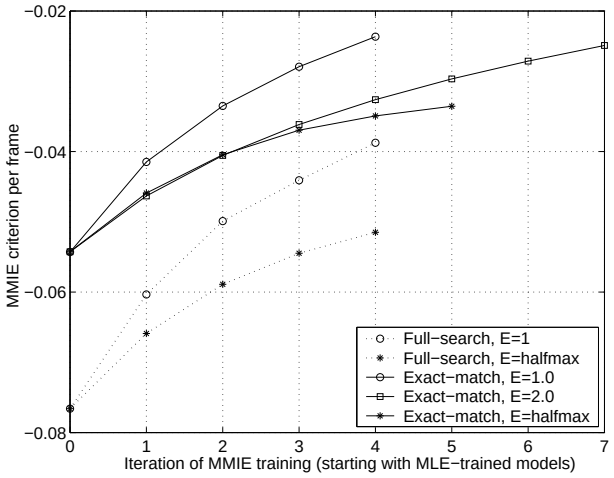

Figure 1: MMIE criterion optimisation.

The increase in MMIE objective function and the corresponding test-set error rate (eval97sub) were measured using both the full-search and exact-match schemes and also several values of the global smoothing constant: $E = 1$, $E = 2$, and $E = \mathrm{halfmax}$. The experiments used the 68 hour h5train00sub training setup with acoustic scaling. The change in objective function as training proceeds is shown in Figure 1 and the corresponding error rates in Figure 2. While there is no consistent difference in WER between

[7]The eval00 test-set consistently yields much lower error rates than eval98 across all recognition systems.

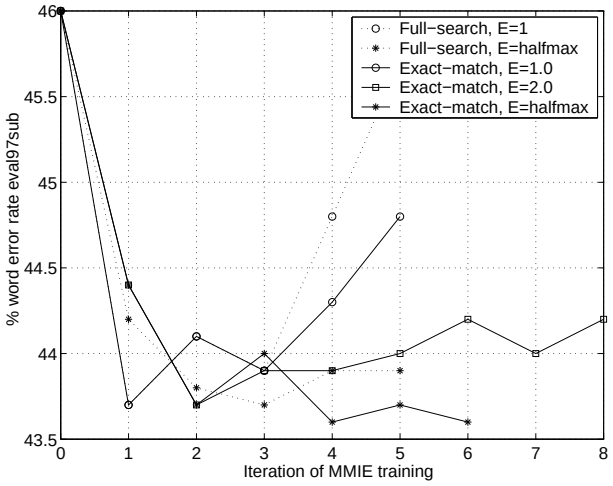

Figure 2: Error rates for several MMIE training variants.

full-search and exact-match, our implementation of exact-match search ran significantly faster.

One problem with using $E = 1$ is that over-training easily occurs, although the second iteration of MMIE training yields good results. Using a higher value of the global smoothing constant E which further increases during training, such as $E = \mathrm{halfmax}$, results in the objective function being optimised to a poorer final value, but with less danger of over-training. However, the underlying problem is that improving the objective function past a certain point causes the test-set accuracy to deteriorate.

H-Criterion Objective Function

An alternative solution to over-training is to modify the objective function. In particular, an interpolation of the MMIE and MLE objective functions, which gives a type of H-criterion [8], was examined. The function investigated here was

$$0.8\mathcal{F}_{\mathrm{MMIE}} + 0.2\mathcal{F}_{\mathrm{MLE}}$$

This objective function can be implemented simply by an appropriate scaling of the MMIE numerator statistics. The exact-match method was used on the h5train00sub training set with $E = 1$. Evaluation using the eval97sub test-set showed that the error-rate converged as the objective function was optimised to yield 43.7% error on the 5th iteration.

While these models gave the same test-set accuracy as conventional MMIE training, it was noted that the model parameters had changed rather less from the MLE parameter values than the pure MMIE ones with the same accuracy: 90% of means were within 0.1 standard deviations of the MLE values, compared to 90% within 0.25 standard deviations for similarly performing pure MMIE models.

9. DISCUSSION & CONCLUSIONS

This paper has discussed the use of discriminative training for large vocabulary HMM-based speech recognition for a training set size and level of task difficulty not previously attempted. It has been shown that significant reductions in word error rates can be obtained for the transcription of conversational telephone speech.

The MMIE objective function was reviewed and the two key issues for its application to large vocabulary tasks were discussed: the efficiency of objective function optimisation and generalisation to test data.

The Extended Baum Welch algorithm, with Gaussian specific D constants, was used and it was shown that two iterations of updating were sufficient to obtain good performance over a large range of data set sizes and model types. Furthermore, a novel updating formula for the mixture weight parameters was introduced.

The use of a weakened language model (a unigram), and, more importantly, acoustic likelihood scaling were investigated as methods of increasing the amount of relevant confusable data during MMIE training. Both these techniques improve generalisation and allow better performance to be obtained with MMIE training using more complex models. Therefore, in contrast to previously held beliefs, it is possible to use MMIE training for the most challenging large vocabulary tasks to reduce error rates over the best MLE models, and not just provide good performance with a reduced number of parameters.

A lattice-based approach to calculating the statistics related to the objective function denominator was used, and two specific implementations of lattice search were described. Both methods, unlike previous work on lattice-based discriminative training algorithms, perform a full forward-backward pass at the model level. However they differ in the constraints used at the model boundaries and were found to be comparable in error rate, although the exact-match scheme has a lower computational cost.

While MMIE training is effective, it is clear that over-training can easily occur. One possible solution is to modify the objective function to aid generalisation directly. One method for doing this is to use an interpolation of the MMIE and MLE objective functions and this seems to be effective. We intend to further investigate other modifications to improve generalisation performance.

While this paper has concentrated on the MMIE objective function, much of what has been discussed can be directly applied to other objective functions. A general formulation to lattice-based discriminative training was proposed in [26], which discuses how other measures, such as MCE, can be used in the lattice framework.

The MMIE training scheme was applied to transcription of Hub5 data for training sets up to 265 hours in size for both triphone and quinphone models and resulted in a 2-3% absolute reduction in word error rate. The trained MMIE triphone and quinphone HMMs were used in the March 2000 CU-HTK Hub5 system which had the lowest error rate in the evaluation by a statistically significant margin. While the method is still very computationally expensive, it is now becoming feasible to investigate MMIE training on this scale. We believe that there is much exciting research on large-scale discriminative training still to be done.

ACKNOWLEDGMENTS

This work is in part supported by a grant from GCHQ. Dan Povey holds a studentship from the Schiff Foundation.

REFERENCES

[1] L.R. Bahl, P.F. Brown, P.V. de Souza & R.L. Mercer (1986). Maximum Mutual Information Estimation of Hidden Markov Model Parameters for Speech Recognition, *Proc. ICASSP'86*, pp. 49–52, Tokyo.

[2] W. Chou, C.-H. Lee & B.-H. Juang (1993). Minimum Error Rate Training Based on N-Best String Models. *Proc. ICASSP'93*, pp. 652-655, Minneapolis.

[3] Y.L. Chow (1990). Maximum Mutual Information Estimation of HMM Parameters for Continuous Speech Recognition Using the N-Best Algorithm. *Proc. ICASSP'90*, Albuquerque.

[4] G. Evermann & P.C. Woodland (2000). Large Vocabulary Decoding and Confidence Estimation using Word Posterior Probabilities. *Proc. ICASSP'2000*, Istanbul.

[5] G. Evermann & P.C. Woodland (2000). Posterior Probability Decoding, Confidence Estimation and System Combination. *Proc. Speech Transcription Workshop*, College Park.

[6] M.J.F. Gales & P.C. Woodland (1996). Mean and Variance Adaptation Within the MLLR Framework. *Computer Speech & Language*, Vol. 10, pp. 249–264.

[7] M.J.F. Gales (1998). Maximum Likelihood Linear Transformations for HMM-Based Speech Recognition. *Computer Speech & Language*, Vol. 12, pp. 75-98.

[8] P.S. Gopalakrishnan, D. Kanevsky, A. Nádas, D. Nahamoo & M.A. Picheney (1988). Decoder Selection Based on Cross-Entropies, *Proc. ICASSP'88*, pp. 20–23, New York.

[9] P.S. Gopalakrishnan, D. Kanevsky, A. Nádas & D. Nahamoo (1991). An Inequality for Rational Functions with Applications to Some Statistical Estimation Problems. *IEEE Trans. Information Theory*, Vol. 37, pp. 107–113.

[10] T. Hain, P.C. Woodland, T.R. Niesler & E.W.D. Whittaker (1999). The 1998 HTK System for Transcription of Conversational Telephone Speech. *Proc. ICASSP'99*, pp. 57–60, Phoenix.

[11] T. Hain, P.C. Woodland, G. Evermann & D. Povey (2000). The CU-HTK March 2000 Hub5E Transcription System. *Proc. Speech Transcription Workshop*, College Park.

[12] M.M. Hochberg, L.T. Niles, J.T. Foote & H.F. Silverman (1991). Hidden Markov Model/Neural Network Training Techniques for Connected Alpha-Digit Speech Recognition. *Proc. ICASSP'91*, pp. 109–112, Toronto.

[13] S. Kapadia, V. Valtchev & S.J. Young (1993). MMI Training for Continuous Parameter Recognition of the TIMIT Database. *Proc. ICASSP'93*, pp. 491–494, Minneapolis.

[14] S. Kapadia (1998). *Discriminative Training of Hidden Markov Models*. Ph.D. Thesis, Cambridge University Engineering Dept.

[15] C.J. Leggetter & P.C. Woodland (1995). Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density HMMs. *Computer Speech & Language*, Vol. 9, pp.171–186.

[16] X. Luo & F. Jelinek (1999). Probabilistic Classification of HMM States for Large Vocabulary Continuous Speech Recognition *Proc. ICASSP'99*, pp. 2044–2047, Phoenix.

[17] L. Mangu, E. Brill & A. Stolcke (1999). Finding Consensus Among Words: Lattice-Based Word Error Minimization. *Proc. Eurospeech'99*, pp. 495–498, Budapest.

[18] B. Merialdo (1988). Phonetic Recognition Using Hidden Markov Models and Maximum Mutual Information Training. *Proc. ICASSP'88*, pp. 111–114, New York.

[19] A. Nádas (1983). A Decision Theoretic Formulation of a Training Problem in Speech Recognition and a Comparison of Training by Unconditional Versus Conditional Maximum Likelihood. *IEEE Trans. ASSP*, Vol. 31, pp. 814–817.

[20] A. Nádas, D. Nahamoo & M.A. Picheny (1988). On a Model-Robust Training Algorithm for Speech Recognition. *IEEE Trans. ASSP*, Vol. 36, pp. 1432–1435.

[21] Y. Normandin (1991). An Improved MMIE Training Algorithm for Speaker Independent, Small Vocabulary, Continuous Speech Recognition. *Proc. ICASSP'91*, pp. 537–540, Toronto.

[22] Y. Normandin (1991). *Hidden Markov Models, Maximum Mutual Information Estimation and the Speech Recognition Problem.* Ph.D. Thesis, McGill University, Montreal.

[23] Y. Normandin, R. Lacouture & R. Cardin (1994). MMIE Training for Large Vocabulary Continuous Speech Recognition. *Proc. ICASLP'94*, pp. 1367-1371, Yokohama.

[24] D. Povey & P.C. Woodland (1999). Frame Discrimination Training of HMMs for Large Vocabulary Speech Recognition. *Proc. ICASSP'99*, pp. 333–336, Phoenix.

[25] D. Povey & P.C. Woodland (1999). *An Investigation of Frame Discrimination for Continuous Speech Recognition.* Technical Report CUED/F-INFENG/TR.332, Cambridge University Engineering Dept.

[26] R. Schlüter & W. Macherey (1998). Comparison of Discriminative Training Criteria. *Proc. ICASSP'98*, pp. 493–496, Seattle.

[27] R. Schlüter, B. Müller, F. Wessel & H. Ney (1999). Interdependence of Language Models and Discriminative Training. *Proc. IEEE ASRU Workshop*, pp. 119–122, Keystone, Colorado.

[28] V. Valtchev (1995). *Discriminative Methods in HMM-Based Speech Recognition*. Ph.D. Thesis, Cambridge University Engineering Dept.

[29] V. Valtchev, P.C. Woodland & S.J. Young (1996). Lattice-based Discriminative Training for Large Vocabulary Speech Recognition. *Proc. ICASSP'96*, pp. 605–608, Atlanta.

[30] V. Valtchev, J.J. Odell, P.C. Woodland & S.J. Young (1997). MMIE Training of Large Vocabulary Speech Recognition Systems. *Speech Communication*, Vol. 22, pp. 303–314.

[31] P.C. Woodland, C.J. Leggetter, J.J. Odell, V. Valtchev & S.J. Young (1995). The 1994 HTK Large Vocabulary Speech Recognition System. *Proc. ICASSP'95*, pp. 73–76, Detroit.

[32] P.C. Woodland, T. Hain, G.L. Moore, T.R. Niesler, D. Povey, A. Tuerk & E.W.D. Whittaker (1999). The 1998 HTK Broadcast News Transcription System: Development and Results. *Proc. DARPA Broadcast News Workshop*, pp. 265-270, Morgan Kaufmann.

[33] S.J. Young, J.J. Odell & P.C. Woodland (1994). Tree-Based State Tying for High Accuracy Acoustic Modelling. *Proc. 1994 ARPA Human Language Technology Workshop*, pp. 307-312, Morgan Kaufmann.