Automatic Transcription of Conversational Telephone Speech

Thomas Hain, *Member, IEEE*, Philip C. Woodland, *Member, IEEE*, Gunnar Evermann, *Student Member, IEEE*, Mark J. F. Gales, *Member, IEEE*, Xunying Liu, *Student Member, IEEE*, Gareth L. Moore, Dan Povey, and Lan Wang, *Student Member, IEEE*

***Abstract*—This paper discusses the Cambridge University HTK (CU-HTK) system for the automatic transcription of conversational telephone speech. A detailed discussion of the most important techniques in front-end processing, acoustic modeling and model training, language and pronunciation modeling are presented. These include the use of conversation side based cepstral normalization, vocal tract length normalization, heteroscedastic linear discriminant analysis for feature projection, minimum phone error training and speaker adaptive training, lattice-based model adaptation, confusion network based decoding and confidence score estimation, pronunciation selection, language model interpolation, and class based language models.**

The transcription system developed for participation in the 2002 NIST Rich Transcription evaluations of English conversational telephone speech data is presented in detail. In this evaluation the CU-HTK system gave an overall word error rate of 23.9%, which was the best performance by a statistically significant margin. Further details on the derivation of faster systems with moderate performance degradation are discussed in the context of the 2002 CU-HTK 10 $\times$ RT conversational speech transcription system.

***Index Terms*—Large-vocabulary conversational speech recognition, telephone speech recognition.**

I. Introduction

THE transcription of conversational telephone speech is one of the most challenging tasks for speech recognition technology. State-of-the-art systems still yield high word error rates typically within a range of 20%–30%. Work on this task has been aided by extensive data collection, namely the Switchboard-1 corpus [10]. Originally designed as a resource to train and evaluate speaker identification systems, the corpus now serves as the primary source of data for work on automatic transcription of conversational telephone speech in English.

The first reported assessment of word recognition performance on the Switchboard-1 corpus was presented in [9] with an absolute word error rate of around 78%.[1] In this experiment only a small portion of the Switchboard-1 corpus was used in training. Over the years the performance of systems on this task has gradually improved. Progress is assessed in the yearly "Hub5E" evaluations conducted by the U.S. National Institute for Standards in Technology (NIST). The Cambridge University HTK group first entered these evaluations in 1997 using speech recognition technology based on the Hidden Markov Model Toolkit (HTK) [37] and has participated in evaluations on this task ever since. This paper describes the CU-HTK system for participation in the 2002 NIST Rich Transcription (RT-02) evaluation. We focus on two test conditions: the unlimited compute transcription task where the only design objective is the word error rate (WER); and the less than 10 times real-time (10 $\times$ RT) transcription task where the system processing time is not allowed to exceed 10 times the duration of the speech signal.

This paper is organized as follows: the first section briefly reviews basic aspects of the HTK Large Vocabulary Recognition (LVR) system, followed by a detailed description of the data used in experiments. In Section IV we present the acoustic modeling techniques essential to our system and discuss particular data modeling aspects. Section V outlines the pronunciation modeling, followed in Section VI by a description of the language models used in our systems. In Section VII we discuss issues in decoding and system combination. The structure of the full transcription system is presented in Section VIII, including a detailed analysis of the performance on large development and evaluation test sets. This system served as the basis for the 10 $\times$ RT system described in Section IX.

II. HTK LVR Systems

The HTK large vocabulary speech recognition systems are built using the Hidden Markov Model Toolkit [37] and are based on context dependent state clustered HMM sets with Gaussian mixture output distributions. The same basic model training methodology is used for a variety of tasks. The acoustic data is normally represented by a stream of 39 dimensional feature vectors with a frame spacing of 10 ms, based on 12 Mel-frequency perceptual linear prediction (MF-PLP) coefficients [33] and the zeroth cepstral coefficient c_0 representing the signal energy. The first and second order derivatives of each coefficient are appended to form the full feature vector. The words are mapped into phoneme strings using dictionaries based on a modified and regularly updated version of the LIMSI 1993 WSJ pronunciation dictionary [8]. The dictionaries contain multiple pronunciations per word. Cross-word context-dependent phone models using a context of either ± 1 in the case of triphones or ± 2 for the quinphones are used as the acoustic models. In addition to models for speech,

Manuscript received December 9, 2003; revised August 9, 2004. This work was supported by GCHQ and by DARPA under Grant MDA972-02-1-0013. This paper does not necessarily reflect the position or the policy of the U.S. Government and no official endorsement should be inferred. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Geoffrey Zweig.

The authors are with the Department of Computer Science, University of Sheffield, Sheffield S1 4DP, U.K. (e-mail: t.hai@dcs.shef.ac.uk).

Digital Object Identifier 10.1109/TSA.2005.852999

[1]The focus of the work was topic and speaker identification rather than word recognition.

the acoustic model set usually contains two silence models, one for silence, and one for short inter-word pauses with the latter preserving context across words.[2] In order to avoid under-training and the effect of unseen phone contexts, the HMM states are clustered using phonetic decision trees trained with a maximum likelihood (ML) criterion [37]. Initial single Gaussian per state models are created prior to state clustering by two-model re-estimation [35]. After state clustering and several iterations of Baum-Welch re-estimation, the number of mixture components is gradually increased, interleaved with multiple re-estimation steps.

The language models (LMs) are based on N-grams with backoff for smoothing. For conversational telephone speech large amounts of task-dependent material is not available and other schemes have to be adopted (see Section VI).

In order to allow for the use of complex acoustic and language models word graphs (lattices) are used extensively in re-scoring passes. The recognition system normally operates in multiple stages where an important aspect is acoustic model parameter adaptation, for example using maximum likelihood linear regression [21]. A more detailed description of the basic setup can be found in [35] or [37].

III. Training and Evaluation Data

The experiments in this paper made use of data from the *Switchboard-1* (Swbd1) corpus [10], the CallHome English corpus and small parts from the Switchboard-2 corpus. The Switchboard-1 corpus covers more than 2900 conversations from about 550 U.S. speakers. The speakers were unknown to each other and were requested to converse on a certain topic. The data used had 4-wire recordings with a sample rate of 8 kHz and μ-law encoded with a resolution of 8 b per sample. An initial manual segmentation and transcription of more than 250 h of speech was provided by the Linguistic Data Consortium (LDC). The inherent difficulty even in manual transcription of this type of data had forced many research sites to create their own segmentations and corrected versions of the transcriptions. In order to provide an improved baseline the data was more recently re-transcribed and re-segmented by Mississippi State University (MSU).[3] The *CallHome English* (CHE) corpus consists of a total of 200 conversations between family members and close friends, no restriction was placed on the conversation topics. The LDC distributed 120 conversations, comprising a total of 18.5 h of speech. The remainder was used in Hub5E evaluations in the years 1997–2001. A particular, though minor effect is the occasional existence of multiple speakers per conversation side on this data. The *Switchboard-2* (Swbd2) corpus was collected with the intention to serve as test-bed for speaker recognition, consequently most of it is not transcribed. The corpus was collected in a total of 5 phases, all calls were made within the U.S.A. The phases were recorded in different regions of the U.S.A, the fourth phase is also called Switchboard Cellular (Cell) collecting data over mobile phone channels with special focus on the GSM channel.[4] Each corpus has specific unique attributes and automatic speech recognition (ASR) system performance varies significantly with the corpus from which the data is drawn. In the following word error results, are also presented for each of the data sources in the test sets.

TABLE I
Data Sets Used for Training and Test. Training Sets are Described by the Source of Transcripts, Test Sets by the Originating Corpus

	Dataset	Description	#hours
Train	`h5train98`	LDC(Swbd1,CHE)	180
	`h5train00`	MSU(Swbd1), LDC(CHE)	265
	`h5train00sub`	Subset of `h5train00`	68
	`h5train02`	`h5train00` + LDC(Cell)	282
Test	`eval98`	Swbd1/CHE	3
	`dev01`	Swbd1/Swbd2/Cell	6
	`dev01sub`	Half of `dev01`	3
	`eval02`	Swbd1/Swbd2/Cell	6.5

Multiple training and test sets are used in the experiments in this paper. The selection of data for training of acoustic models allows a scaling of the complexity of experiments. Table I shows details of the training sets used in this paper. Note that most experiments are based on the `h5train02` set which covers data from Swbd1, CHE and Cell.

The Switchboard-1 part of `h5train00` and consequently `h5train00sub` and `h5train02` are based on a January 2000 snapshot of the MSU transcripts. The segment boundaries for these sets have been modified to reduce the amount of silence present in the training data. Based on forced alignments of the training data, a collar of only 0.2 s of silence on either side was retained and segments were split at long pauses. The table only shows data used for acoustic training. Details on the data used for language model training data can be found in Section VI.

Table I also shows the test-sets used in this paper. Note that `dev01` is the official 2001 Hub5 evaluation development set [28] consisting of 40 sides of Switchboard-2 (from the 1998 evaluation), 40 sides of Switchboard-1 (from the 2000 evaluation) and 38 sides of Switchboard-2 cellular data. The `dev01sub` set was selected to show similar word error rates to the full `dev01` set. For all cases a manual segmentation into speaker turns was available.

IV. Acoustic Modeling Experiments

In this section fundamental acoustic modeling techniques for conversational telephone data are presented. We discuss front-ends, the use of feature transformation schemes, data issues, the use of discriminative and speaker adaptive training schemes, and test-set speaker adaptation.

A. Acoustic Analysis

Due to the special transfer characteristics of telephone channels, the lower and upper frequency regions of the speech signal are attenuated and often very greatly so. In order to avoid the placement of filter-banks in regions containing only noise, the frequency analysis has been restricted to a range of 125–3800 Hz [13]. Initial experiments indicate WER improvements using these band limits after cepstral normalization.

1) Cepstral Mean and Variance Normalization: Cepstral mean normalization (CMN) can be used to reduce the effects

[2]The silence models differ only in the skip transition present in the model for short pause.

[3]See http://www.isip.msstate.edu/projects/switchboard/index.html.

[4]A more detailed description of the data can be obtained from the LDC website: http://www.ldc.upenn.edu.

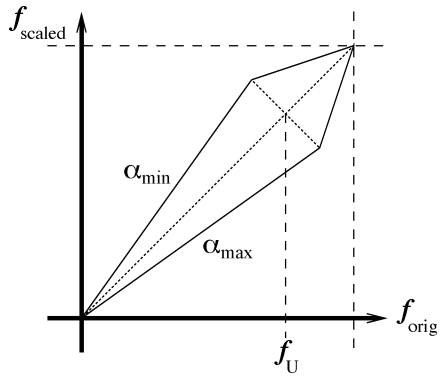

Fig. 1. Piecewise linear frequency scaling. Two parameters are needed to determine the frequency scaling are the scale factor α, and the cut-off frequency f_U.

of constant channel characteristics. The normalization can be performed on a per segment basis if the segments are of sufficient length. However, the audio data for this task has an average utterance length of 3.4 s (`h5train02`) which also includes a collar of 0.2 s of silence at the boundaries. Therefore segment based mean normalization is problematic. Since acoustic conditions can be assumed to be relatively stable over the duration of a conversation the mean can be calculated over a complete conversation side. This approach will be referred to as side-based mean normalization (Side-CMN). In preliminary experiments [12] significant reductions in WER by about 1% absolute with Side-CMN compared to segment-based CMN were observed.

In a similar manner to CMN, variance normalization can also be used. Again normalization on a per-side basis (Side-CVN) is advisable. Initial results indicate a 1–1.5% absolute improvement with both Side-CVN and Side-CMN over Side-CMN only. Another important advantage of side-based CVN with respect to its effect on vocal tract length normalization are discussed in the following section. For more detailed results the reader is referred to [12], [13].

2) Vocal Tract Length Normalization: Maximum likelihood vocal tract length normalization (VTLN) implements a per speaker frequency scaling of the speech spectrum [20]. The optimal scale factor α_{opt} is obtained by searching for the factor that yields the highest data likelihood. The optimal scale factor is then applied to yield a speaker specific feature stream. Normalization can be performed on the test data only or both on the training and test data. The advantage of VTLN lies in its simplicity and effectiveness. Since only a single parameter needs to be estimated, the method is robust and can be implement very efficiently.

The frequency scaling can be implemented by inverse scaling of the Mel filter-bank centre frequencies. In [12] we proposed a piecewise linear approach of the form presented in Fig. 1. This form ensures that frequencies tie up at the boundaries. The cut-off frequency is determined in advance. Warp factors are found by searching over a range of warp factors where the data likelihood is computed by performing an alignment of a previously obtained word level transcript. In our experience the quality of this transcript only has a minor effect on performance given enough speaker-specific data is available. Note that in this process the cepstral mean and variance normalization vectors have to be recomputed. Warping with a certain factor has an effect on the data likelihoods that would introduce a systematic bias in the warp factor search. This bias should be corrected using Jacobian compensation, but the application of CVN to the warped data achieves the same effect.

TABLE II
%WER FOR SYSTEMS TRAINED ON `h5train98` AND TESTED ON `dev01`. VTLN WARPING IN TEST ONLY OR TRAINING AND TEST

VTLN	`dev01`
—	42.1
test	41.1
train & test	38.0

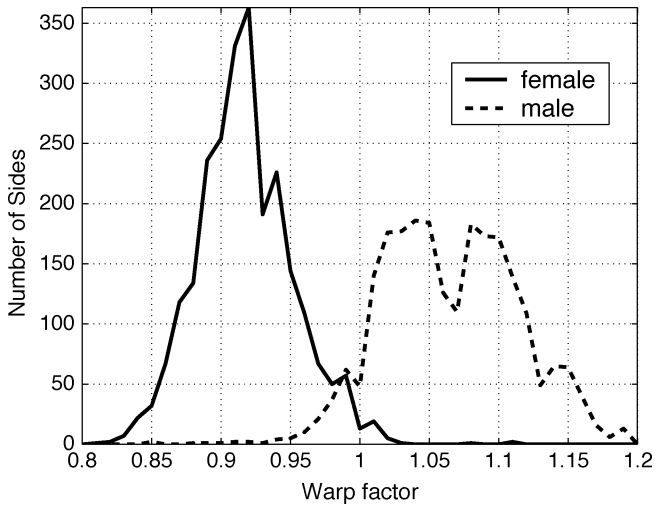

Fig. 2. Frequencies of warp factors per gender on `h5train02`. Triphone models for likelihood measurement were trained on `h5train98`.

Table II shows a performance comparison of VTLN in test only and in both training and test. Triphone acoustic models and a trigram language model were used in the experiments. The gain from test-only VTLN is less than half the gain obtained when VTLN is used in both training and test. Overall a relative reduction in WER of about 10% is usually observed over a range of test sets. In order to obtain a reasonable warp factor distribution on the training data multiple iterations of warp factor estimation and model retraining are necessary. Fig. 2 shows a typical distribution of warp factors across speakers on the `h5train02` training set. Note the clear split in the warp factor per gender with a broad distribution within-gender.

B. Acoustic Training Data

The training sets used for transcription of conversational telephone speech are relatively large compared to those available on other tasks. Such large training sets are required due to the considerable amount of variability in the data. In [14] we showed that, not unexpectedly, the incremental gain from an increase in the amount of training data slowly decreases even on a logarithmic scale. Starting from 20 h of data trebling the amount

TABLE III
%WER ON dev01sub USING 16COMP VTLN ML TRIPHONES. RESULTS ARE OBTAINED USING RE-SCORING OF 4-g LATTICES

Data	Swbd1	Swbd2	Cellular	Total
h5train00	25.2	42.1	42.5	36.5
h5train02	24.9	41.3	41.7	35.8

resulted in a WER improvement of more than 4%, a further trebling of the amount of data only gave an additional 1.6% gain.[5]

Another aspect is the appropriateness for the acoustic conditions. Table III shows experimental results on dev01sub using triphone models trained on different training sets. Note that the dev01sub test set also contains data from Swbd2 and Cell. The WER on the cellular data is similar to the Swbd2 performance. By adding the about 17 h of cellular data to the training set (h5train02) the word error rate can be reduced by 0.7% absolute. Note that the improvement is mostly on the Swbd2 and Cell portions of the data.

All experimental results in Table III were obtained using triphone models with 16 mixture components per speech state and approximately 6000 decision tree clustered states. Additional experiments with higher numbers of components gave an optimum result at 28 mixture components. This further decreased the overall error rate to 35.1%.

C. Heteroscedastic LDA

In this work a mixture of Gaussians with diagonal covariance is used to model the output distribution of each state. However, it is known that there are correlations in the feature vector which may limit the ability of the mixture model to accurately represent the underlying data. One solution to this is to linearly transform the features so that the data associated with each component is approximately uncorrelated. In addition, some dimensions contain little discriminatory information and should be removed.

Heteroscedastic linear discriminant analysis (HLDA) [18] is a linear projection scheme and may be viewed as a generalization of LDA. It removes the restriction that all the within class covariance matrices are the same. The HLDA projection matrix, $\mathbf{A}$, for a d-dimensional feature space, $\mathbf{o}_t$, may be written as

$$\hat{\mathbf{o}}_t = \mathbf{A}\mathbf{o}_t = \begin{bmatrix} \mathbf{A}_{[p]}\mathbf{o}_t \\ \mathbf{A}_{[d-p]}\mathbf{o}_t \end{bmatrix} \tag{1}$$

where the top p dimensions are deemed to be those dimensions that contain discriminatory information, the useful dimensions. The final $(d-p)$-dimensions, contain no useful information and are called the nuisance dimensions. Those are modeled using a global distribution and hence can be ignored for the purpose of further training and evaluation.

The maximum likelihood estimate of the transform parameters can be obtained in an iterative process [7]. In this work the projection matrix is initialized to an identity matrix. The useful dimensions where selected based on Fisher ratios.

[5]It is important to note that these results were obtained by testing on Swbd2 and CHE data. The smallest training set however only contained data from Swbd1, while the larger sets included CHE data.

TABLE IV
%WER ON dev01sub USING 28COMP h5train02 TRIPHONES, WITH MF-PLP (STD), SEMI-TIED (ST) COVARIANCE MATRIX AND HLDA FRONT-ENDS. RESULTS WERE OBTAINED BY RE-SCORING OF 4-g LATTICES

	Feature Transform	Swbd1	Swbd2	Cellular	Total
ML	—	24.2	40.8	40.4	35.1
	ST	23.0	39.2	39.3	33.7
	HLDA	22.2	38.8	39.1	33.3
MPE	—	21.0	37.0	36.6	31.4
	HLDA	19.3	35.6	35.8	30.1

The HLDA transforms built for this work projected a 52-dimensional feature down to 39 dimensions.

The 52-dimensional source vector consisted of the standard 39-dimensional feature vector with third order derivatives appended. HMMs with 16-component mixtures based on the standard feature vectors were trained and then extended to incorporate the third derivatives. After further steps of Baum-Welch re-estimation the transform was estimated and the number of model components increased by mixing-up [37] to 28 mixture components per state. For semi-tied covariance systems [7], the process is identical except no addition of third derivatives was used.

Table IV compares systems using the 39-dimensional front-end with the use of both a global semi-tied covariance system, and HLDA system and with ML training. Using a global semi-tied covariance matrix reduced the error rate by 1.4% absolute. An additional 0.4% absolute was obtained by using HLDA with third order differential coefficients rather than a semi-tied system. Hence there is additional discriminatory information that can be extracted from the third derivatives. Discriminative training (see Section IV-D) was then applied to the ML-trained system. Though the reduction in WER due to HLDA with discriminatively trained models is 1.3% absolute rather than the 1.8% obtained with the ML-trained system, there was still a significant advantage in using HLDA [25].

D. Discriminative Training

The standard criterion for estimation of HMM parameters is maximum likelihood. The maximum likelihood estimate is optimal in the sense that it is consistent with minimum variance. Two important assumptions are made: a large number of training samples is available; and the model itself is correct, i.e., reflects the true nature of the data. Neither of these assumptions is true for HMM based speech recognizers [26]. Consequently discriminative training schemes are of interest.

In conjunction with HMM based speech recognizers several discriminative training schemes have been proposed. Most importantly we have shown that Maximum Mutual Information (MMI) yields better performance than ML for the transcription of conversational telephone speech. Use of discriminative criteria in training is well known in small vocabulary tasks [29]. The next section gives a brief description of MMI training, followed by a more detailed description of an alternative discriminative training scheme, minimum phone error (MPE) training.

1) Maximum Mutual Information: For R training observation sequences $\{\mathcal{O}_1, \ldots, \mathcal{O}_r, \ldots \mathcal{O}_R\}$ with corresponding transcriptions $\{s_r\}$, the MMI objective function for HMM param-

TABLE V
%WER ON eval98 AND A SUBSET OF THE TRAINING SET (train) USING TRIPHONE MODELS. train RESULTS WERE OBTAINED USING A LATTICE UNIGRAM LM, TEST-SET RESULTS BY RE-SCORING OF 4-g LATTICES

Criterion	Training set	τ^I	train	eval98
ML	h5train00sub	—	47.8	46.5
MMI	h5train00sub	50	32.2	43.8
ML	h5train00	—	47.2	45.6
MMI	h5train00	200	35.8	41.4
MPE	h5train00	100	34.4	40.8

eter set λ, including the effect of scaling the acoustic and LM probabilities can be written

$$\mathcal{F}_{\mathrm{MMI}}(\lambda) = \sum_{r=1}^{R} \log \frac{p_\lambda(\mathcal{O}_r|\mathcal{M}_{s_r})^\kappa P(s_r)}{\sum_s p_\lambda(\mathcal{O}_r|\mathcal{M}_s)^\kappa P(s)} \tag{2}$$

where $\mathcal{M}_s$ is the composite model corresponding to the word sequence s, $P(s)$ is the probability of this sequence as determined by the language model and κ is a scale factor.[6] The summation in the denominator of (2) is taken over all possible word sequences allowed in the task. Hence MMI maximizes the posterior probability of the correct sentences. The denominator in (2) can be approximated by a word lattice of alternative sentence hypotheses.

The Extended Baum-Welch (EBW) algorithm is used for parameter optimization [11]. The parameter update formulae require the collection of numerator (num) and denominator (den) statistics derived from word lattices based on recognition of the training set. In order to allow a broader range of confusable hypotheses the use of a weak language model has been shown to be important [34]. The parameter update formulae for the means and variances of Gaussian m in state j is given by

$$\hat{\mu}_{jm} = \frac{\left\{\theta^{\mathrm{num}}_{jm}(\mathcal{O}) - \theta^{\mathrm{den}}_{jm}(\mathcal{O})\right\} + D_{jm}\mu_{jm}}{\left\{\gamma^{\mathrm{num}}_{jm} - \gamma^{\mathrm{den}}_{jm}\right\} + D_{jm}} \tag{3}$$

$$\hat{\sigma}^2_{jm} = \frac{\left\{\theta^{\mathrm{num}}_{jm}(\mathcal{O}^2) - \theta^{\mathrm{den}}_{jm}(\mathcal{O}^2)\right\} + D_{jm}\left(\sigma^2_{jm} + \mu^2_{jm}\right)}{\left\{\gamma^{\mathrm{num}}_{jm} - \gamma^{\mathrm{den}}_{jm}\right\} + D_{jm}} - \hat{\mu}^2_{jm} \tag{4}$$

where $\theta^{\mathrm{num}}(\mathcal{O}^n)$ and $\theta^{\mathrm{den}}(\mathcal{O}^n)$ denote n-th order Gaussian occupancy weighted sums of the data based on the numerator and denominator lattices respectively and γ is the Gaussian occupancy summed over time. The constant D can be used to control convergence speed and robustness and is set on a per Gaussian level.

It was shown that data weighted per Gaussian interpolation between ML and MMI parameter estimates substantially increases the robustness of the training process. I-smoothing [31] is a way of applying an interpolation between a ML and a discriminative objective function in a way which depends on the amount of data available for each Gaussian. In the case of MMI this means that the numerator occupancies are increased by a certain amount τ^I, while leaving the average first and second order data values unchanged. I-smoothing is used in training of all discriminative models in this paper.

Table V shows a comparison of ML versus MMI, on different training set sizes. Note that the ML system serves as a starting point for further parameter estimation steps using the MMI criterion. In conjunction with I-smoothing substantial improvements in WER are obtained. I-smoothing trades performance on the training set against improved generalization to the test set. Furthermore the relative gain from using MMI training on a 68 h training set is about 6%, however on 265 h of data the relative WER improvement is 10%.

2) Minimum Phone Error Training: The aim in discriminative training is to choose the model parameters such as to minimize the word error rate. The minimum word error rate (MWE) criterion [31] is designed to maximize the expected word accuracy on the training set. It was found that an equivalent formulation at the phone level yields better generalization. The minimum phone error criterion is defined as

$$\mathcal{F}_{\mathrm{MPE}}(\lambda) = \sum_{r=1}^{R} \frac{\sum_s p_\lambda(\mathcal{O}_r|\mathcal{M}_s)^\kappa P(s)\mathrm{RawAccuracy}(s)}{\sum_s p_\lambda(\mathcal{O}_r|\mathcal{M}_s)^\kappa P(s)} \tag{5}$$

where $\mathrm{RawAccuracy}(s)$ is a measure of the number of phones accurately transcribed in hypothesis s. The objective function takes into account that many phone sequences are at least partially correct and consequently cannot fully count as a competitor. This is expressed by adding the correct fraction to the numerator. Details on how the computation of the $\mathrm{RawAccuracy}(s)$ and the re-estimation procedure can be found in [31].

Table V also shows WER results on eval98 for MPE training using the h5train00 training set. MPE outperforms MMI with I-smoothing by 0.6% absolute on this test set. Improvements of a similar kind have been verified on other test sets. It is important to note the necessity for I-smoothing when using the MPE criterion. Without I-smoothing the absolute WER is 50.7% which is even poorer than the baseline ML result. More detailed results and descriptions of MPE/MWE are given in [31] and [36].

E. Unsupervised Test-Set Adaptation

Adaptation to the test speaker and the acoustic environment greatly improves the performance of automatic speech recognizers. The VTLN and side based normalization schemes discussed above are essentially adaptation techniques and show substantial performance improvements. Techniques discussed in this section use errorful transcripts of the data generated in previous decoding stages for adaptation supervision.

1) Maximum Likelihood Linear Regression: MLLR [21] is a well known adaptation technique that is widely used in many speech recognition tasks. The model parameters are adjusted by a linear transform (including a bias) such that the likelihood on the adaptation data is increased. On this task, assuming a single speaker on a conversation side, side-based adaptation is performed. Typically mean vectors are adapted using block-diagonal transforms. However, the use of HLDA (see Section IV-C) removes the association of blocks to the static cepstra and the derivatives. Consequently a full transform for the means was used in these experiments. Variance adaptation is performed using diagonal transforms [5].

The sets of Gaussians can be grouped into multiple classes, each associated with a particular transform. This is especially helpful if more data is available. The classes can be found using

[6]It is assumed that the LM probabilities $P(s)$ have already been "scaled" (raised to the power) by the normal LM scale factor $1/\kappa$ and hence further scaling by κ takes them back to their original values.

TABLE VI
%WER ON dev01sub USING 28COMP TRIPHONES TRAINED ON h5train02. ALL RESULTS ARE OBTAINED BY RESCORING OF 4-g LATTICES

	Adaptation	Swbd1	Swbd2	Cellular	Total
ML	-	22.2	38.8	39.1	33.3
	global MLLR	20.2	35.8	36.4	30.7
MPE	-	19.3	35.6	35.8	30.1
	global MLLR	18.0	33.6	34.3	28.5

regression class trees [22] or by manual definition. In practise most of the WER gain is obtained by using one transform for speech and one for the silence models, further termed global MLLR.

Table VI shows results using a single iteration of global MLLR using ML and MPE trained model sets. Note that the models also use side-based mean, variance and VTL-normalized data. The relative improvement using ML trained models is about 8%, compared to a 5% relative reduction in WER using MPE trained models. The relative improvements are similar for all sub-sets of the data.

2) Full Variance Transforms: In the above experiments diagonal transforms for variance adaptation were used. Even with multiple transform classes this guarantees that the model covariances remain diagonal after adaptation. Improved performance can be obtained by using a full matrix for speaker based variance adaptation. In this case the transformed covariance matrix, $\hat{\mathbf{\Sigma}}^{(m)}$, for mixture component m is given by

$$\hat{\mathbf{\Sigma}}^{(m)} = \mathbf{H}\mathbf{\Sigma}^{(m)}\mathbf{H}^{\top} \tag{6}$$

where $\mathbf{\Sigma}^{(m)}$ is the speaker-independent covariance matrix and $\mathbf{H}$ is a linear transform estimated on the adaptation data. This can be interpreted as speaker-dependent global semi-tied covariance matrix [7]. A summary of this adaptation scheme is given in [6]. There are a number of options. Theoretically the transforms may be full, diagonal, or block-diagonal. In practice a full transform was used in all cases in this paper. The full variance (FV) transform was computed after standard mean and variance MLLR. Typically a WER reduction of 0.4% to 0.8% was obtained. However as a side effect, we found that there were reduced benefits from multiple MLLR regression classes when used with a full variance transform.

3) Lattice Based MLLR: In unsupervised adaptation the word level output of a previous recognition pass is used as the word level supervision. Errors in this transcription will affect the effectiveness of the adaptation. To compensate for the uncertainty in the transcription a set of alternative word hypotheses can be used as the supervision. The lattice MLLR technique presented in [32] employs word lattices to represent these alternatives. During the transform estimation the contribution of the alternatives is weighted by posterior probability estimates based on a lattice forward-backward pass.

Table VII shows a break-down of results for the adaptation techniques on dev01sub. When using HLDA, global MLLR adaptation brings an improvement of 1.6% WER absolute over the baseline. Iterative lattice MLLR using two speech transforms brings a further 0.9%. In this case the result can be improved only slightly when using a FV transform. Overall the improvement in WER from adaptation in this case is about 9% relative. Table VII also shows results of adaptation with or without the use of an HLDA transform (Section IV-C). Note that the

TABLE VII
%WER ON dev01sub USING 28-COMPONENT HLDA MPE TRIPHONES TRAINED ON h5train02. ALL RESULTS ARE OBTAINED BY RESCORING OF 4-g LATTICES

HLDA	Adapt	Swbd1	Swbd2	Cellular	Total
×	-	21.0	37.0	36.6	31.4
×	LatMLLR+FV	18.6	33.7	34.6	28.9
√	-	19.3	35.6	35.8	30.1
√	MLLR	18.0	33.6	34.3	28.5
√	LatMLLR	17.5	32.7	32.9	27.6
√	LatMLLR+FV	17.6	32.4	32.7	27.5

TABLE VIII
%WER ON dev01sub USING 28COMP HLDA TRIPHONES TRAINED ON h5train02. ALL RESULTS ARE OBTAINED BY RESCORING OF 4-g LATTICES USING CONSTRAINED MLLR ADAPTATION ONLY

	SAT	Swbd1	Swbd2	Cellular	Total
ML		20.5	36.0	36.4	30.9
	√	20.5	35.1	35.9	30.4
MPE		17.9	33.4	34.0	28.3
	√	18.1	33.3	33.6	28.2

difference between systems without or with HLDA transforms for unadapted models is 1.3% WER absolute, compared to 1.4% with adapted models. This indicates that the improvements from these techniques are approximately additive.

F. Speaker Adaptive Training

Adaptive training is a powerful training technique for building speech recognition systems on nonhomogeneous data. Variability in the training data may result from the speaker changing, differing acoustic environments or varying channel conditions. The basic idea of adaptive training is to use one or more transformations of features or model parameters to represent these speaker and environment differences. A canonical model can then be trained, given the set of speaker/environment transforms. This canonical model should be more compact and amenable to being transformed to a new speaker, or acoustic condition, than standard speaker independent (SI) systems.

Forms of adaptive training have already been incorporated into the training process in the form of VTLN and mean and variance normalization (see Section IV-A). These methods use constrained transformations of the feature space for normalization. However, gains are increased further by incorporating linear transformations of model parameters, for example MLLR [21], into the training process. This was the original form of speaker adaptive training (SAT) described in [1]. One of the issues with the original SAT scheme is the cost of training. This may be solved by using constrained MLLR transforms [6]. Then SAT can be implemented by transforming the features, and there is no need to change the model parameter optimization process. As constrained MLLR is a feature space transform it is simple to incorporate it into the discriminative training framework.

Table VIII shows the performance of SAT systems, trained using constrained MLLR transforms. In testing, the systems are adapted by using constrained MLLR. An improvement of 0.5% absolute over the baseline can be observed, for the ML trained models, where the gain originates from the more difficult data Switchboard-2 and Switchboard Cellular. With MPE training the gain is reduced. Note that these results, for the purpose of consistency, only involve test set adaptation using constrained

TABLE IX
%WER ON dev01sub USING 28COMP TRIPHONE MODELS TRAINED ON h5train02, HLDA AND OPTIONALLY PRONUNCIATION PROBABILITIES. RESULTS ARE OBTAINED BY RESCORING OF 4-g LATTICES

	Training	PrProb	Swbd1	Swbd2	Cell	Total
MPron	ML	×	22.2	38.8	39.1	33.3
		√	21.5	37.9	38.1	32.4
	MPE	×	19.3	35.6	35.8	30.1
		√	19.1	35.0	35.6	29.8
SPron	ML	×	21.6	37.9	37.8	32.3
		√	21.3	37.7	37.4	32.0
	MPE	×	19.4	35.2	35.1	29.8
		√	19.6	34.9	34.9	29.7

MLLR. Using the MPE HLDA SAT models as in Table VIII in conjunction with LatMLLR+FV adaptation gives a WER of 27.3% absolute. The use of SAT is important to yield complimentary system output for the purpose of system combination (see Section VII-B).

V. PRONUNCIATION MODELLING

CU-HTK systems use a pronunciation dictionary for translation of words into phoneme sequences, where each word in the dictionary has one more possible pronunciations associated with it. The dictionaries used in training and test are obtained from the CU base dictionary. The core of this base is the 1993 LIMSI WSJ lexicon [8], with manually generated additions and corrections. Pronunciations for new words if needed are added manually. On average 1.1 to 1.2 pronunciations per word are included.

A. Pronunciation Probabilities

Unigram pronunciation probabilities, that is the probability of a certain pronunciation variant for a particular word, can be estimated based on an Viterbi-alignment of the training data. Counting the number of occurrence of pronunciation variants gives rise to an estimate for the probabilities. Considerable smoothing is necessary to account for the inevitable data sparsity.

The dictionaries in the HTK system explicitly contain silence models as part of a pronunciation. This allows the inclusion of the silence models when estimating probabilities. The most successful scheme in our experiments uses three separate dictionary entries for each real pronunciation which differed by the word-end silence type: no silence; a short pause preserving cross-word context; and a general silence model altering context. Smoothing of the probability estimates used the overall distribution for each silence variant. Finally all dictionary probabilities are renormalized so that for a given word the probability of the most likely pronunciation is set to one to avoid an additional penalty for words with many variants. During recognition the pronunciation probabilities are scaled by the same factor as used for the language model. Table IX shows that the use of pronunciation probabilities gives a reduction in WER of 0.9% absolute on dev01sub for ML. Even larger gains were observed on some other test-sets [15].

B. Single Pronunciation (SPron) Dictionaries

The standard approach to pronunciation modeling is to use multiple pronunciations (MPron) for each word. However, the considerable pronunciation variation in conversational data makes the use and selection of multiple pronunciations difficult and causes additional confusability in decoding. Theoretically Gaussian mixture based HMMs should be able to cope with phone or sub-phone substitutional effects. These phone substitutions are the main cause of the existence of multiple pronunciations in dictionaries. In this case the training of model parameters can implicitly perform a similar task to manual phonemic labeling.

An automated scheme for deriving a single pronunciation from the multiple pronunciation dictionary was developed. This is described in detail in [16]. The algorithm obtains pronunciation information from the acoustic training data to train simple statistical models that allow the selection of pronunciation variants. Since the list of words used in training usually differs from that used in recognition, the algorithm also provides a method for the selection of pronunciations for words not observed in training. An MPron dictionary and an HMM set trained using that dictionary are used to obtain pronunciation frequencies from the training data.

Table IX shows a comparison of the SPron system with the standard MPron system, and in addition the use of pronunciation probabilities. Note that for the SPron system the "pronunciation probability" is simply the probability of the word being followed by an optionally deletable "short" silence model, or a standard silence model. In the ML training case the SPron system outperforms the baseline MPron system by 1% absolute, in conjunction with pronunciation probabilities this is reduced to 0.4% absolute. In the case of MPE training both pronunciation probabilities and SPron system give reduced gains. However, the output of SPron and MPron systems still differ significantly and consequently can be used for system combination (see Section VII-B).

VI. LANGUAGE MODELLING AND WORD LISTS

In most speech transcription tasks such as for example dictation or Broadcast News transcription large amounts of representative text data are available. In the case of transcription of spontaneous speech over the telephone a very large corpus is not available as the cost of transcribing the data is considerable. Consequently the amount of in-domain data for the training of language models and vocabulary selection is fairly small and restricted to the transcription of the acoustic training data.

CU-HTK systems have followed a certain strategy for building language models and selecting word lists on this task for several years [13]. All words from the acoustic training set are used in decoding.[7] In order to minimize the Out-of-Vocabulary (OOV) rate on the test sets, this set of words is merged with the 50 000 most frequent words occurring in 204 million words (MW) of Broadcast News (BN) training data, yielding a vocabulary size of around 55 000. Given the vocabulary, word bigram, trigram, and 4-g language models are trained on the acoustic LM training set. These models are then interpolated with corresponding models trained on the BN corpus. The resulting 4-g LM is further interpolated with a class-based

[7]Note that for the purpose of LM training some text processing steps are necessary to deal with for example partial words.

TABLE X
PERPLEXITIES ON VARIOUS TEST SETS USING INTERPOLATED LANGUAGE MODELS (tg = trigram, fg = 4 − g)

Model	N-gram	eval98	dev01cell	eval02
AcAllLM	fg	90.9	78.4	77.4
CellLM	fg	112.9	91.5	104.1
BNLM	fg	111.6	106.5	121.0
fgint02	fg	72.4	63.8	64.1
ClassLM	tg	112.9	99.7	102.6
fgintcat02	fg	71.5	62.8	63.3

trigram language model where the classes are automatically generated based on word bigram statistics [17], [24], [27].

The vocabulary selected for the CU-HTK 2002 Hub5E system is derived from h5train02 in the aforementioned manner, giving a word list of 55 449 distinct entries. This yielded an OOV rate of 0.38% on eval98, 0.17% on the cellular part of dev01 and 0.54% on eval02. Three language models were trained on different sets of data: AcAllLM is trained on h5train00 and h5train98 to encompass both MSU and LDC transcript styles. This version of the transcripts includes the false starts and covers a total of 6.2 MW. As the amount of training was relatively small the model was trained using modified Kneser-Ney discounting [2]. CellLM was trained on the Switchboard Cellular part of the h5train02 set with a size 0.2 MW. Again modified Kneser-Ney discounting was used. The BNLM model was trained on 204 MW of Broadcast News data ranging in epoch from January 1992 to December 1997 to cover approximately the dates of data collection. Smoothing used Katz-backoff and Good-Turing discounting. The individual language models were merged to form a single language model that effectively interpolates the component models with interpolation weights 0.43:0.25:0.32 for the three language models (AcAllLM:CellLM:BNLM) where interpolation weights were chosen by perplexity minimization. The merged language model (fgint02) contained 4.77 million bigrams, 6.33 million trigrams and 7.35 million 4-g.

The class trigram language model used 350 automatically generated word classes. Classes and trigram models were trained on the h5train02 transcriptions only. The final class model contained 75 k bigrams and 337 k trigrams. In the final interpolation stage the optimal weight was 0.81 for the word 4-g and 0.19 for the class model. Table X shows the perplexities on several test sets. Note that before merging the individual perplexities are high compared to the merged model (fgint02). Despite significantly higher perplexities the category LM yields a further reduction of about one point in perplexity on all test sets.

VII. DECODING AND SYSTEM COMBINATION

In Section II we briefly mentioned the generic decoding strategy for HTK LVR systems. Initial Viterbi decoding passes are used to produce lattices that allow the search space to be constrained in subsequent passes. This is useful for using more complex acoustic and language models, for example using quinphone models or 4-g LMs.

A. Minimum Word Error Rate Decoding

The standard criterion in ASR uses the Maximum A Posteriori (MAP) principle. For continuous speech recognition this implies the search for the sentence yielding the highest posterior probability. This is notably different from the desired objective of word error rate (rather than sentence error rate) minimization. The use of confusion networks [23] allows an efficient implementation of the minimum word error rate principle. For a particular word lattice link posterior probabilities are estimated using the forward-backward algorithm. The lattice is transformed into a linear graph, or confusion network (CN) employing a link clustering procedure [23]. Both temporal information as well as information on phonetic similarity of words is used in clustering. The linear graph consists of a sequence of so called confusion sets, which contain competing single word hypotheses with associated posterior probabilities. By picking the word with the highest posterior from each set the sentence hypothesis with the lowest overall expected word error rate can be found. The use of CN decoding normally reduces the WER by 0.5%–1% absolute. More detailed results on the CU-HTK 2002 Hub5E evaluation system can be found in Section VIII-B.

The estimates of the word posterior probabilities encoded in the confusion networks can be used directly as word confidence scores. As the posteriors tend to be over-estimates of the true posteriors they are mapped to confidence scores using a piecewise linear function based on a decision tree [3].

B. System Combination

In recent years interest in the development of complementary systems, i.e. systems that substantially differ in their word level output while retaining a similar word error rate, was stimulated by techniques such as ROVER [4]. ROVER allows the combination of the system outputs either by voting or by the use of confidence scores. This approach can be generalized to the use of confusion networks in the combination [3]. In this case CN output from each system is generated and dynamic programming is used to align the confusion networks. The cost function is an estimate for the probability of a word match given two confusion sets. After alignment the networks are merged and standard CN-decoding is applied. Confusion network combination (CNC) allows the weighting of systems, however normally with limited effect. In this paper no weighting is used in system combination. Results of CNC are discussed in Section VIII-B.

VIII. CU-HTK APRIL 2002 Hub5E SYSTEM

In the previous sections we have presented a set of techniques that are important for the transcription of conversational telephone speech and have discussed the performance of each technique. However, in practise the performance improvements are rarely additive and the selection of techniques is nontrivial. Accordingly the use of new techniques cannot be assessed purely using baseline comparisons, their operation in a complete speech recognition system is of at least equal importance. Consequently research of the development of large ASR systems is not only interesting for the purpose of finding the best performance, but also for an improved understanding of the relationship of techniques.

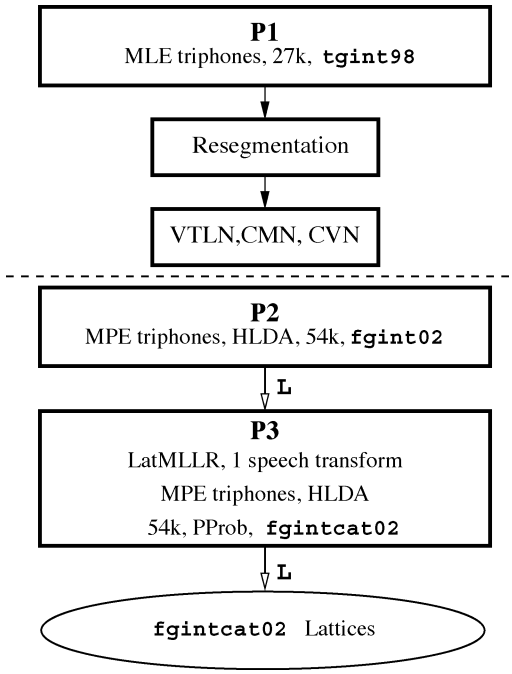

Fig. 3. First three passes (P1)–(P3) in the CU-HTK 2002 Hub5E system. Normal arrows denote word level output, arrows with "L" denote lattices. The output of these stages is a set of word lattices.

The following describes the system designed for participation in the RT-02 evaluations for transcription of conversational telephone speech.[8] We discuss the broad structure and the processing and performance of the system.

A. System Structure

The system operates in a total of six passes. In general each pass employs techniques that have not been used in previous passes. A pass can sometimes be split into a certain number of sub-processes. The output of a pass is either a word string, a word lattice or a confusion network. Input to the system is a set of audio files and time information of segments to be recognized. The system presented here is based on manual segmentation of the audio stream. Each stage of the system operates on a conversation side basis.[9] Thus the following discussion will describe the transcription of a single conversation side only.

The first part of the system is shown in Fig. 3 and is designed for the generation of lattices to be used in the second part. In the first step the aim is to perform a robust estimation of mean and variance normalization vectors and VTLN warp factors. The data is encoded in MF-PLP coefficients as described in Section IV-A. A set of transcripts is generated with ML triphone models and a trigram language model (for details see [13]). As this part of the system originates from the 1998 CU-HTK system, the word error rate of this pass is fairly high. However, no word level information is passed on to later stages. The initial set of word level transcripts also allows a re-segmentation of the data.[10] CMN and CVN vectors are recomputed and the new segments and the word level transcripts are used in VTLN estimation as described in Section IV-A2.

In the second pass (P2) transcripts for the purpose of HMM parameter adaptation are produced. For this purpose state clustered triphone models with 6155 speech states are used. Initially the models were trained on h5train02 using the ML criterion using the standard HTK mix-up procedure [37]. After estimation of an HLDA matrix, the number of mixture components was increased to the total number of 28 Gaussians per speech state. Further re-estimation steps using the MPE criterion with I-smoothing gave the model set used in this pass. Decoding used the 54 k MPron dictionary described in Section VI. The Viterbi decoding pass used a trigram language model (tgint02) produced in the fashion described in Section VI. Further rescoring using the fgint02 LM gave the output of this pass.

In the third pass (P3) a set of lattices for use in all subsequent rescoring passes is generated. Global MLLR transforms are estimated using the output from P2, followed by lattice-based MLLR. The acoustic models used are identical to those in P2 and the interpolated and class smoothed 4-g LM fgintcat02 was used to obtain a set of lattices. In addition pronunciation probabilities were used in decoding. Fig. 3 summarizes the essential components of the first three stages. The output of P3 forms the basis for all subsequent processing steps.

The second part of the system is based on the principle of system combination and contains stages based on complementary systems that differ in model training and adaptation strategies. This part is split into branches where each branch corresponds to a specific model construction strategy. In total three branches were used: one associated with MPE SAT-trained models, together with HLDA feature projection and a standard MPron dictionary (branch 1); one associated with a non-HLDA MPE trained model (branch 2); and one branch using models based on an HLDA transform, MPE training and an SPron dictionary (branch 3). All branches are further subdivided into two passes: The (P4.[123]) passes are based on triphone models, the (P5.[123]) passes use quinphone models. Fig. 4 shows an outline of the essential operation blocks and the data flow.

For adaptation in the triphone stage all branches use the same strategy: the fgintcat02 lattices obtained in the first part of the system and the associated first best output is used in a lattice-based MLLR+FV scheme as described in Section IV-E3. A total of four transforms for the speech classes are estimated. The decoding step of these branches consists of an acoustic rescoring step using the fgintcat02 lattices generated in (P3) with the adapted models. Confusion networks were generated and word strings obtained using CN decoding. The acoustic models in the first two branches have the same characteristics in terms of the number of states and mixture components as those used in (P2)/(P3). The models in the third branch differ in so far as the number of system parameters is slightly smaller with a total of 5955 states with 28 mixture components each.

The quinphone models use a cross-word context of ± 2 phones with word boundary information and are trained in a

[8]The interested reader is referred to http://www.nist.gov/speech/tests/rt/rt2002.

[9]This means that no information from the second channel or other conversations is used in the transcription process.

[10]Only 0.2 s of silence are left at segment boundaries and segments are split if more than a second of silence occurs. This ensures a similar silence/speech ratio found on the training data which is important for side-based normalization.

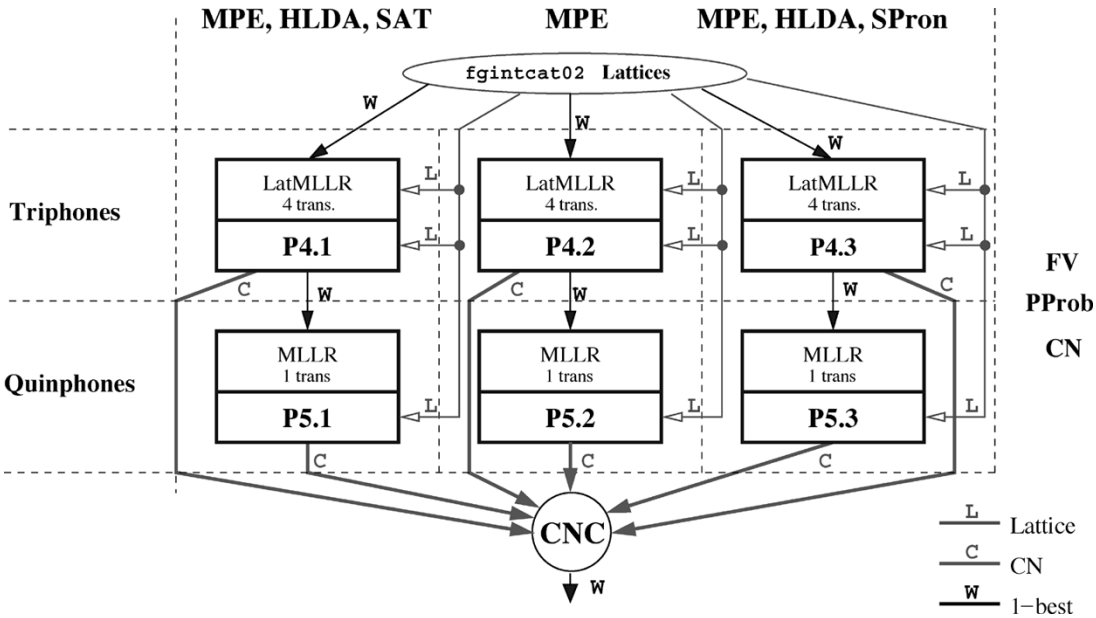

Fig. 4. Stages P4.[123], P5.[123] and the system combination stage P6 of the CU-HTK 2002 Hub5E system.

TABLE XI
%WER ON dev01 USING h5train02 TRAINING DATA FOR ALL STAGES OF THE CU-HTK 2002 Hub5E SYSTEM

	Passes	Comment	CN	dev01				eval02			
				Swbd1	Swbd2	Cell	Total	Swbd1	Swbd2	Cell	Total
Part 1	P1	trans for VTLN	×	31.7	46.9	48.1	42.1	35.6	44.6	50.5	44.0
	P2	trans for MLLR	×	20.1	34.7	34.3	29.6	24.6	30.9	34.8	30.4
	P3	bigram	×	22.0	36.2	35.1	31.0	26.3	31.5	35.4	31.4
		trigram	×	19.2	33.2	31.9	28.0	23.4	28.5	32.1	28.3
		4-gram	×	18.6	32.1	31.4	27.3	22.9	28.2	31.4	27.8
		4-gram + class-LM	×	18.5	32.2	31.1	27.2	22.5	28.0	31.3	27.5
Part 2	P4.1	SAT tri	×	18.0	31,4	30.2	26.5	22.1	26.7	30.6	26.8
		SAT tri	√	17.5	30.7	29.6	25.9	21.6	26.3	29.6	26.1
	P4.2	non-HLDA tri	×	19.8	32.7	32.4	28.2	23.7	28.5	32.8	28.6
		non-HLDA tri	√	18.8	31.4	31.0	27.0	22.3	27.4	31.2	27.2
	P4.3	SPron tri	×	18.4	31.5	30.0	26.5	22.1	27.3	29.7	26.6
		SPron tri	√	18.0	31.0	29.7	26.2	21.5	26.6	29.1	26.0
Contrast		tri	×	18.2	32.0	30.7	26.8	22.0	27.0	30.6	26.8
		tri	√	17.6	31.2	29.9	26.2	21.5	26.5	29.6	26.2
	P5.1	SAT quin	×	17.9	31.6	30.1	26.4	22.5	26.3	29.9	26.5
		SAT quin	√	17.2	30.8	29.2	25.7	21.5	25.5	28.6	25.4
	P5.2	non-HLDA quin	×	19.5	32.7	32.0	28.0	23.7	27.7	32.3	28.2
		non-HLDA quin	√	18.5	31.8	30.6	26.9	22.4	26.7	30.7	26.9
	P5.3	SPron quin	×	18.6	31.8	29.7	26.6	22.2	27.0	29.8	26.6
		SPron quin	√	18.1	31.1	28.8	25.9	21.5	26.4	28.8	25.8
Contrast		quin	×	18.1	32.0	30.3	26.7	22.3	26.6	30.0	26.6
		quin	√	17.5	31.2	29.2	25.9	21.4	25.8	28.8	25.5
CNC	P6	P4.[123] + P5.[123]	√	16.4	29.2	27.4	24.2	19.8	24.3	27.0	23.9

similar fashion to the triphones. The average number of states associated with speech models is higher, with 9640 states for the models used in the first two branches and 9418 states in case of the SPron based models. Models are first trained up to 16 mixture components. After estimation of HLDA matrices (branches 1 and 3 models only) the number of mixture components was increased to 24. In the (P5.[123]) stages adaptation is performed using global MLLR, together with a full-variance transform. As the use of full cross-word quinphones substantially increases the size of static phone networks the quinphone decoding stages use the dynamic network decoder [30] for the rescoring of lattices.

The output of each of the stages (P4.[123]) and (P5.[123]) is a set of confusion networks. These are merged in the final confusion network combination stage (P6). In this stage minimum word error rate decoding is used to arrive at the final word level transcription for the conversation side. The overall structure of this second system part is represented in Fig. 4. Note that the arrows denote the flow of information, either in the form of word strings (W), lattices (L), or confusion networks (C). The final output is a word string with word level times and confidence scores.

B. Recognition Performance

Table XI shows WERs on the full dev01 test set and the full 2002 evaluation set for all system passes. Since performance of the individual stages is very similar for both test sets the following discussion will concentrated on results on dev01. As

dev01 has served in development this shows that the results generalize to independent test sets.

The WER of the first pass (P1) is rather high. As no word level information is transferred beyond this stage this is of little concern. Re-segmentation removed a total of 2628 s of audio or about 12% of the audio data to obtain consistent amounts of silence at segment boundaries. The second pass (P2) shows the unadapted performance of triphone models using VTLN, HLDA, MPE training and a 4-g language model. Note that the word error rate on Switchboard-1 data is about 20%, substantially lower than the data originating from Switchboard-2 and Cellular sources. The (P2) output lattices are used to estimate a global MLLR transform in (P3). Initial lattices are produced using the bigram language model. In subsequent lattice expansion and pruning steps more complex language models are applied. A more than 3% absolute performance difference can be observed between bigram and trigram language models. The use of 4-g yields another 0.7% absolute. Smoothing with the class-based LM gave only a slight improvement, mostly due to performance on the Cell data. On the evaluation set performance gains are similar apart from applying the class LM.

As discussed above the second part of the system is split into three branches. The lowest WER is obtained in branch 1 using the MPE-SAT-HLDA model sets. Compared to (P3) the gain from using SAT, lattice MLLR with multiple speech transforms and FV transforms is 0.7% WER absolute. A further WER reduction by 0.7% is obtained when using CN decoding. For contrast purposes, the results for acoustic models as used in (P3) but with (P4) style adaptation was included in the table (labeled "Contrast"). Note that the effective gain from SAT after CN decoding is 0.3% absolute. The output of stage (P4.1) is used for adaptation in (P5.1) which, after CN decoding, gives an absolute WER of 25.7%. The second branch yields poorer performance due to lower complexity of the model set, but was found to be useful in system combination. Branch 3 differs from the first by the use of a SPron dictionary and non-SAT training, obtaining the same results as the contrast system. Note that in general the error rates on Cell data are lower for this branch. The gain from CN decoding is on average 0.7% absolute for the first branch, 1.1% on the second and 0.3%–0.7% for the third. The quinphone stages give only marginal improvements over the results of the triphone stages. Compared to the contrast system the performance of the SAT models is slightly better whereas the SPron quinphones give identical word error rates. The value of these systems lies in their contribution to system combination. Combining the output of the triphone stages (P4.[123]) gives a WER of 24.9% whereas the final result of CNC of all system output is 24.2%, or a 1.5% absolute gain from system combination.

The performance of the individual passes on the evaluation set is similar. The performance of the SPron triphone models was better giving the lowest triphone word error rate. Overall the gain from CN-decoding of quinphone model output was higher, especially in the case of SAT models with 1.1% WER absolute. The final word error rate of 23.9% was the lowest in the NIST RT-02 evaluations by a statistically significant margin [19]. The confidence scores obtained from confusion networks gave a normalized cross-entropy value (see, e.g., [28]) of 0.289 on eval02.

TABLE XII
EXECUTION TIMES OF DECODING STAGES USING IBM x330 SERVERS (WITH PENTIUM III 1 GHz PROCESSORS)

Pass	P1	P2	P3	P4.[123	P5.[123]
Speed (×RT)	12	11	37	131	147

TABLE XIII
%WERS AND REAL TIME FACTORS OF THE CU-HTK 2002 10 × RT SYSTEM ON THE eval02 TEST SET. SPEED AS MEASURED USING AN AMD ATHLON XP 1900+

Stage	Comment	%WER	Speed (×RT)
P1-10x	initial trans., VTLN	45.2	1.65
P2-10x	LSLR, lat-gen, fgint02	28.5	5.36
P3-10x	MLLR, re-score, CN	27.2	2.24

Table XII shows the execution times for the recognition stages associated with each of the passes. The individual numbers exclude times for estimation of lattice-based MLLR. The overall system had a real-time (RT) factor of about 320. In comparison the result on eval02 after CN-decoding of (P3) output lattices is 26.7% WER absolute using only 67 × RT.

IX. BUILDING FASTER SYSTEMS

The system presented in the previous section was designed for optimal WER performance. In practical scenarios it is not feasible to take minutes of processing time for the transcription of a second of speech. Consequently there is considerable interest in research on how to modify recognition systems to yield optimal performance under computational constraints. One test-condition in the 2002 NIST Rich Text Evaluation focused on operation in less than 10 × RT. The system described before was modified to meet this objective. The first part of the full system has relatively low complexity. Thus this part was chosen to form the basis of development. Several issues were important in development: the processing stages with relatively low gain but high computational costs were excluded (for example lattice-based MLLR); Lattices allow fast decoding with complex models or adaptation strategies, but a three stage approach to lattice generation is too costly; pruning parameters in the decoding stages can be tuned to substantially decrease the real-time factor with moderate degradation in WER performance; the use of faster computers with local disk storage allows for considerable speed improvements.

The final system is structured as follows: The first stage is identical to the full system P1 pass, however, much tighter pruning in decoding is used. In the second stage fast adaptation using least squares linear regression (LSLR) is performed. Using the HLDA MPE triphone models lattices are produced with the interpolated trigram LM tgint02 and further re-scored using fgint02; the output of this stage is used as supervision in adaptation using standard MLLR with two transforms for speech models and one for silence. Lattices are re-scored and CN decoding is performed on the output lattices.

Table XIII shows WER results for the CU-HTK RT-02 10 × RT evaluation system. The high error rate of the first pass gives only poor supervision, a second MLLR based rescoring step allows further improvements. Note that the final result of 27.2% is only 0.5% absolute from the CN output of the full system (P3) stage which took 67 × RT.

X. Conclusion

We have presented a complete state-of-the-art system for the transcription of conversational telephone speech and we described a range of techniques in acoustic, pronunciation and language modeling specifically important for this task. Particularly powerful methods in acoustic modeling are the use of side-based cepstral normalization, VTLN, discriminative training using the MMI or MPE criteria, and heteroscedastic LDA. Speaker adaptation using standard or lattice-based MLLR and full variance transforms yields considerable word error rate improvements. In language modeling the use of a background Broadcast News corpus together with class based language models allows to reduce the effect of the general lack of training data for this task. Pronunciation probabilities give consistent performance improvements. The use of lattices allow the use of confusion network decoding and the efficient implementation of system combination. We have discussed several systems with similar performance and their use in system combination.

Overall the word error rate achievable on the original Swbd1 corpus is now below 20%. More natural data is available in the form Swbd2 and Cell data where the error rates are just below 30% absolute. The reasons for these high error rates are manifold and can only partly be attributed to lack of data. Undoubtedly error rates are still too high for many applications but development of speech recognition systems for this task is an ongoing process.

Acknowledgment

Many people have contributed to the development of HTK LVR systems in the authors' group and thus have indirectly helped in putting this system together. The authors would like to especially acknowledge the work of T. Niesler and E. Whittaker who helped to build earlier Hub5 systems.

References

[1] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proc. ICSLP*, 1996, pp. 1137–1140.

[2] S. F. Chen and J. Goodman, "An Empirical Study of Smoothing Techniques for Language Modeling," Computer Science Group, Harvard Univ., Cambridge, MA, Tech. Rep. TR-10-98, 1998.

[3] G. Evermann and P. C. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *Proc. Speech Transcription Workshop*, College Park, MD, 2000.

[4] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)," in *Proc. IEEE ASRU Workshop*, Santa Barbara, CA, 1997, pp. 347–354.

[5] M. J. F. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Comput. Speech Lang.*, vol. 10, pp. 249–264, 1996.

[6] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Comput. Speech Lang.*, vol. 12, pp. 75–98, 1998.

[7] ——, "Semi-tied covariance matrices for hidden Markov models," *IEEE Trans. Speech Audio Processing*, vol. 7, pp. 272–281, 1999.

[8] J.-L. Gauvain, L. F. Lamel, G. Adda, and M. Adda-Decker, "The LIMSI Nov93 WSJ System," in *Proc. SLT'94*, Plainsboro, NJ, 1994, pp. 125–128.

[9] L. Gillick, J. Baker, J. Baker, J. Bridle, M. Hunt, Y. Ito, S. Lowe, J. Orloff, B. Peskin, R. Roth, and F. Scattone, "Application of large vocabulary continuous speech recognition to topic and speaker identification using telephone speech," in *Proc. ICASSP'93*, 1993, pp. 471–474.

[10] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "SWITCHBOARD: Telephone speech corpus for research and development," in *Proc. ICASSP'92*, 1992, pp. 517–520.

[11] P. S. Gopalakrishnan, D. Kanevsky, A. Nadas, and D. Nahamoo, "An inequality for rational functions with applications to some statistical estimation problems," *IEEE Trans. Inform. Theory*, vol. 37, pp. 107–113, 1991.

[12] T. Hain and P. C. Woodland, "CU-HTK acoustic modeling experiments," in *Proc. NIST Hub5 Workshop*, Linthicum Heights, MD, 1998.

[13] T. Hain, P. C. Woodland, T. R. Niesler, and E. W. D. Whittaker, "The 1998 HTK system for transcription of conversational telephone speech," in *Proc. ICASSP'99*, 1998, pp. 57–60.

[14] T. Hain, P. C. Woodland, G. Evermann, and D. Povey, "The CU-HTK March 2000 Hub5E transcription system," in *Proc. Speech Transcription Workshop*, College Park, MD, 2000.

[15] ——, "New features in the CU-HTK system for transcription of conversational telephone speech," in *Proc. ICASSP'01*, Salt Lake City, UT, 1999.

[16] T. Hain, "Implicit modeling of pronunciation variation in automatic speech recognition," *Speech Commun.*, 2003, to be published.

[17] R. Kneser and H. Ney, "Improved clustering techniques for class-based statistical language modeling," in *Proc. Eurospeech'93*, Berlin, Germany, 1993, pp. 973–976.

[18] N. Kumar, "Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition," Ph.D. Thesis, Johns Hopkins Univ., Baltimore, MD, 1997.

[19] A. Le and A. Martin, "The 2002 NIST RT evaluation speech-to-text results," in *Proc. Rich Transcription Workshop 2002*, 2002.

[20] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," in *Proc. ICASSP'96*, 1996, pp. 353–356.

[21] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," *Comput. Speech Lang.*, vol. 9, pp. 171–186, 1995.

[22] ——, "Flexible speaker adaptation using maximum likelihood linear regression," in *Proc. Eurospeech'95*, Madrid, Spain, 1995, pp. 1155–1158.

[23] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus among words: lattice-based word error minimization," in *Proc. Eurospeech'99*, Budapest, Hungary, 1999, pp. 495–498.

[24] S. Martin, J. Liermann, and H. Ney, "Algorithms for bigram and trigram clustering," in *Proc. Eurospeech'95*, Madrid, Spain, 1995, pp. 1253–1256.

[25] J. McDonough and W. Byrne, "Single-pass adapted training with all-pass transforms," in *Proc. Eurospeech'99*, Budapest, Hungary, 1999, pp. 2737–2740.

[26] A. Nadas, D. Nahamoo, and M. A. Picheny, "On a model-robust training algorithm for speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1432–1435, 1988.

[27] T. R. Niesler, E. W. D. Whittaker, and P. C. Woodland, "Comparison of part-of-speech and automatically derived category-based language models for speech recognition," in *Proc. ICASSP'98*, Seattle, WA, 1998, pp. 177–180.

[28] The NIST Speech Group. (2000) The 2001 NIST Evaluation Plan for Recognition of Conversational Speech Over the Telephone. [Online] Available: www.nist.gov/speech/tests/ctr/h5_2001/h5-01v1.1.pdf

[29] Y. Normandin, "An Improved MMIE training algorithm for speaker independent, small vocabulary, continuous speech recognition," in *Proc. ICASSP'91*, Toronto, ON, Canada, 1991, pp. 537–540.

[30] J. J. Odell, V. Valtchev, P. C. Woodland, and S. J. Young, "A one pass decoder design for large vocabulary recognition," in *Proc. 1994 ARPA Human Language Technology Workshop*, 1994, pp. 405–410.

[31] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. ICASSP'02*, Orlando, FL, 2002.

[32] L. F. Uebel and P. C. Woodland, "Speaker adaptation using lattice-based MLLR," in *Proc. ISCA ITRW Adaptation Methods for Speech Recognition*, Sophia Antopolis, Greece, 2001.

[33] P. C. Woodland, M. J. F. Gales, D. Pye, and S. J. Young, "Broadcast news transcription using HTK," in *Proc. ICASSP'97*, Munich, Germany, 1997, pp. 719–722.

[34] P. C. Woodland and D. Povey, "Large scale discriminative training for speech recognition," in *Proc. ISCA ITRW ASR2000*, Paris, France, 2000, pp. 7–16.

[35] P. C. Woodland, "The development of the HTK broadcast news transcription system: an overview," *Speech Commun.*, vol. 37, pp. 47–67, 2002.

[36] P. C. Woodland, G. Evermann, M. J. F. Gales, T. Hain, X. L. Liu, G. L. Moore, D. Povey, and L. Wang, "CU-HTK April 2002 switchboard system," in *Proc. Rich Transcription Workshop*, Vienna, VA, 2002.

[37] S. J. Young, G. Evermann, T. Hain, D. Kershaw, G. L. Moore, J. J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland, *The HTK Book*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

Thomas Hain (M'95) studied electrical engineering at the University of Technology, Vienna, Austria, and received the Ph.D. degree from Cambridge University, Cambridge, U.K.

He has been working in the field of speech processing and speech recognition since 1993. In 1997, he joined the Speech, Vision and Robotics Group, Cambridge University Engineering Department (CUED), as Research Associate to work on large vocabulary speech recognition systems. In 2001, he continued to work as a Lecturer at CUED. In 2004 he joined the Department of computer Science, Sheffield University, Sheffield, U.K. His main research interests are in automatic speech recognition, low bit-rate speech coding, and machine learning.

Philip C. Woodland (M'90) is currently Professor of information engineering with the Engineering Department, Cambridge University, Cambridge, U.K., where he has been a member of faculty staff since 1989. His research interests are in the area of speech technology, with a focus on all aspects of large vocabulary speech recognition systems. Other work has included auditory modeling, statistical speech synthesis, named entity recognition, and spoken document retrieval. He has led the development of CUED large vocabulary speech recognition systems for the last decade. He was a Director of Entropic from 1995 until 1999. He is a member of the editorial board of *Computer Speech and Language*.

Mr. Woodland is a former member of the IEEE Speech Technical Committee.

Gunnar Evermann (S'04) received the M.Phil. degree in computer speech and language processing from the University of Cambridge, Cambridge, U.K., and the B.Sc. degree in computer science from the University of Hamburg, Hamburg, Germany.

He is currently a Research Associate in the Engineering Department, Cambridge University. His main research interests are large vocabulary decoding and LVCSR system design.

Mark J. F. Gales (M'97) received the B.A. degree in electrical and information sciences from the University of Cambridge, Cambridge, U.K., in 1988. In 1995, he received the Ph.D. degree for a doctoral thesis supervised by Prof. Steve Young.

In 1991, he took up a position as a Research Associate in the Speech Vision and Robotics Group, Engineering Department, Cambridge University. From 1995 to 1997, he was a Research Fellow at Emmanuel College, Cambridge. He was then a Research Staff Member in the Speech Group at the IBM T. J.Watson Research Center, Yorktown Heights, NY, until 1999. He is currently a University Lecturer with the Engineering Department, Cambridge University, and a Fellow of Emmanuel College.

Dr. Gales is a member of the IEEE Speech Technical Committee.

Xunying Liu (S'04) received the M.Phil. degree from the University of Cambridge, Cambridge, U.K., and the B.Sc. degree from Shangai Jiao Tong University. Since 2001, he has been a Ph.D. student at Cambridge University.

His research interests include model complexity control, dimensionality reduction schemes, and discriminative training.

Gareth L. Moore received the Ph.D. degree in topic adaptive class-based language modeling and the M.Phil. degree in computer speech and language processing from Cambridge University, Cambridge, U.K., as well as a B.Sc. degree in computer science from Warwick University, Warwick, U.K.

He was a member of the Engineering Department, Cambridge University, from 1996 to 2002, and now works on all aspects of speech recognition systems for SoftSound, Ltd., Cambridge.

Dan Povey received the B.A. and M.Phil. degrees from the University of Cambridge, Cambridge, U.K., and has completed his Ph.D. work at Cambridge University.

His speech recognition interests include discriminative training. He is currently working at the IBM T.J. Watson Research Center, Yorktown Heights, NY.

Lan Wang (S'04) received the B.Eng. degree in electrical and electronic engineering from Beijing Institute of Technology, China, and the M.Eng. degree in signal processing from the Peking University, China. She is now pursuing the Ph.D. degree at Cambridge University, Cambridge, U.K.

Her main research interests are discriminative adaptive training and discriminative adaptation for automatic speech recognition.