Technology and Corpora for Speech to Speech Translation
http://www.tc-star.org

Project no.:	FP6-506738
Project Acronym:	TC-STAR
Project Title:	Technology and Corpora for Speech to Speech Translation
Instrument:	Integrated Project
Thematic Priority:	IST

Deliverable no.: D7
Title: ASR Progress Report

Due date of the deliverable:	1st of April 2005
Actual submission date:	13th of May 2005
Start date of the project:	1st of April 2004
Duration:	36 months
Lead contractor for this deliverable:	LIMSI-CNRS
Authors:	J.L. Gauvain, L. Lamel, H. Schwenk (LIMSI), F. Brugnara (ITC-irst), R. Schlueter, M. Bisani (RWTH), S. Stüker, T. Schaaf, S.I. Mohammed (UKA), M. Bacchiani, M. Westphal (IBM), S. Sivadas, I. Kiss (Nokia), F. Giron (Sony)

Revision: 10-May-05

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium(including the Commission Services)	
CO	Confidential, only for members of the consortium(including the Commission Services)	

Table of Contents

1 Introduction

The main WP2 objectives for the first year of the project were to first develop baseline Automatic Speech Recognition (ASR) systems for the three TC-STAR languages (English, Spanish, and Mandarin) and then to develop models and algorithms to better target the TC-STAR data (European Parliament Plenary Sessions for English and Spanish, Voice of American broadcasts for Mandarin) thus significantly improving the system accuracies. Another aim of this WP is to ensure that the ASR system outputs are suitable for machine translation systems.

The activities within this workpackage are divided in 4 tasks: integration of ASR and translation (Task 2.1), new powerful learning techniques (Task 2.2), new powerful adaptation and factorisation methods (Task 2.3), and baseline systems and API (Task 2.4). For the first year of the project activities were planned on all 4 of these tasks.

The objectives of Task 2.1 are to develop models and data structures to enable an effective integration of the ASR systems with the MT systems, and to develop low complexity models and algorithms to facilitate the technology transfer to resource limited real-life platforms. This task includes three main subtasks: (i) global decoding for enriched transcription (meta-data), (ii) design of a proper interface for speech translation, (iii) and low complexity modeling and decoding.

Task 2.2 activities concern the development of learning methods that take better advantage of the available training data and knowledge about speech to build more precise acoustic and language models. This task includes four main subtasks: (i) automatic learning methods, (ii) integration of knowledge sources in the speech decoding process, (iii) new optimisation criteria, and (iv) Bayesian network for acoustic modeling.

The objectives of Task 2.3 are essential in the development of ASR systems as this task directly concerns the development of the robust and adaptive models as well as the adaptation mechanisms needed to make the technology viable for real-life applications. This task includes five main subtasks: (i) adaptive pronunciation model, (ii) acoustic model adaptation, (iii) open vocabulary, (iv) adaptation for home and office environments, and (v) adaptation for noisy mobile environments.

The last task (Task 2.4) has the objective of developing complete ASR systems that can evaluated and integrated with the MT systems. This task has three main subtasks: (i) development of baseline recognizers for the 3 languages of interest in TC-STAR, (ii) development of APIs to interface the recognition and translation components, and (iii) development and evaluation of prototype ASR systems for the EPPS data.

2 Work summary and highlights

As planned, work on ASR for the first six months has focused on developing the baseline systems for the two targeted tasks: broadcast news (BN) and European Parliament Plenary Speeches (EPPS). Although the BN task is viewed as a transition task, it also enables comparison of ASR technology using publicly available and widely used NIST benchmark test sets. The original plans foresaw the baseline evaluation only for the BN task, however given the early availability of some EPPS data it was decided to also evaluate on this task (ahead of schedule) in the baseline evaluation. Most of the partners involved in WP2 developed systems and submitted results for the baseline evaluation held in September 2004. The best baseline word error rate on the English EPPS data was 32% using manual segmentations, which is the number against which subsequent evaluation results can be compared to measure the project progress in ASR. The ASR research activities in the following six months has been culminated by the first formal TC-STAR evaluation in March 2005. Each WP2 partner submitted at least one system to one of the 3 evaluation conditions (English EPPS, Spanish EPPS, and Mandarin BN), with a total of 30 submissions with of contrastive conditions. The best word error rate on the English EPPS was 10.6% which represents a very significant improvement over the the baseline systems. Comparable results were also reported for EPPS Spanish (11.5%) and BN Mandarin (≃10% CER). In order to get the best possible ASR results for translation, a voting algorithm (ROVER) has been used to combine the system hypotheses for both English and Spanish, resulting in word error rates of 9.5% for English and 10.1% for Spanish.

3 Integration of ASR and translation (Task 2.1)

The partners involved in this task are ITC-irst, RWTH, LIMSI, UKA, and NOKIA. Activities are reported on generating enriched ASR transcriptions, where the enriched output includes case sensitive transcriptions, sentence break detection, confidences measures, word lattices and confusion networks. Experiments have been carried out to demonstrate the benefit of using n-best and lattice interfaces to improve the overall speech-to-text translation result. The main outcome of these activities are that a fully integrated speech translation should work best, i.e. is able to give results that are better than simply using the single best hypothesis as the interface for ASR/MT. This is exemplified by work on a fully integrated speech translation system based on finite state transducers. Work has also been carried out on reducing the footprint of the acoustic models and reducing the computational complexity of the decoder.

3.1 ITC-irst

In task 2.1, the activity carried out at ITC-Irst was aimed at providing suitable input for experiments performed in the parallel task 1.2 within WP1. In particular, appropriate tools were developed for generating n-best lists and word graphs, to be exploited by the multiple-hypotheses search algorithm under experimentation within the MT research.

3.2 RWTH

In speech translation, we are looking for a target language sentence e_1^I which is the translation of a speech utterance represented by acoustic observation vectors x_1^T. In order to minimize the number of sentence errors we maximize the posterior probability of the target language translation given the speech signal (see [36]):

$$\hat{e}_1^I = \underset{I,e_1^I}{\operatorname{argmax}} \max_{f_1^J} Pr(f_1^J, e_1^I) \cdot Pr(x_1^T | f_1^J).$$

Note that we made the natural assumption that the speech signal does not depend on the target sentence and that we approximated the sum over all possible source language transcriptions by the maximum.$Pr(f_1^J, e_1^I)$ refers to the translation model, while $Pr(x_1^T | f_1^J)$ may be a standard acoustic model. Compared to automatic speech recognition alone, the translation model is plugged in instead of the usual language model.

Based on this decision rule for speech translation, word lattices of automatic speech recognition hypotheses were exploited as input to the RWTH translation system which is based on weighted finite-state transducers [33]. RWTH could show that acoustic recognition scores of the recognized words in the lattices positively and significantly affect the translation quality. In the experiments, consistent improvements were obtained on three different corpora in comparison with translations of single best recognized results. In addition a fully integrated speech translation model was built and evaluated at RWTH.

The interface to translation is given by word lattices, which are defined as XML representations of finite state transducers representing the search space obtained from the speech recognition systems used. In the word lattices arcs are labeled with start and end time, the recognized entity (word, noise, hesitation, silence), the negative log probability of acoustic vectors between start and end time given the entity and the negative log language model probability of the entity. In a first step, all recognition entities that are not spoken words are mapped onto the empty arc label ϵ. As language model probabilities and the time information are not used in this approach, they were removed from the lattices and the structure was further compressed by applying ϵ-removal and determinization. As most of the subsequent translation experiments were done without pruning, this step significantly reduced runtime while preserving the translation performance.

Translation was performed directly from the word lattices produced by speech recognition and consistently benefited from the acoustic scores (cf. deliverable D5). The dependency of word lattice density on the translation quality was thoroughly analyzed and it could be shown that lattices with higher densities improved translation error measures. This lead to the conclusion that fully integrated speech translation should work. Consequently a fully integrated speech translation system prototype was implemented, where both the speech recognition model parts and the translation model parts were implemented efficiently using a generic finite-state toolkit which supports on-demand computation, which also was developed at RWTH [27].

3.3 LIMSI

LIMSI has developed case sensitive lexicons and language models for EPPS English and Spanish. A pronunciation graph is associated with each word so as to allow for alternate pronunciations, including optional phones to model reductions. For the English lexicon, mappings were developed to combine text sources to account for differences in British and American written forms and the pronunciations of new proper names occurring in the TCStar EPPS English training data were verified. Pronunciations for the Spanish 65k word lexicon are generated via letter to sound conversion rules, with a limited set of automatically derived pronunciation variants.

The ASR systems developed for the three languages generate word lattices and consensus networks with confidence scores. In addition, a decoding method has been developed to automatically locate sentence breaks in the audio stream. This additional information (in addition to the 1-best output) are intended to be used by the MT systems to improve the speech-to-text translation results.

3.4 UKA

UKA developed during the first year of the project an integrated speech-to-speech translation system, between English and Chinese. In developing the system UKA studied the possibilities in coupling the recognizer with the translation system. Experiments in providing more information from the speech recognition system to the translation system were performed. In experiments acoustic and language model scores from the source language were provided to the translation model as additional information. UKA also studied the possibilities of providing a representation of competing recognizer hypotheses to the translation system. By comparing different representations, such as n-best lists and word lattices, UKA came to the conclusion that confusion networks are a suitable interface for transporting this kind of information to the recognition system. On the border between the speech recognizer and the spoken language translation system UKA studied the influence of a disfluency cleaning system.

3.5 NOKIA

The main memory and computationally intensive components of an ASR system are recognition networks, acoustic models and N-gram Language Models (LM). Nokia achieved significant progress in reducing the footprint of the acoustic models and reducing the computational complexity of the decoder. The details are summarised in the following paragraphs.

Finite State Transducers (FST) are an attractive choice for reduction in recognition network size. Although they provide compact recognition networks, the general FST operations require relatively large amounts of memory. For the Wall Street Journal(WSJ) 5000 words bigram task, FST reduced the network size by a factor of 3-4 when compared to a linear recognition network as used in HMM Tool Kit (HTK). The relative gain increases with the size of the vocabulary/language model. A general implementation of the FST operations required an order of magnitude more process size. This can be minimized dramatically through a task specific implementation.

On the LM front, work has started on class based models. Class based approach not only reduces the size of the LM but also generalizes commonly occurring out of vocabulary words such as proper nouns.

To reduce the memory requirement of acoustic models, Nokia investigated scalar quantization of acoustic model parameters and subspace clustering of Gaussian components. An acoustic model with 58,000 Gaussian components with 39 elements in the feature vector requires 2.66 Mega bytes with an 8 bit scalar quantizer and 1.58 Mega bytes with a 4-bit vector quantizer using subspace-clustering algorithm. Both the approaches were evaluated on the first EPPS English transcription task. The results showed only a minor degradation in performance compared to the non-quantized models.

The most computationally intensive part is the calculation of observation probabilities. Nokia investigated various techniques to reduce the computation such as calculating observation probabilities for every second or third feature vector, selection of most likely Gaussian components based on a codebook and dropping the feature vectors marked as non-speech by a VAD. All these approaches were tested on an in-house large vocabulary task. The results showed 30-50% savings in computational load without any reduction in recognition accuracy.

4 New powerful learning techniques (Task 2.2)

Activities are reported on automatic learning for acoustic and language modeling, on articulatory features for acoustic modeling, on discriminative acoustic model training, on connectionist language modeling and on Bayes risk minimization for decoding. Four partners contributed to this task: RWTH, LIMSI, UKA, and IBM. Automatic learning methods have been pursued to reduce the level of supervision and the cost of the training data to build models for ASR systems. Experiments were carried out using an acoustic model training procedure which only requires the raw orthographic transcription of the training data while keeping the same level of performance obtained with a detailed transcription including manual segmentation and speaker information. Methods were developed to automatically and semi-automatically acquire and use language model training data from the WEB to build language models for spontaneous speech recognition. Combining a number of articulatory acoustic features with standard acoustic features through linear discriminant analysis (LDA) or log-linear model combination has been investigated. Discriminative training experiments for acoustic modeling have been carried out, exploring a variety of discriminative training criteria including MMI, MWE, MPE and MCE. The performance of recently developed algorithms aiming at discriminative training of both the model parameters as well as the features through a linear transformation referred to as fMPE has also been investigated for the TC-Star tasks. A continuous space neural-network language modeling technique is reported on which can offer three advantages over regular n-gram language modeling: improved distribution smoothing, discriminative training, and potentially more effective LM adaptation.

4.1 Articulatory Features (RWTH)

At RWTH, a number of articulatory acoustic features were developed and combination of standard existing features and new articulatory features were investigated. In a first investigation, acoustic features were combined by Linear Discriminant Analysis, and/or individual acoustic models based on different acoustic features were combined by log-linear model combination. In [64] the two feature combination approaches were compared experimentally. Experiments performed on the large-vocabulary task VerbMobil II (German conversational speech) show that the accuracy of automatic speech recognition systems can be improved by the combination of different acoustic features. In [29], an additional novel articulatory motivated acoustic feature is introduced, namely the spectrum derivative feature. The new feature is tested in combination with the standard Mel Frequency Cepstral Coefficients (MFCC) and the voicing features. Linear Discriminant Analysis is applied to find the optimal combination of different acoustic features. Experiments have been performed on small and large vocabulary tasks. Significant improvements in word error rate have been obtained by combining the MFCC feature with the articulatory motivated voicing and spectrum derivative features: improvements of up to 25% on the small-vocabulary task and improvements of up to 4% on the large-vocabulary task relative to using MFCC features alone with the same overall number of parameters in the system.

4.2 Comparing Discriminative Training Criteria (RWTH)

By now, a number of different discriminative training criteria are used in state-of-the-art speech recognition systems, like the Maximum Mutual Information (MMI), the Minimum Word Error (MWE), or the Minimum Phone Error (MPE) criteria. Because of difficulties in both having an efficient implementation of Minimum Classification Error (MCE) training that takes into account a large number of competing word sequences and still being able to exclude the correct word sequence from the set of competing word sequences, efficient MCE implementations for large vocabulary speech recognition were not presented up to now or compared to existing discriminative training methods for large vocabulary speech recognition.

At RWTH, a number of discriminative training criteria were implemented for comparison purposes. Especially Minimum Classification Error (MCE) training (i.e. sentence error rate minimizing training) was implemented and tested for large vocabulary speech recognition. Whereas in earlier work on MCE training for large vocabulary speech recognition N-best lists had to be used to represent the set of competing sentences, at RWTH both the correct and the competing word sequences were represented by word graphs, which allowed for efficient calculation of the statistics for both the correct and the competing models. The MCE criterion was embedded into an extended unifying framework for a class of discriminative training criteria which allows for direct comparison of performance gains obtained with other discriminative training criteria. Criteria implemented and tested include, among others, MMI, MCE, MWE, MPE, the Gini, and the generalized Gini criterion. Experiments conducted on large vocabulary speech recognition (Wall Street Journal corpus) [35] showed a consistent performance gain of MCE on MMI training, and improvements obtained with MCE training were similar to those obtained with MWE training. For all criteria implemented, a method for the estimation of iteration constants for parameter optimization based on a positive variance constraint were used, which apply for both pooled variances, arbitrary tied variances, as well as density specific variances [34].

4.3 Bayes Risk Minimization (RWTH)

In speech recognition the standard evaluation measure is word error rate (WER). On the other hand the standard decision rule for speech recognition (maximization of the sentence posterior probability) is realized by using a sentence error based (or $0-1$) cost function for *Bayes* decision rule. Due to the complexity of the *Levenshtein* alignment needed to compute the number of word errors, for a long time it was prohibitive to use the number of word errors as cost function for *Bayes* decision rule. Although a number of approaches to *Bayes* risk minimization were developed recently, still all of these approaches rely on approximations to be efficient in practice.

At RWTH, fundamental properties of *Bayes* decision rule using general loss functions were derived analytically and were verified experimentally for automatic speech recognition [45]. It was shown that for maximum posterior probabilities larger than 1/2 *Bayes* decision rule with a metric loss function always decides for the posterior maximizing class independent of the specific choice of (metric) loss function. Also for maximum posterior probabilities less than 1/2 a condition is derived under which the *Bayes* risk using a general metric loss function still is minimized by the posterior maximizing class. For a speech recognition task with low initial word error rate it is shown that nearly 2/3 of the test utterances fulfill these conditions and need not be considered for *Bayes* risk minimization with *Levenshtein* loss, which reduces the computational complexity of *Bayes* risk minimization. In addition, bounds for the difference between the *Bayes* risk for the posterior maximizing class and minimum *Bayes* risk are derived which can serve as cost estimates for *Bayes* risk minimization approaches.

4.4 Lightly supervised training (LIMSI)

Lightly supervised acoustic model training has been attracting growing interest, since it can help to substantially reduce the development costs for speech recognition systems. Large quantities of audio data may be useful for acoustic model training if approximate transcriptions or related texts can be efficiently used, instead of detailed transcripts. Compared to supervised training with accurate transcriptions, the key problem in lightly supervised training is getting the approximate transcripts to be as close as possible to manually produced detailed ones, i.e. finding a proper way to provide the information for supervision.

Most of the acoustic models developed at LIMSI are now trained in a lightly supervised manner, where only the raw orthographic transcription is used (removing the need for manual segmentation, annotation of non-speech events labeling, or a complete detailed transcription). This training method will allow us to significantly reduce the data collection costs by eliminating some tedious manual work.

Light supervision was used to develop acoustic models for the LIMSI BN Mandarin system. There are only about 24 hours of data with accurate manual transcriptions available for training the acoustic models. However, there is a large quantity of Mandarin BN audio data from the TDT2, TDT3, and TDT4 corpora for which there are only closed-captions. Since in general closed-captions are much less precise than detailed manual transcriptions, previous work at LIMSI on light supervision used the closed-captions to provide indirect supervision via the language model as opposed to trying to align the captions as a reference transcription. Using the language model trained with the captions, a large corpus of unannotated audio data are transcribed automatically, and the recognition hypothesis are then used in forced alignment prior to carrying out standard EM training. A major problem with using the closed-captions for indirect supervision, is that while it reduces problems due to errors in the captions, acoustic model training assumes that the erroneous hypotheses are truth. Several approaches have been explored to remove probable recognition such as using confidence measures or by filtering the recognizer hypotheses with the closed-captions (it is extremely unlikely that the recognizer and the closed-caption both have the same error).

For TC-Star two lightly supervised approaches were explored. In the first a language model was estimated on all of the VOA captions, and this language model was interpolated with a general Mandarin language model and used to automatically transcribe the audio data. In the second approach the general Mandarin language model and global VOA language model were interpolated with a language model estimated on only the captions corresponding to the particular broadcast show. Training with 170h of VOA data with automatic transcripts produced by the second approach (show-specific language models) results in slightly better acoustic models. Training on the manually transcribed audio data pooled with the 170h of lightly transcribed VOA data reduces the character error rate by about 15%.

LIMSI also experimented with discriminative training (MMI) using automatically transcribed broadcast news data in American English, and will port this to the Mandarin system during the next period.

4.5 Connectionist Language Modeling (LIMSI)

It is difficult to build back-off n-gram language models for the EPPS tasks since only a limited amount of training data is available (about 35M words of EPPS transcripts collected by the TC-STAR consortium). In the open condition, additional general newspaper texts were used, but this was not allowed in the restricted condition.

To deal with this data sparseness problem, LIMSI has developed a neural network approach to language modeling [46, 47, 48]. The basic idea is to project the word indices onto a continuous space and to use a probability estimator operating on this space. Since the resulting probability functions are smooth functions of the word representation, better generalization to unknown n-grams can be expected, making by these means better use of the limited amount of training material. A neural network can be used to simultaneously learn the projection of the words onto the continuous space and the n-gram probability estimation. This is still a n-gram approach, but the LM posterior probabilities are ”interpolated” for any

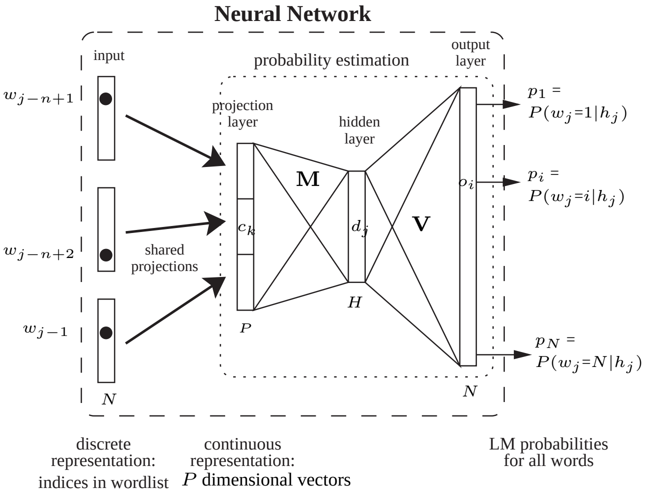

Figure 1: *Architecture of the neural network language model. h_j denotes the context $w_{j-n+1}, ..., w_{j-1}$. P is the size of one projection and H and N is the size of the hidden and output layer respectively. When shortlists are used the size of the output layer is much smaller then the size of the vocabulary.*

possible context of length n-1 instead of backing-off to shorter contexts. The neural network LM is used after the last decoding pass to rescore the lattices. This usually takes less than 0.1xRT.

The architecture of the neural network n-gram LM is shown in Figure 1. A standard fully-connected multi-layer perceptron is used. The inputs to the neural network are the indices of the $n-1$ previous words in the vocabulary $h_j = w_{j-n+1}, ..., w_{j-2}, w_{j-1}$ and the outputs are the posterior probabilities of *all* words of the vocabulary:

$$P(w_j = i|h_j) \qquad \forall i \in [1, N] \tag{1}$$

where N is the size of the vocabulary. The input uses the so-called 1-of-n coding, i.e., the ***i-th*** word of the vocabulary is coded by setting the ***i-th*** element of the vector to 1 and all the other elements to 0. The ***i-th*** line of the $N \times P$ dimensional projection matrix corresponds to the continuous representation of the i-th word. Let us denote c_k these projections, d_j the hidden layer activities, o_i the outputs, p_i their softmax normalization, and m_{jk}, b_j, v_{ij} and k_i the hidden and output layer weights and the corresponding biases. Using matrix/vector notation the neural network performs the following operations:

$$\begin{aligned} \mathbf{d} &= \tanh\left(\mathbf{M} * \mathbf{c} + \mathbf{b}\right) & (2) \\ \mathbf{o} &= \tanh\left(\mathbf{V} * \mathbf{d} + \mathbf{k}\right) & (3) \\ \mathbf{p} &= \exp(\mathbf{o}) \;/\; \sum_{k=1}^{N} e^{o_k} & (4) \end{aligned}$$

where lower case bold letters denote vectors and upper case bold letters denote matrices. The tanh and exp function as well as the division are performed element wise. The value of the output neuron p_i corresponds directly to the probability $P(w_j = i|h_j)$. Training is performed with the standard back-propagation algorithm using the cross-entropy as error function, and a weight decay regularization term. The targets are set to 1.0 for the next word in the training sentence and to 0.0 for all the other ones. It can be shown that the outputs of a neural network trained in this manner converge to the posterior probabilities. Therefore, the neural network directly minimizes the perplexity on the training data. Note also that the gradient is back-propagated through the projection-layer, which means that the neural network learns he projection of the words onto the continuous space that is best for the probability estimation task.

Table 1 summarizes the results for EPPS English and Spanish. The word error rates are lower than the ones of the official TC-STAR evaluation since neural network LM has not been used in our Spanish evaluation system due to time constraints, and the neural network LM for the English system has been improved since then.

	English EPPS				Spanish EPPS	
	Backoff LM		Neural LM		Backoff LM	Neural LM
	restricted	open	restricted	open	restricted	restricted
LM data	32M	478M	32M	32M	33.5M	33.5M
Perplexity dev	99.7	95.5	87.8	85.7	81.0	71.8
Word Error dev	12.13%	11.48%	11.26%	10.82%	10.64%	10.05%
eval	12.04%	10.98%	11.04%	10.48%	11.55%	11.07%
additional time	-	-	0.08xRT	0.09xRT	-	0.07xRT

Table 1: Comparison of the back-off with the neural network language model.

In the restricted condition the neural network LMs are trained on exactly the same data than the back-off LMs. A perplexity reduction of 12% relative was obtained for the English system (99.7 → 87.8). The word error rate improved by as much as 0.87% absolute (12.13 → 11.26%). The improvements brought by the neural network are slightly lower for the open system: 10% perplexity reduction and a gain of 0.66% absolute in word error. Both neural network LMs need less than 0.1xRT to rescore the lattices. It is also interesting to note the good generalization behavior of the neural network LM. Network selection and tuning of the parameters have been done on the development data, but the word error reduction obtained on the evaluation data (-1.00%), that was never used during development, is even better than the one obtained on the development data itself (-0.87%).

For the Spanish system the neural network LM achieves an improvement in perplexity of 10% relative and a word error reduction of 0.59% absolute. This gain is smaller than with the neural network LM for the English system, but still an significant improvement.

4.6 Improving Language Modeling with WEB Data (UKA)

UKA examined methods in automatically and semi-automatically acquiring language model training data from the world wide web. Here the difficulty is that the web-material cannot be easily classified in terms of publication date. Since for the evaluation all training material has to obey a cut-off date, we could not use this training material so far, since it was not possible to ensure the observation of the cut-off date. Comparative experiments were performed using the collected web data which did not yield any significant improvement so far, but which have not been pursued in depth for this evaluation because of the conflict with the evaluation conditions.

Also UKA started an investigation of different transformation techniques of language model training material in order to train language models for spontaneous speech using non-spontaneous training material from the same domain. In first step UKA started to implement and modify methods that can be found in literature. For the English EPPS task large amount of in-domain but non-spontaneous text material is available for training the language model. But only small amounts of spontaneous text material in form of the transcriptions of the acoustic training material is available. In this situation it is a standard technique to interpolate language models trained on the two different corpora. This can be viewed as a form of transforming the language model itself. But one could also think of transforming the training material instead, trying to make the large non-spontaneous corpus more spontaneous. For example one way in which the non-spontaneous material is different is that no filled pauses are present in the corpus. Using the small spontaneous material it is possible to estimate a model for the occurrence of filled pauses and to enrich the non-spontaneous material using this model. Also a weak language model calculated on the small amount of training material can be used as a measure of goodness for the training material in the large corpus, weighting its influence accordingly. This method is called weighted counting. The experiments are still on-going and no definite conclusions can be drawn so far.

4.7 Feature and Model Discriminative Training (IBM)

For the development of the English IBM system a significant amount of effort was put into investigating the performance of the recently developed algorithms for discriminative training. These algorithms aim at discriminative training of both the model parameters as well as the features through a linear transformation referred to as fMPE. Both model-space as well as feature space use the Minimum Phone Error (MPE) criterion as the figure of merit. The MPE objective function for discriminative training of acoustic models was previously described in [39]. The basic notion is the same as other discriminative objective functions such as MMI, i.e. training the acoustic parameters by forcing the acoustic model to recognize the training data correctly.

The MPE criterion is an average of the transcription accuracies of all possible sentences s, weighted by the probability of s given the model:

$$\mathcal{F}_{\mathrm{MPE}}(\lambda) = \textstyle\sum_{r=1}^{R} \sum_{s} P_{\lambda}^{\kappa}(s|\mathcal{O}_r)\mathrm{A}(s, s_r) \tag{5}$$

where $P_{\lambda}^{\kappa}(s|\mathcal{O}_r)$ is defined as the scaled posterior sentence probability $\frac{p_{\lambda}(\mathcal{O}_r|s)^{\kappa}P(s)^{\kappa}}{\sum_u p_{\lambda}(\mathcal{O}_r|u)^{\kappa}P(u)^{\kappa}}$ of the hypothesized sentence s, where λ is the model parameters and $\mathcal{O}_r$ the r'th file of acoustic data.

The function $\mathrm{A}(s, s_r)$ is a "raw phone accuracy" of s given s_r, which equals the number of phones in the reference transcription s_r for file r, minus the number of phone errors.

The MPE training process has been formulated for the case where time-marked lattices derived from recognition of the training data are available. Special attention is needed for the language model used when generating the lattices for this purpose. If the model relies on the language model too lightly or too heavily, it hampers learning where the acoustic model can benefit from a discriminative update.

MPE training involves collecting two sets of statistics similar to the sufficient statistics for ML training: one set for segments in the lattices whose acoustic likelihoods have a positive differential w.r.t. the MPE objective function, and one set where the differential is negative. The update is a version of the Extended Baum-Welch update, which is similar to the update equation used in ML training. Intuitively, the aim is to increase the probability of one set of statistics and decrease the other. In addition, the regular statistics for ML training are needed for backoff purposes (referred to as I-smoothing); this is important because if there is insufficient data for a particular Gaussian and no smoothing is used, discriminative training can lead to wildly inaccurate estimates of parameters.

Similarly, the discriminant feature space transform fMPE, is a way of increasing the MPE objective function by transforming the feature vectors. The first stage of fMPE is to transform the features into a very high dimensional space. A set of Gaussians is created by likelihood-based clustering of the Gaussians in SAT acoustic model to an appropriate size (in the EPPS experiments, 400 or 1000). On each frame, the Gaussian likelihoods are evaluated with no priors, and a vector of posteriors is formed. The vector is further expanded with left and right acoustic context similarly to what was done in the LDA of the speaker independent model.

The high dimensional features are projected down to the dimension of the original features $\mathbf{x}_t$ and added to them, so

$$\mathbf{y}_t = \mathbf{x}_t + \mathbf{M}\mathbf{h}_t \tag{6}$$

i.e. the new feature $\mathbf{y}_t$ equals the old features plus the high-dimensional feature $\mathbf{h}_t$ obtained as described above, times a matrix $\mathbf{M}$. Initializing $\mathbf{M}$ to zero gives a reasonable starting point for training, i.e. the original features.

The matrix is trained by linear methods, because in such high dimensions accumulating squared statistics would be impractical. The update on each iteration is:

$$M_{ij} := M_{ij} + \nu_{ij}\frac{\partial\mathcal{F}}{\partial M_{ij}}, \tag{7}$$

i.e. gradient descent where the parameter-specific learning rates are:

$$\nu_{ij} = \frac{\sigma_i}{E(p_{ij} + n_{ij})}, \tag{8}$$

where p_{ij} and n_{ij} (see below) are the sum over time of the positive and negative contributions towards $\frac{\partial \mathcal{F}}{\partial M_{ij}}$, E is a constant that controls the overall learning rate and σ_i is the average standard deviation of Gaussians in the current HMM set in that dimension. Since $\frac{\partial \mathcal{F}}{\partial M_{ij}} = p_{ij} - n_{ij}$, the most each M_{ij} can change is $1/E$ standard deviations, and the most any given feature element y_{ti} can change is n/E standard deviations, where n is the number of acoustic contexts by which the vector H_t has been expanded (e.g. n =7).

It follows from Equation 6 that

$$\frac{\partial \mathcal{F}}{\partial M_{ij}} = \sum_{t=1}^{T} \frac{\partial \mathcal{F}}{\partial y_{ti}} h_{tj}, \tag{9}$$

where h_{tj} is the j'th dimension of $\mathbf{h}_t$ and y_{ti} is the i'th dimension of the transformed feature vector $\mathbf{y}_t$. The differential $\frac{\partial \mathcal{F}}{\partial M_{ij}}$ is broken into the positive and negative parts needed to set the learning rate in Equation 8:

$$p_{ij} = \textstyle\sum_{t=1}^{T} \max(\frac{\partial \mathcal{F}}{\partial y_{ti}} h_{tj}, 0) \tag{10}$$

$$n_{ij} = \textstyle\sum_{t=1}^{T} \max(-\frac{\partial \mathcal{F}}{\partial y_{ti}} h_{tj}, 0). \tag{11}$$

Further details on computing the derivative of the MPE objective can be found in [38].

For the experiments on the EPPS development set, the starting point for discriminative training was the SAT model, i.e. a model obtained from VTLN and linear transform-based normalization. The discriminative approaches were applied in test after VTLN and CMA-based feature normalizations were applied. The performance of the system before discriminative training was 15.5%. When applying only MPE training of the model parameters (3 iterations), the performance improved to 14.3%. Using only the feature space transform fMPE, a single iteration improved the performance from 15.5 to 14.9. Three additional iterations improved the performance to 14.2%. Discriminative training of the model in the fMPE feature space dropped the error rate to 13.8%. These experiments used 400 clustered Gaussians for the fMPE algorithm. Performing a similar experiment with a larger number of Gaussians for the fMPE (1000 vs. 400) did not improve the performance (13.9 vs. 13.8). For the feature-space as well as model space training, lattices were generated with a bigram language model obtained by mixing a unigram and a bigram LM using a mixing weight of 0.5. Changing the LM to a model obtained by mixing the bigrams with a weight of 0.25 instead improved the first iteration error rate of the fMPE training to 14.2% but gave only a 0.1% gain at the 4th iteration (14.1 vs. 14.2). However, it also improved the model space training and resulted in the best error rate of 13.5 (vs. 13.8 using the stronger LM). Experiments setting the learning rate 8 more aggressively did not result in improved performance.

5 Powerful adaptation and factorization methods (Task 2.3)

Activities are reported on adaptive acoustic and language modeling, on noise robustness, auditory modeling, open vocabulary methods with automatic pronunciation generation, and on robustness in various noisy conditions. All WP2 partners (ITC-irst, RWTH, LIMSI, UKA, IBM, NOKIA and SONY) contributed to this task. Work was carried out on speaker adaptive acoustic modeling for unsupervised adaptation and such techniques were integrated in several systems submitted to the TC-STAR March'05 ASR evaluation. Adaptive language modeling techniques were explored for the broadcast news task. Adaptation techniques were investigated for porting models from one task to another. Experiments were carried out for open vocabulary speech recognition using the Wall Street Journal dictation task. Robust speech recognition work to deal with speech in noisy conditions (including background and reverberation noises) has also been carried out using a variety of methods (MVDR normalisation, TANDEM acoustic features, modified Wiener filtering, auditory and binaural models).

5.1 Acoustic normalization and adaptive training (ITC-irst)

In Task 2.3, the activity at ITC-Irst was focused on refining the method for acoustic normalization, named Constrained MLLR Speaker Normalization (CMLSN). The technique, consisting of a speaker-wise (or cluster-wise) normalization of the acoustic features towards a set of simple target models based on Constrained Maximum Likelihood Linear Regression (CMLLR), was applied and evaluated on Italian and English Broadcast News, and also on the EPPS task of TC-STAR.

Building on previous work already carried out at ITC-Irst, activity in the last year has focused mainly on generalizing and extending the technique in order to make its application in many different speech recognition tasks feasible [54]. In particular, the following aspects were considered:

- Generalization of the CMLSN adaptive training algorithm to many different sources of variability by performing cluster-wise normalization;
- Comparison of the performance of CMLSN with SAT, a popular adaptive training method;
- Comparison of target-models with different model structures, verifying that it is advantageous to use HMMs with a single Gaussian density per state;
- Introduction of a variant of the algorithm that can be applied in the first recognition pass;
- Development of an algorithm for porting an adaptively trained speech recognizer to a new task.

Each of the mentioned activities are briefly described in the following.

Cluster-wise normalization for adaptive training. In previous work [22], in which the CMLSN technique was compared to VTLN, segments were grouped according to given speaker labels. When the CMLSN procedure was applied to BN and EPPS tasks, the clustering was completely data-driven and automatic, based on the BIC criterion. This is still appropriate, because, unlike VTLN, the CMLLR transformation that is estimated and applied in CMLSN does not refer to any specific type of variability, and in fact the technique was found to be still effective.

Comparison to Speaker Adaptive Training (SAT). SAT is a popular technique for adaptive training of a speech recognizer. For these experiments, a variant of SAT was considered that has been introduced by Gales [16]. As in CMLSN, this variant of SAT utilizes CMLLR to reduce acoustic variability. The most prominent difference between SAT and CMLSN is that in SAT feature normalization and model training steps are interleaved and not performed one after another. As alternation of normalization and training requires sufficiently trained models, SAT is added on top of a conventional training procedure, and the state tying structure of the resulting models has to be the same of the initial models. On the other hand, when training models with CMLSN, the tying structure is determined exploiting normalized features, so that the state tying and the context-dependent allophones are consistent with the normalized feature space. In theory, this property should be advantageous, however, an experimental comparison was required as well.

Simple vs. complex target-models. The CMLSN algorithm makes use of two sets of acoustic models, the target-models and the recognition-models. The structure of the two model sets can be determined independently. In fact, there is a wide range of model types with different structural complexities that can be used as target-models. The most effective target models were found to be tied-state triphones with a single Gaussian per state, and this was verified comparing the performance of a system obtained by using the recognition models of a fully trained recognizer as target models.

Using a Gaussian Mixture Model (GMM) as a target-model. Leveraging again on the possibility to have different models for normalization and recognition, experiments were performed by using a single Gaussian Mixture Model (GMM) as target. This has the advantage that word transcriptions of test utterances are not required for estimating feature transformations. Thus, acoustic data normalization does not require a transcription provided by a preliminary decoding step, and the normalized models can be applied at the first recognition step. It turned out that also this simpler form of normalization was effective.

Table 2 summarizes results of some of the experiments mentioned above, obtained on the HUB4 corpus of American English broadcast news. For acoustic model training, the BN-E data released by the Linguistic Data Consortium in 1997 and 1998 were used. The corpora contain a total of about 143 hours of usable speech data. For evaluation, the 1998 HUB4 evaluation data were used, including about 3 hours of speech. The language model was trained on ≈132 million words of broadcast news transcripts distributed by LDC and on the transcripts of the BN-E training data.

	baseline	*SAT*	*simple-target*	*GMM*	*SAT-GMM*	*complex-target*
first pass	20.5	-	-	19.1	18.9	-
after MLLR	18.7	17.7	17.1	17.9	17.3	17.9

Table 2: *Word Error Rate of a baseline and different adaptively trained ITC-Irst systems on the HUB4 task*

Different recognizers are compared with a baseline, which consists of a conventional system with ≈9000 tied states and ≈146k Gaussians, trained with cluster based mean and variance normalization. All the considered system have a similar number of degrees of freedom. Values on the first row report performance at the first decoding step, and are therefore presented only for the baseline and GMM normalized systems. The second row reports results after three iteration of unsupervised static MLLR adaptation. For all the systems except *GMM* and *SAT-GMM*, supervision was provided by the baseline system. For the two GMM systems, supervision was provided by the output of the respective first decoding step.

For the system *GMM* the features have been normalized using CMLLR w.r.t. a mixture model that has been trained using ten iterations of the EM-algorithm. Based on preliminary experiments we decided to use 512 mixture components. The only difference between *SAT-GMM* and *GMM* is that the mixture model is trained with ten iterations of SAT. Clearly, a GMM is effective in reducing irrelevant speaker variability: the relative improvements in WER for *GMM* and *SAT-GMM* over the baseline are around 7%. However, the use of SAT training for the target model in *SAT-GMM* leads only to a small additional reduction in WER over *GMM*. The system *complex-target* uses the recognition-models of a fully adaptively trained recognizer as target-models. Each tied HMM-state of the target-models in this system corresponds to a mixture of Gaussians. The system *simple-target* is based on target-models that have just a single Gaussian per tied state.

As can be seen from Table 2, all adaptively trained systems lead to improvements over the adapted baseline. Best results are achieved for *simple-target*. A simple target model has the advantage, that it is not able to represent too much speaker variability in its output densities when it is trained on unnormalized data and thus may force a stronger normalization on the data. Thus, it seems reasonable to prefer a simple target model over a complex one. A GMM has an even simpler structure. However, a GMM is not able to represent phonetic variability as precisely as HMMs.

Porting of adaptively trained acoustic models. In order to perform a rapid system development for new tasks using a limited amount of data, a technique called *porting* can be applied. This method requires a set of acoustic models that have been trained on a large data set and adapts them to the new task by performing supervised MLLR adaptation on a smaller set of development data that represents the new task. In the project this method has been extended in order to perform porting of models trained with CMLSN. The procedure first adapts target models to the development data, then transforms the data with the adapted models, and finally performs adaptation of the recognition models to the transformed development data. Starting from acoustic models trained on the HUB4 American English broadcast news data, this procedure was applied to the TC-STAR-P Broadcast News English (BCAST-EN) corpus. A system trained with CMLSN on HUB4, and ported to BCAST-EN with this procedure, achieved a WER of 37.8%, with a 5% relative WER reduction with respect to the conventional porting of the baseline system.

5.2 Open Vocabulary Speech Recognition (RWTH)

The goal of open vocabulary speech recognition is a transcription system that can handle any spoken word without help from the user and without recourse to document meta data - even if this word was not seen at design time. To see how this could be achieved, we briefly review the decision rule and knowledge sources used by a standard large vocabulary speech recognition system:

$$\boldsymbol{w}(\boldsymbol{x}) = \operatorname*{argmax}_{\boldsymbol{w}'} p(\boldsymbol{w}') \max_{\boldsymbol{\varphi}} p(\boldsymbol{x}|\boldsymbol{\varphi}) p(\boldsymbol{\varphi}|\boldsymbol{w}') \tag{12}$$

The acoustic model $p(\boldsymbol{x}|\boldsymbol{\varphi})$ relates acoustic features $\boldsymbol{x}$ to phoneme sequences $\boldsymbol{\varphi}$. A pronunciation lexicon $p(\boldsymbol{\varphi}|\boldsymbol{w})$ assigns one (or more) phoneme string(s) $\boldsymbol{\varphi}$ to each word $w \in V$. The language model $p(\boldsymbol{w})$ assigns probabilities to sentences over a finite vocabulary $\boldsymbol{w} \in V^*$. A theoretically stringent approach to open vocabulary recognition is to conceptually abandon the words in favor of individual letters. Unlike words, the set of different letters G in a writing system is finite. Concerning the link to the acoustic realization, the set of phonemes Φ can also be considered finite for a given language. These considerations suggest the following model:

$$\boldsymbol{g}(\boldsymbol{x}) = \operatorname*{argmax}_{\boldsymbol{g}'} p(\boldsymbol{g}') \max_{\boldsymbol{\varphi}} p(\boldsymbol{x}|\boldsymbol{\varphi}) p(\boldsymbol{\varphi}|\boldsymbol{g}') \tag{13}$$

Here the acoustic model is in unchanged. The finite lexicon is replaced by a pronunciation model $p(\boldsymbol{\varphi}|\boldsymbol{g})$, which provides a pronunciation $\boldsymbol{\varphi} \in \Phi^*$ for any string of letters $\boldsymbol{g} \in G^*$. A sub-lexical language model $p(\boldsymbol{g})$ assigns probabilities to arbitrary character strings $\boldsymbol{g} \in G^*$. Alternatively the pronunciation model and sub-lexical language model can be combined into a joint “graphonemic” model $p(\boldsymbol{\varphi}, \boldsymbol{g})$.

Obviously this approach to open-vocabulary recognition is strongly connected to grapheme-to-phoneme conversion (G2P), where we seek the most likely pronunciation for a given orthographic form:

$$\boldsymbol{\varphi}(\boldsymbol{g}) = \operatorname*{argmax}_{\boldsymbol{\varphi}' \in \Phi^*} p(\boldsymbol{\varphi}', \boldsymbol{g}) \tag{14}$$

In particular “graphonemic” joint sequence models have been shown to perform very well on G2P tasks. (e.g. [12, 7]) The underlying assumption of this model is that for each word its orthographic form and its pronunciation are generated by a common sequence of graphonemic units. Each unit is a pair $q = (\boldsymbol{g}, \boldsymbol{\varphi}) \in Q \subseteq G^* \times \Phi^*$ of a letter sequence and a phoneme sequence of possibly different length. We refer to such a unit as a “graphone”. The joint probability distribution $p(\boldsymbol{\varphi}, \boldsymbol{g})$ is thus reduced to a probability distribution over graphone sequences $p(\boldsymbol{q})$ which we model using a standard M-gram. The graphone-based model integrates very easily with the standard speech recognition architecture: Any graphone can simply be added to the normal pronunciation dictionary. I.e. we combine the lexical entries with the (sub-lexical) graphones derived from grapheme-to-phoneme conversion, to form a unified set of recognition units $U = V \cup Q$. From the perspective of OOV detection the sub-lexical units Q have been called “fragments” or “fillers”. but are typically not associated with spelling information. By treating words and fragments uniformly the decision rule becomes

$$\operatorname*{argmax}_{\boldsymbol{u} \in U^*} p(\boldsymbol{x}|\boldsymbol{u}) p(\boldsymbol{u}) \tag{15}$$

The sequence model $p(\boldsymbol{u})$ can be characterized as “hybrid” because it contains mixed M-grams containing both words and fragments. It can also be characterized as “flat”, as opposed to structured approaches that predict and model OOV words with different models.

While various refinements and extensions of this model are possible, we have focused so far on evaluating the potential of the flat-hybrid model. Tests have been carried out on the Wall Street Journal dictation task (to be published in [9]). In all tested circumstances the flat hybrid model performs better than the corresponding baseline system. For the 20k vocabulary baseline system, which has an OOV rate of 2.6% on the test set, word error rate improves from 11.58% to 9.79% (15% relative). As expected the improvement in error rate depends strongly on the OOV rate: For very high OOV rates above 10%, error rate reductions of over 30% relative are possible. And even for very low OOV rates the performance does not deteriorate but is still slightly better than baseline. Future work includes tests on the EPPS tasks as well as investigations into advanced OOV modeling.

5.3 Cross-system adaptation (LIMSI)

Unsupervised acoustic model adaptation is used in many ASR systems, in particular when multiple pass decoding is a viable solution. Such adaptation is usually done using the Maximum Likelihood Linear Regression (MLLR) technique [30] using the hypotheses of a first decoding pass as supervision. In the LIMSI systems, the unsupervised acoustic model adaptation is generally done by grouping the HMM states into two classes (speech and non-speech) and by estimating a full regression matrix for each class to adapt the Gaussian mean vectors. This is also often preceded by a single class constrained MLLR transformation [16], and the adaptation procedure is often applied to speaker adaptive acoustic models.

LIMSI found however that for cross-system adaptation (for example using the UKA Mandarin system hypotheses to adapt the LIMSI system) much better results could be obtained by using a larger number of transformations depending upon the amount of available adaptation data. This is particularly attractive for broadcast news data where the amount of speech per speaker cluster is quite variable. A new technique was developed where the HMM states are clustered using a decision tree relying of a set of questions relative to the state position and to the phonemic classes. This decision tree is then used during decoding to determine the state classes based on the amount of data available at each node of the tree. Although no gain was obtained using this approach in a multipass system based only on LIMSI models, significant gains were observed for cross system adaptation. Using this method the character error rate on the Mandarin DEV data with LIMSI acoustic models was reduced from 11.9% to 10.9% after acoustic model adaptation based on UKA hypotheses with a CER of 13.9%.

LIMSI also experimented with various ways of improving unsupervised acoustic model adaptation by using acoustic model sets suited to maximize the gain with cross-model adaptation. The resulting knowledge was used in designing the LIMSI EPPS systems submitted to the March 2005 evaluation. Discriminative training was also used to build the acoustic models for these systems.

5.4 Language model adaptation (LIMSI)

Unsupervised language model adaptation has been investigated for both BN Mandarin and English using information retrieval methods. While n-gram models are successfully used in speech recognition, their performance is influenced by any mismatch between the training and test data. The idea of language model (LM) adaptation is to use a small amount of domain specific data to adjust the LM to reduce the impact of linguistic differences between the training and testing data.

Broadcast news transcription is a complicated task for language modeling since the content of BN data is open and any given BN show covers multiple topics. As a consequence, it is difficult to predict the topics of a BN show without looking at the data itself. The only information that is available for the show are the hypotheses output from the speech recognizer. However, for any given broadcast, the number of words in the hypothesized transcript is quite small and contains recognition errors. Therefore the transcripts are not sufficient for use as an adaptive corpus. Information retrieval methods provide a means to address this problem. Instead of directly using the ASR hypotheses for LM adaptation, they can be used as queries to an IR system in order to select additional on-topic adaptation data from a large general corpus. This approach reduces the effect of transcription errors in the hypotheses and at the same time provides substantially more textual data for LM estimation.

The performances of a variety of popular techniques for LM adaptation using automatically selected adaptation data has been compared. The investigated techniques are linear interpolation, maximum a

Test *Set*	*base* *line*	*MDI*	*MAP*	*Linear* *interp.*	*Mixture* *models*	*Dynamic* *mixture*
bn99en_1	280.9	260.2	262.0	250.7	249.7	238.1
bn99en_2	269.6	250.7	252.2	244.0	242.1	235.9
bn99en_1	18.3	18.1	18.3	18.2	18.3	17.9
bn99en_2	16.3	15.9	16.3	16.0	16.0	16.1
average	17.1	16.8	17.1	16.9	16.9	16.8

Table 3: Comparison of perplexity and word error rate (%) for the 5 different adaptation methods for the English BN system.

Test Set *h4ne97ma*	*Base* *line*	*MDI*	*MAP*	*Linear* *interp.*	*Mixture* *models*	*Dynamic* *mixture*
perplexity	447.0	381.3	412.8	389.7	376.4	388.8
CER	17.8	17.2	17.7	17.4	17.4	17.4

Table 4: Comparison of perplexity and character error rate (%) for the 5 different adaptation methods for the Mandarin BN system on the NIST 1997 evaluation data (h4ne97ma).

posteriori (MAP) adaptation, mixture models, dynamic mixture models, and minimum discrimination information (MDI) adaptation. To address the changing property of BN data, static and dynamic models for LM adaptation are investigated. In static modeling the LM is updated once for the whole show, which means that the LM must be simultaneously fit to multiple topics. Dynamic modeling updates the LM at each automatically detected story change, which entails estimating multiple story-based LMs for each BN show. Experiments were carried out for BN transcription in American English and Mandarin Chinese [14].

The NIST BN 1999 test data was used to evaluate the different adapted LMs for the English system [19]. This test set consists of 3 hours of audio data split in two subsets. The first set (bn99en_1) was taken from episodes broadcast in June 1998, and the second set (bn99en_2) was taken from a different set of shows broadcast in August/September of 1998. For Mandarin, the 1997 NIST Hub4 Mandarin evaluation data (h4ne97ma) containing 1h of speech was used for test purposes. All experimental results (in terms of perplexity and decoding error rates) are given for the individual test sets.

The 10x BN system [19] has 3 decoding passes: 1) initial hypotheses generation, 2) word graph generation, 3) final hypothesis generation. In these experiments, the hypotheses of the first decoding pass is used to generate a topic dependent LM, which is used in the second and third decoding passes.

The top part of Table 3 gives the perplexities and the bottom part the word error rates for the English BN system on the NIST 1999 test data for the 5 adaptation methods, as well as the baseline. The MDI and dynamic mixture adaptation methods result in the best average performance, bringing a 0.3% absolute gain in WER. The MAP model is seen to give no gain over the baseline system.

Table 4 gives the results in terms of perplexity and character error rate for the Mandarin BN system on the 1997 NIST evaluation data. The results are similar to those observed for the English BN system, that is MDI adaptation gives the largest improvement, and the smallest gain is with the MAP adaptation. However, for the Mandarin system, dynamic mixture models, linear interpolation and mixture models all yield the same result in terms of CER, even though the dynamic mixture model gives the lowest perplexity.

The experiments reported above made use if show-based adaptive language models, i.e., the entire show which contains stories on a variety of topics, were decoded using a common adaptive LM. Therefore, the adaptive LM includes the adaptation information for multiple topics. However, for any

LM	*bn99en_1*	*bn99en_2*	*average*
baseline model	18.3	16.3	17.1
show-based MDI model	18.1	15.9	16.8
story-based MDI models	17.7	15.4	16.3

Table 5: WER (%) of the BN English system with story based LMs.

LM	*h4ne97ma*
baseline	17.8
show based MDI model	17.2
story based models	16.8

Table 6: CER (%) of the BN Mandarin system with story based LMs.

particular story, it is generally the case that only the information on a single topic is particularly useful for language modeling and the information from other topics may even have a negative effect.

The IR based adaptation corpus selection method was also used to extract story specific adaptation data for each story, and MDI adaptation was carried out to train an LM for every story. Since the story segmentation method starts from the result of audio partitioner, the story boundaries are aligned with the audio speech segments. Therefore, it is straightforward to use a different story based LM to decode the speech segments corresponding to the story. The results for English and Mandarin systems using story based LMs in the second and the third decoding passes are given in Tables 5 and 6 respectively. It can be seen that the story-based language models result in better performance than the baseline systems and the show-based adapted LMs. For the English system, the improvement with the best show-based LM (the MDI model and dynamic models) has an absolute gain of 0.3% whereas the story based LM gives a 0.8% absolute gain. For the Mandarin system, the gain of the best show based LM (the MDI model) is 0.6% absolute, while the story based LM bring a gain of 1.0%.

5.5 Improving Robustness to Noise (UKA)

For making acoustic models more robust to noise and adverse acoustic conditions UKA investigated the application of the Minimum Variance Distortionless Response to the English EPPS task which shows a lot of reverberation and some background cross-talk. The warped minimum variance distortionless response (MVDR) is a time domain technique to estimate an all-pole model using a warped short time frequency axis such as the Mel scale. The use of the MVDR eliminates the overemphasis of harmonic peaks typically seen in medium and high pitched voiced speech when spectral estimation is based on linear prediction (LP). Moreover, warping the frequency axis prior to MVDR spectral estimation ensures more parameters in the spectral model are allocated to the low, as opposed to high, frequency regions of the spectrum, thereby mimicking the human auditory system. Using the MVDR front-end the UKA English EPPS transcription system reached a word error rate of 15.2% on the 2005 development set.

As another alternative approach to robust preprocessing UKA started to investigate and implement TANDEM features for the English EPPS task. TANDEM is an additional step performed in the regular feature extraction block of the speech recognition system, to nonlinearly transform the regular features vectors in a data-driven manner. The nonlinear transformation is performed using a Multi-Layered Perceptron (MLP). It is well known from the previous literature that, when a MLP is trained with feature vectors at the input and phoneme labels per node at the output it learns to approximate the phonetic-posterior probability distribution of the input feature vector space. After having trained, the MLP will transform the input feature vector space onto the phonetic-posterior space. The main idea in TANDEM is to use such posterior space as the feature space, as that would constitute an optimal space if there is no confusion among phonemes in a given language. TANDEM can also be seen as a nonlinear equivalent of the Linear Discriminant Analysis (LDA). As like LDA, MLP also transforms feature vectors with some context to the posterior space.

For the experiments performed, Mel-Frequency Cepstral Coefficients (MFCC) of 13 static coefficients, 13 delta coefficients, and 13 acceleration coefficients (39-D) are used as the base features. A frame size of 16msec and a frame shift of 10msec is used. These features, with a context of 4 feature vectors to the left and 4 feature vectors to the right, are transformed to a phonetic-posterior space of dimension 50 with a MLP of size, 351 input units, 600 hidden units, and 50 output units, with softmax output function. This MLP has been trained with the feature vectors extracted from 40h of EPPS training data and phoneme labels obtained by a forced alignment of an existing recognition system. Usually it is not possible to use the MLP outputs directly as the feature vectors because the output distribution is highly skewed and also the output components are highly correlated. To handle this pre-nonlinearity outputs of the MLP are collected and a Karhunen-Loeve Transformation (KLT) is performed. A performance of 19.2% WER is obtained on the spring 2005 English EPPS development set.

5.6 Porting Acoustic Models Across Tasks (UKA)

For adaptation of the acoustic model UKA demonstrated that it is possible to port components of the acoustic model that has been trained on large out of domain data to the English EPPS task using the comparatively small amount of English EPPS data to adapt and retrain some of the model parameters such as the mean vectors and covariance matrices. The experiments were performed with the help of the ISL Meeting Transcription System. In a first step the cluster tree and the global semi-tied covariances transformation matrix were taken over and trained with the EPPS acoustic training material. The forced alignments of the training material were done with the help of a small broadcast news transcription system, the same that was used for the training of the Meeting Task System. The system yielded a word error rate of 16.8% in the first unadapted pass using a language model in compliance with the conditions of the restricted task. An analysis of the acoustic model revealed that some of the acoustic models did not receive sufficient, in some case no, training samples. An indication of the bad match of the polyphone cluster tree to the given training material. Using this system new forced alignments were written, the polyphone cluster tree was retrained and the global STC transformation matrix was re-estimated as well. But though we expected to see a better performance, since now the cluster tree matched the available training material and the STC transformation matrix now better matched the acoustic conditions, the system also only yielded a word error rate of 16.8%. This demonstrates that it is possible to reuse components of an acoustic model that has been trained on a completely different domain with different acoustic conditions.

5.7 Adaptive Acoustic Models (IBM)

The IBM system uses a static decoding graph approach for all decoding passes. The static decoding graph is constructed by first compiling the 4-gram language model consisting of 58k 1-grams, 2M 2-grams, 2M 3-grams and 1M 4-grams into a finite state automata G. The lexicon is then compiled into a transducer accepting phone strings on the input side and accepting the corresponding word strings on the output. This L transducer was constructed for a 58k entry lexicon. The pronunciations were derived from the widely used Pronlex lexicon and augmented with automatically designed pronunciations using a letter to sound algorithm for those entries found in the training text but not in Pronlex. The tied-state triphone HMMs as defined by the decision trees, built during the acoustic model design, are compiled into a transducer C accepting phone strings and matching context dependent HMM strings. The number of tied states used in the system was 6123. Finally, the HMM topology of the HMMs is expressed as a transducer H which maps HMM strings to strings of tied-states. The final static decoding graph is constructed by composition as

$$HoCoLoG.$$

The H, C and L transducers are also used in training where the acoustics are matched to the correct state sequence which in that case is constructed using the reference text string. In that case the reference string is compiled into a linear automata G to facilitate the same modeling approach as in testing. The acoustic model provides the likelihoods for the states that are allowable (weighted) strings as defined by the final static decoding graph.

The adaptive acoustic model consists of 4 parts. The first part is the speaker independent acoustic model which was estimated on PLP features after an LDA projection. For each 10 ms. frame a PLP analysis is performed resulting in a 13 dimensional observation vector. Then, an LDA projects 9 temporally consecutive frames into a 40 dimensional space in which the acoustic model operates. The LDA attempts to jointly maximize the inter-class distance and minimize the within-class distance. The class definition for this analysis are the tied states as defined by the decision trees. To optimize the training of the SI model, it is built iteratively. All training uses the Expectation-Maximization algorithm to estimate the mixture model parameters. The model complexity is slowly increased over the training process by iterating mixture component splitting and parameter updating using the EM algorithm, An initial context independent (CI) model was obtained by EM training a CI model from a flat start. Then, an initial segmentation of the training data was obtained using that 40 component mixture CI model. A context dependent (CD) model was then trained by building decision trees on that segmentation followed by mixture training of the CD model still using the segmentation obtained from the CI model. After the CD training is completed, the LDA is performed on the tied state classes and segmentation. In addition, speaker dependent feature transforms are computed using the same algorithm described below for the speaker adaptively trained model. Using the LDA and speaker dependent transformations, the training data is re-segmented and the training process, starting from the decision tree building is repeated. On the EPPS development set, this iterative training process improved the performance from 18.2% for the non-LDA initial training pass to 17.2% after 2 iterations.

The second part of the acoustic model incorporates normalization for the vocal tract length variation among speakers. This parts makes use of 2 acoustic models. The first, referred to as the voicing model, uses the same state tying as the SI model, however it uses full covariance Gaussian models instead of the diagonal covariance Gaussian mixture models used in the SI model. This voicing model is trained iteratively on speaker dependent frequency warped features. For each speaker the most likely frequency warping is estimated using an analysis by synthesis approach. Features are computed for various frequency warps and their likelihood is evaluated using the voicing model. Once the most likely warps are estimated, the voicing model is re-estimated. This process of re-estimating warpings and re-estimating the voicing model is repeated 4 times. Then a VTLN model is constructed by repeating the training procedure as used for the SI model but now on the frequency warped features. The state-tying is kept constant and hence matches that of the SI model, however the mixture distributions are re-estimated from scratch in the VTLN space. In test, the VTLN frequency warping is estimated using the last hypothesized transcript and segmentation, again using the voicing model and decoding the VTLN space is repeated using the VTLN model.

The third part of the acoustic model incorporates speaker variability by incorporation of linear transforms. It makes uses of the Constrained Model-space Adaptation algorithm which linearly transforms the model means and variances as

$$\mu' = H\mu + b$$

and

$$\Sigma' = H\Sigma H^T$$

respectively. The approach is referred to as constrained as it applies the same linear transformation H to both the means and variances of the model. A naive implementation of this approach becomes computationally prohibitive as it transforms a diagonal covariance model into a full covariance model unless the transformation is itself diagonal. However, the constraint has the advantage that the model can be implemented efficiently as a diagonal covariance model operating on linearly transformed features. In, fact if the transform H is invertible, the adapted model can be implemented by operating on features

$$\mathbf{x}'_t = A\mathbf{x}_t - b$$

where $\mathbf{x}_t$ is the original observation for time t and $A = H^{-1}$ and by including an additive determinant term $|H| = -|A|$ in the likelihood evaluation. It has been shown that the transform that maximizes the data likelihood given a state sequence can be estimated iteratively using the EM algorithm. To incorporate this speaker normalization in training, the transforms are computed for each training speaker and the Speaker Adaptive Training (SAT) model is computed by re-training the acoustic model on the speaker adaptive features. In test the transform is estimated like in training but on the last hypothesized transcript.

The fourth part of the adaptive acoustic model consists of another linear transform based adaptation pass using the Maximum Likelihood Linear Regression (MLLR) algorithm. This algorithm transforms the means of the model similarly to the CMA approach but retains the variances of the model before MLLR adaptation. As for CMA, there is an iterative update based on the EM algorithm that ensures monotonically non-decreasing likelihoods ensuring convergence to a maximum.

For the English development test set of the EPPS task, the adaptation algorithms improve the system performance from 17.2% for the SI system to 16.4% using only VTLN and further to 15.5% using both VTLN and CMA. MLLR provides a small additional gain further reducing the error rate to 15.2%. However, for the English system, the speaker normalized features were used as the input space for discriminative training and MLLR was added as an additional pass after the discriminative step where it provided a gain of only 0.1%. For the Spanish system, where only transform-based speaker normalization was used, the error rate on the development test set improved from 15.0% for the SI system to 13.0% after adaptation.

5.8 Noise Compensation for Noisy Mobile environments (NOKIA)

Sensitivity to environmental noise is the single most reason for performance degradation of ASR systems, especially in mobile environments. Nokia tested various spectral and cepstral domain noise compensation methods on an in-house multilingual isolated word recognition task under realistic noise conditions with Signal to Noise Ratios (SNR) ranging from 5dB to 20dB and clean. The noise spectral parameters are estimated using a VAD. They are updated during the pauses between the words. Using a modified Wiener filtering approach Nokia obtained a relative word error rate reduction of 20-25% in noisy conditions compared to the baseline system that uses mean and variance normalised cepstral features. Temporal filtering of features for noise robustness and improved discrimination of phonemic classes are being investigated.

5.9 Adaptation for Home and Office Environments (SONY)

It is widely recognized that one of the major flaws of current ASR technology is an excessive sensitivity to characteristics of the speech signal, which are not related to the linguistic content, like e.g. change in the acoustical environment, due to the room effect (reverberation), background noises (concurrent speaker(s), office, street noises, etc.) or signal conditioning (microphone and channel distortions). Mainly two different approaches have been envisaged up to now in the ASR community to try solving this problem.

One way to cope with it is factorization, that is to enrich the models so that they can cover all these aspects, so as to correctly classify many different observed phenomena into the same class (more Gaussians, more training data, condition-specific models, novel structures for the acoustic models...). This is the approach used currently by the majority of institutes or companies working in speech recognition, when dealing with new tasks. It has the advantage of giving currently the best results, but the drawbacks of being not portable to new tasks and environments. Moreover there is the need of collecting for each new task a non negligible number of training data. This can be very cumbersome and needs a lot of effort. For example, in the area of applications of ASR to consumer devices, the European Project SPEECON has been created specially with this respect of data collection; speech databases have been conceived and recorded with the purpose of covering typical situations for future products in this area and for several languages.

Another way is normalization, which attempts to reduce this variability by modifying the representation of the utterance. This may involve, for example, devising acoustic parameters based on auditory models that are less sensitive to irrelevant aspects of the signal.

A third way, combining factorization and normalization, tries to alleviate the data collection for new environmental conditions for a particular application or area of applications, by applying diverse signal processing methods to adapt existing speech database, e.g. recorded in a clean studio environment, in a way that the resulting speech signals are closer to those that would be recorded in the real environment of the application. The data obtained in this simulated way are then used to train an ASR system in the same way as it is done for the factorization approach.

Given the wide potential for speech recognition application in home and office environments, Sony has decided to concentrate its research efforts on investigating signal capture problems associated with noise and room reverberation. Unrestricted voice input in arbitrary home and office environments still requires major advances in far field speech recognition. Sony investigates methods for the combined modeling of background noise and far field speech. It is also necessary to be able to handle unexpected events and spontaneous effects found in home and office environments (door slams, background speech, telephone, ...). In order to deal with simultaneous speech from multiple speakers a binaural (dummy head) approach for source separation is also investigated.

In the following, a more detailed overview of the activities performed during the first year of the project in this area is given.

5.9.1 Auditory Model

Auditory models have been investigated for more than fifteen years now, with the ultimate goal of achieving the superb ability of the human auditory system to process speech in the presence of environmental disturbances. The most widely used Front-End systems in ASR, like Mel-frequency cepstral coefficients (MFCC), perceptual linear predictive analysis (PLP) or RASTA processing implement some auditory effects in a short-time fast (discrete) Fourier transform (FFT) framework [1]. Although more elaborated auditory models demonstrated their efficiency in many different specific tasks, like vowel-recognition in continuous speech [50], speaker-independent phoneme classification [2, 17], alphabet recognition [18] or isolated-words recognition [28], these models have not yet been applied in larger LVCSR systems to our current knowledge. One of the main reason could be the computation load, which is needed for these models, since the number of operations required can be between 35-600 times higher that a traditional LPC processing[1]. Another main difficulty encountered in the application of these models is to find an adequate set of features, which can be integrated easily in an ASR framework without loosing the valuable time-frequency discriminative properties of such a model. Although some carefully carried comparative study between more traditional features like MFCC and auditory models [26] did not show big improvement on word error rate in an isolated-word recognition task, the authors concluded that ”for a fair test, more work is necessary to obtain improved ways of incorporating features from auditory models into speech recognizers”. As these and other authors [24], Sony advocates also for the selective use of auditory knowledge, optimized on real speech data.

Following this research direction, Sony worked intensively on an reimplementation of a model developed originally by Stefanie Seneff [51, 49]. This model has the advantage of being structurally relatively simple, while reproducing important physiological properties of the auditory system, like saturation, adaptation and forward masking. It takes as input a waveform and computes a multi-channels (40) output corresponding to a probability of firing of the corresponding bank of critical-band filters. This model was rewritten from an initial implementation in Matlab to a complete implementation in C++ under a Linux environment to increase processing speed. As mentioned previously the speed factor is important, since this model will be used to process many hours of speech data in the case of a LVCSR system. This model has been modified in its original initialisation phase for a better integration with the Bochumer Binaural Model described in the next section.

Further, an implementation of the Generalized Synchrony Detection (GSD) approach [51, 49, 25, 18] was done under Matlab. This GSD post-processing delivers a pseudo-spectral representation with enhanced spectral contrast. An intensive research activity on different parameters configuration for this GSD and possible alternative features representation to currently used MFCC for robustness under noisy and reverberant conditions was also performed. This research is performed with the objective of preserving the auditory knowledge from these models, while allowing a good integration of the corresponding features in the framework of a Hidden Markov Model based speech recognizer.

Further near future activities in this area are the definition of such a front-end based on this kind of modeling as well as training and testing of a corresponding LVCSR system by using the experience gained meanwhile in settling this kind of system on a MFCC front-end as described in Section 6.7.

5.9.2 Binaural Model

Binaural models have been less investigated than auditory models in general for their application in the ASR field, but it is an approach, which has been followed more intensively in recent years by some institutes like the Ruhr-University of Bochum under the aegis of Prof. J.Blauert [3] or the Carl von Ossietzky Universitt under the aegis of Prof. B. Kollmeier [13]. This approach takes into account that the human auditory perception system is able to concentrate on a certain voice of interest while ignoring the presence of competing speech or noise to some extent. This ability of humans is the so-called cocktail party effect (or attentional selectivity), which is mainly based on the difference between the arrivals of acoustical signals at the two ears (physiological and spatial effect) and the listener's attention (neural processing) [6]. As compared to traditional microphone arrays, where mainly the time delays is used as information for processing, the binaural method takes also into account the distortion and diffraction effects, which occurs around the head and external ears. This influence can be coded with some approximation under what is called Head Related Transfer Function (HRTF). This complex transfer function is depending on azimuth, elevation and frequency and can be used to simulate the impinging sound from a free-field sound source from a particular direction.

The approach of Sony is to basically follow the approach envisaged by the previous institutes in implementing a binaural model [10, 11, 40, 13], which simulate this directional selectivity of the human auditory system and combine it with the previous auditory model.

During this first year period, some Head Related Transfer Function (HRTF) adaptation programs, which are used for the Binaural Model have been ported and adapted to work under Linux for compatibility with the Seneff model and again for speed issues. First output tests of the integrated model with binaurally recorded living room data were performed. During this test it became clear that the initialisation phase of the Seneff Model had to be modified, to cope with wrong localisation artefacts.

Further database-related activities included the labeling of an internal speech database recorded in an home-environment-like scenario, which also includes binaural recordings. This database will serve as a reference for further testing and evaluation of the auditory binaural approach in the case of concurrent speaker and/or running TV in a reverberant environment. These Binaural modeling activities are expected to be performed further only in the framework of a small vocabulary ASR system on the internal Sony living-room database, due to the availability of the corresponding binaural recordings.

6 Systems and API (Task 2.4)

The development of baseline systems has been the focus of much of the WP2 activities in the first few months of the project, culminating in the September 2004 dryrun evaluation. The baseline experiments described in the D6 report showed that the project partners have state-of-the-art technology for BN data in the three TC-STAR languages and that this technology could be successfully adapted to the EPPS domain with a limited amount of training data. The baseline error rates are used as reference numbers with which to demonstrate WP2 progress. All WP2 partners (ITC-irst, RWTH, LIMSI, UKA, IBM, NOKIA and SONY) have developed systems for EPPS tasks and submitted at least one system to the TC-STAR Feb'05 ASR evaluation.

6.1 ITC-irst

For Task 2.4, ITC-irst set up baselines to participate in the September 2004 preliminary evaluation. Results were submitted for the following tasks: American English BN, European English BN, European English parliament, English conference lectures. Details on the systems can be found in report D6. For the March 2005 evaluation, ITC-irst also set up a baseline system for the Spanish language. Adding support for the Spanish language required to develop an automatic grapheme-to-phoneme tool.

6.2 RWTH

The RWTH baseline system for English was derived from an existing American English Broadcast News recognition system. The RWTH baseline system for Spanish was developed within TC-STAR on the Spanish Broadcast News domain. These systems for were used for the September 2004 dryrun for two TC-STAR languages: EPPS English, TC-STAR_P European English and European Spanish Broadcast News, as well as the HUB4 English and Spanish. Starting from the baseline systems, speech recognition systems for the European English and European Spanish EPPS domain were developed using the EPPS language resources produced by RWTH and UPC. The EPPS systems were used for the first official TC-STAR automatic speech recognition evaluation campaign in March 2005.

6.3 LIMSI

LIMSI built baseline systems by updating their broadcast news transcription system for Mandarin Chinese and developing a broadcast news system for Spanish. For the September 2004 dryrun evaluation, LIMSI submitted results for the 3 TC-STAR languages: EPPS in English, BN TCSTAR_P in English and Spanish, and for BN Mandarin. LIMSI also provided speaker segmentations for the English data. With the availability of the EPPS English and Spanish training and development data, LIMSI developed EPPS systems for these two languages. LIMSI participated in the March 2005 evaluation, submitting systems for the 3 languages (English, Spanish, and a joint submission with UKA for Mandarin). For this evaluation the acoustic and language models were considerably improved from those of the baseline systems. LIMSI produced development packages for the EPPS English and Spanish data, including mapping rules for scoring, normalized reference transcripts (derived from those distributed by ELDA) and sentence level segmentations. The software to provide sentence level segmentations was delivered to ELDA so that the test data could be processed in an analogous manner. LIMSI generated word lattices for the English development data which were used in rescoring experiments by Nokia. LIMSI also carried out ROVER based system combination of the system outputs for the EPPS English and Spanish data, providing the best possible hypothesis for MT.

6.4 UKA

UKA constructed baseline recognitions systems by updating its Chinese Broadcast News transcription system and by porting its English Meeting Task System to the English EPPS and European English Broadcast News task. Using these systems UKA submitted hypotheses for two TC-STAR languages in the September 2004 baseline evaluation: EPPS in English, BN TCSTAR_P and Mandarin BN. For the March 2005 evaluation, UKA submitted a system for English EPPS and a joint BN Mandarin submission with LIMSI using cross-site adaptation.

6.5 IBM

For the baseline system contribution, IBM trained a state-of-the-art broadcast news system for both English and Spanish using the publicly available Broadcast News training material as released by LDC for the DARPA evaluations. The key decoding steps of the system consisted of:

1. segmentation into wideband and narrowband segments
2. A bandwidth-appropriate speaker-independent decoding
3. CMA transform estimation and decoding with SAT models
4. MLLR transform estimation and decoding with the MLLR-adapted models

All decodings were done using static decoding graphs representing the language model at the most atomic HMM-state level. Each decoding pass runs in approximately 4-times real time. The Spanish system is as above, excluding the speaker-adaptive steps. IBM's baseline results are on par with those reported by the other partners, and can be found in Report D6.

The IBM team improved its baseline systems through the re-use of discriminative training, including both model and feature-space training, iterated system building with re-alignment, optimized system sizing, and language model rescoring. In the English system, the team also incorporated a large amount of language modeling data from the web, and from transcribed news broadcasts. IBM submitted public and restricted systems for English EPPS, and a restricted system for Spanish EPPS.

6.6 NOKIA

Nokia in collaboration with LIMSI participated in the first evaluation campaign. The submitted results for EPPS English were obtained by rescoring word lattices generated by LIMSI. Nokia's primary system is a context dependent phoneme based recognizer. To evaluate the performance of low complexity acoustic modeling approaches, Nokia submitted contrast systems with Scalar quantized and Subspace distribution clustered acoustic models.

6.7 SONY

The main activity in this period concentrated on the settlement of a Baseline LVCSR system for development, testing and evaluation in a framework compatible with the TC-STAR requirements, thus allowing later comparison with other systems developed by the other consortium members. It should be underlined here that this task needed mostly lot of settlement, pre- and post-processing work, as is always needed in the ASR area, when starting with a new task. No specific effort was (and could have been) engaged in optimizing the existing system and technology to the particular domains requested for the evaluation.

The settlement of a Janus-based system for American English BN ASR was first performed using Sony studio data and a part of the 1996 English Broadcast News Speech data (LDC97S44). A more complete description of the structure of this simple basic system is given in section 8.7.

Secondly, the installation and test of different official NIST evaluation packages for the Hub4e98 task was performed as the next activity. The evaluation of the baseline version of the previous LVCSR system was performed on this task, resulting in a primary word error rate (WER) of 47.4%. Further tuning on language model parameters and beam width with the same system could lead to a word error rate of 31.6%.

Next for the EPPS English May 2003 task the necessary database preparation tools had to be installed or developed. The same Baseline system was evaluated on this task resulting in a WER of 50.7%. A cooperation with UKA kindly lending us their dictionary and language model could improve the results a little bit to a WER of 49.6% demonstrating the limitation of the acoustical modeling on the currently used training data.

Further work was finally performed in the last quarter to evaluate this baseline system using the development and test databases for the official 2005 evaluation on this EPPS English database, resulting in a WER of 47.9% for the development and 50% for the test database.

This is not the primary role and objective of Sony in this consortium to optimize an existing system based on MFCC front-end on the task chosen by the consortium, but more to concentrate its research activities in the area of adaptation to home and office environments and to check the viability of approaches related to auditory and binaural modeling for LVCSR applications. Still, to achieve better results, further near future activities in this area are planned, like training new acoustic models using also the EPPS training database, to possibly obtain an improvement of the current baseline system and get a further reference system more assimilable to a matched condition environment for this task.

7 Evaluation

This section provides an overview of the TCStar first year ASR evaluation held in March 2005. First an overview of the evaluation tasks and conditions are given, followed by a summary of the evaluation results. More detailed results can be found on the TCStar website in the slides from the evaluation workshop held in Trento in April.

7.1 Summary of ASR Evaluation Tasks and Conditions

The March 2005 evaluation assessed speech recognition performance in two tasks. The following language/task combinations were evaluated: European Parliament Plenary Speeches (EPPS) for European English and Spanish; and Broadcast News (BN) from VOA for Mandarin Chinese. Three training conditions were proposed: *restricted* (only TC-Star data can be used), *public* (any publicly available data can be used), and *open* (any data can be used). Partners opting for the open condition were encouraged to also submit a restricted system. As required by WP1, manual segmentations were used for the EPPS data. For BN automatic segmentations were used. The evaluation measure is the word error rate (WER) for English and Spanish, and the character error rate (CER) for Mandarin. The EPPS training data and the development and test data are summarized in Table 7. The complete evaluation plan is given in Section 7.3.

7.2 Summary of ASR Evaluation Results

All WP2 partners participated in the March 2005 evaluation, submitting at least one system to one of the 3 evaluation conditions (English EPPS, Spanish EPPS, and Mandarin BN), with a total of over 30 submissions including the contrastive conditions. The word error rates for EPPS English and Spanish are given in Tables 8 and 9 respectively. The best word error rate for a single system on the English EPPS was 10.6% which represents a very significant improvement over the best baseline system error rate of 32% . Combining the results of 5 systems (LIMSI,IBM,IRST,UKA,RWTH) using the Rover voting scheme [15] results in a word error rate of 9.5%. Comparable results were obtained for EPPS Spanish

Restricted EPPS training data (3 May - 14 Oct. 2004)		
English	40h	32M words
Spanish	40h	36M words (+ 43M from Spanish Parliament)

Development/Evaluation data		
English	Dev	3.7h Oct04 (2nd half)
	Eval	3.5h Nov04
Spanish	Dev	3.8h Oct04 (2nd half)
	Eval	3.8h Nov04
Mandarin	Dev	3.2h, 6 VOA shows, 01-11 Dec98
	Eval	3.2h, 6 VOA shows, 14-22 Dec98

Table 7: Summary of EPPS training data, and dev/eval data for the 3 languages.

Systems	*Open/Public*	*Restricted*
LIMSI	10.6	11.2
IBM	11.6	12.3
IRST	-	13.4
UKA	14.0	-
RWTH	-	14.1
NOKIA	24.6	-
SONY	50.0	-
Rover(LIMSI,IBM,IRST,UKA,RWTH): **9.5**		

Table 8: Summary of EPPS English results (21 submissions)

(11.5%), for which a Rover combination of the 4 systems (LIMSI,IBM,RWTH,IRST) reduces the word error rate to 10.1%.

Since only two sites (LIMSI and UKA) worked on BN Mandarin, they decided to submit a single joint submission making use of cross site adaptation since Rover is not effective with only two systems. The joint submission used a first hypothesis generated by UKA to adapt the LIMSI acoustic models, prior to a full decode with the LIMSI system. The results on the dev and eval data are given in Table 10. The results on the evaluation data are approximative since the final reference transcripts are not yet available.

Some of the characteristics of the 5 best performing systems for the EPPS English task under the restricted training conditions are shown in Table 11. More information about the different systems can be found in the descriptions given in Sections 8.1 to 8.7. It can be seen that despite the wide variety in the combination of parametrization, training methods, and decoding strategies, all systems perform reasonably well and combine well with Rover.

Systems	*WER (restricted condition)*
LIMSI	11.5
IBM	12.2
RWTH	12.7
IRST	13.7
Rover(LIMSI,IBM,IRST,RWTH): **10.1**	

Table 9: Summary of EPPS Spanish results (8 submissions)

System	*Dev data*	*Eval data*
LIMSI/UKA	10.9	<10.0

Table 10: of BN Mandarin results (CER)

	IBM	IRST	RWTH	UKA	LIMSI
Front-End	PLP	MFCC	MFCC	MFCC+MVDR	PLP
window	25ms	20ms	25ms	16ms	30ms
#features	13	13	16+voicing	13	13
LDA	9x13	no	9x17	15x13	no
frame size	40	39	33	42	39
VTLN	yes	no	yes	yes/no	no
MLLT/STC	yes	no	no	yes	yes
#Gaussians	106k	90k	450k	300k	350k
SAT	yes	yes	no	no	yes
MMI/fMPE	MPE+fMPE	no	no	no	MMI
Clustering	GMM	BIC	GMM	GMM	GMM
Adaptation	C+MLLR	MLLR	MLLR	C+MLLR	C+MLLR
Vocab	56k	64k	58k	55k	44k
#phones	42	45	44	45	48
LM	4g	3g	3g	4g	NN-4g
Decoding	6p	2p	2p	6p	2p
ROVER	no	no	no	4way-CN	no

Table 11: Characteristics of the 5 best performing systems for the EPPS English task under the restricted training conditions.

7.3 TC-STAR 2005 ASR evaluation plan

This section defines the ASR tasks, the evaluation conditions, the development and test corpora, the scoring procedures, and the agenda for the 2005 TC-STAR ASR evaluation. The 2005 ASR evaluation is done in conjunction with the 2005 TC-STAR SLT evaluation and takes into account the SLT requirements. The 2005 evaluation plan supports ASR evaluation for three languages (English, Spanish, Mandarin) and two domains (EPPS and broadcast news).

To take part in this evaluation, participants only need to take part in one official condition but it is expected that some participants will submit system outputs for more than one task and/or various contrastive conditions for a given task. Participants are encouraged to submit as many contrastive results as they like to highlight differences in the implemented methods.

Word error rate (or character error rate for Mandarin) will be the primary measure to compare systems, but the normalized cross entropy (for the confidence scores) and the real-time factor will also be tabulated.

Since the ASR/SLT coupling is a key issue for TC-STAR, in addition to the single best hypothesis participants are encouraged to provide N-best outputs and/or word graphs. Participants planning to provide alternate system outputs should distribute sample outputs as soon as possible. They must also distribute full development set system outputs before the specified date in the evaluation schedule. ASR and SLT participants are also encouraged to team up in order to evaluate alternate ASR/SLT interfaces.

7.3.1 The ASR tasks and languages

The TC-STAR'05 evaluation will include data sets in English, Spanish and Mandarin. Two tasks are supported for this evaluation, namely the European Parliament Plenary Session task (EPPS) for English and Spanish, and the Broadcast News task (BN) for Mandarin. BN English and Spanish are currently not included in this plan but may be added in future evaluations.

The following three test conditions will be used for the TC-STAR'05 evaluation:

- European Parliament Plenary Session in English (EPPS_ENG) with a sentence-level manual segmentation provided by RWTH/ELDA, 3-4h test set from November 2004, it should include preferably original speeches (i.e. not the translator's speech).
- European Parliament Plenary Session in Spanish (EPPS_SP) with manual segmentation provided by UPC/ELDA, 3-4h test set from November 2004 should include preferably the original speeches.
- Broadcast news Mandarin (BN_MAN) with automatic segmentation, about 3h of VOA taken from the LDC TDT3 corpus (Dec. 1998).

Participants can build systems for any processing speed, so there are no specific speed categories. However participants must report the total time (wall to wall elapsed time) needed to process the data for each submitted system. The real-time factor will be included in the tabulated results along with the word/character error rates.

In addition to these three supported conditions, participants are encouraged to report results on BN English data on NIST evaluation test sets (full set only). Scoring of these results is the sole responsibility of each site. These results will be included in the WP2 progress reports.

7.3.2 Processing rules

The evaluated systems must be fully automatic requiring no manual interventions that have an impact on the system output. Systems will be provided with audio files (16kHz sampling rate, 16 bit samples, mono) using a standard format. Unless specified in this document, no other information about the test data can be used. Supervised model adaptation on the test data is not allowed.

Data material (audio, texts, etc.) generated after the training cut-off date (or during the blackout period) cannot be used for system training or development (see evaluation schedule) with the exception of the official development data. For broadcast news, the show identity and the broadcast date are allowable side information that systems may use.

For the English and Spanish EPPS task, all training material must predate October 16th, 2004 with the exception of the development data. For the Mandarin broadcast news task, developers can not use any training data from the month of December 1998 (from which the dev and test data will be taken).

Manual segmentations will be provided for the English and Spanish EPPS tasks in form of a NIST PEM file. These segments will be used to filter the system output (CTM file) for scoring and to generate a sentence based format for the translation systems. This file can be used as prior information by the ASR systems. In particular, sites planning to produce word lattices for the SLT systems must use these segments and provide one lattice per segment. The segments can be processed in any order. A NIST UEM file will also be provided for contrast systems not using this manual segmentation.

Participants can use data collected within TC-STAR and any publicly available data (essentially from LDC and ELDA) predating the training cut-off date (see schedule below) for system development. They can also use any data they may find to be useful under the condition that this data predates the cut-off date (or doesn't fall in the blackout period for BN in Mandarin) and that they also submit results for a system using only TC-STAR and/or public data. In all cases, the training data should be fully and unambiguously documented in the system description.

It follows that for the EPPS tasks we can distinguish three training conditions:

1. a restricted training condition for which systems must be trained only on data collected within the TC-STAR project and listed in the next section.

2. a public data condition for which systems can be trained on any publicly available data

3. an open condition where the only constraint concerns the cutoff date of the training data.

Participants must submit a complete result for at least one of the three tasks (EPPS English, EPPS Spanish, and BN Mandarin). They can submit as many contrastive results as they like. For the EPPS tasks, participants are strongly encouraged to submit systems trained on the restricted set of corpora.

7.3.3 Data for the 2005 evaluation

The development and test data will be carefully transcribed and formatted for use with the NIST scoring tools. In addition the EPPS data will be manually segmented at the sentence level. This manual segmentation can be used by the ASR systems to decode the EPPS data. Automatic segmentation will be provided by one or more sites for both the development and evaluation BN sets.

The following table summarizes the main attributes of the development and evaluation data for the TC-STAR 2005 evaluation.

Language	*DataType*	*Domain*	*Epoch*	*Amount*	*Delivery*
English	Dev	EPPS	25-28 Oct 04	3-4h	Dec 2004
Spanish	Dev	EPPS	25-28 Oct 04	3-4h	Dec 2004
Mandarin	Dev	BN	1-15 Dec 98	3h	1 Mar 05
English	Eval	EPPS	15-18 Nov 04	3-4h	1 Mar 05
Spanish	Eval	EPPS	15-18 Nov 04	3-4h	1 Mar 05
Mandarin	Eval	BN	16-30 Dec 98	3h	1 Mar 05

The EPPS development and test data will preferably be comprised of original speeches and not the translated speech from interpreters. The same data will also be used to evaluate the translation systems.

The following table contains a non-exhaustive list of corpora that participants may want to use to train their acoustic models:

- English TC-STAR EPPS, about 40h of transcribed data, (May 3 - Oct. 14, 2004)
- English LDC 1995 (CSR-IV Hub 4 Marketplace LDC96S31), 1996, 1997, official NIST Hub4 training sets, LDC97S44 and LDC98S71, USC Marketplace Broadcast News Speech (LDC99S82)
- English LDC TDT2 and TDT3 data with closed-captions, about 2000h, LDC99S84 and LDC2001S94
- English LDC Switchboard 1, 2-I, 2-II, 2-III, LDC97S62, LDC98S75, LDC99S79
- English LDC Callhome, LDC97S42, LDC2004S05, LDC2004S09
- English LDC Meeting corpora, ICSI LDC2004S02, ISL LDC2004S05, NIST LDC2004S09
- Spanish TC-STAR EPPS, about 40h of transcribed data
- Spanish LDC 1997, BN speech (Hub4-NE), LDC98S74
- Spanish LDC CallHome, LDC96S35
- Mandarin LDC 1997, BN speech (Hub4-NE), about 30h of transcribed data, LDC98S73
- Mandarin TDT2 and TDT3 data with quick transcriptions, LDC2001S93 and LDC2001S95

The following corpora can be used for language model development:

- All transcriptions (detailed, quick or CC) of the above mentioned audio corpora
- English EPPS final transcriptions, about 36M words (from parallel texts)
- English LDC NAB text corpus
- English LDC Gigaword (over 1 billion words)
- Spanish EPPS final transcriptions, about 36M words (from parallel texts)
- Mandarin LDC news text, about 250 million GB-encoded text characters
- Mandarin LDC Gigaword, about 1.1 billion words

For more complete listings of possible corpora, participants are referred to the LDC and ELRA catalogs.

For the EPPS tasks, participants are encouraged to submit systems trained only a restricted set of training corpora including:

- English TC-STAR EPPS, about 40h of transcribed data, (May 3 - Oct.14, 2004)
- English EPPS final transcriptions, about 36M words (from parallel texts)
- Spanish TC-STAR EPPS, about 40h of transcribed data
- Spanish EPPS final transcriptions, about 36M words (from parallel texts)
- Spanish parliament transcription from 1979 to October 15h 2004

The English and Spanish EPPS data collected within TC-STAR is available from RWTH. The transcriptions of the training data are available at http://www-i6.informatik.rwth-aachen.de/∼tcstar, and the transcription of the development data at http://www.elda.org/en/proj/tcstar-wp4/tcs-asr-data.htm.

7.3.4 System outputs

For each input audio file the ASR hypothesis are to be formatted in a NIST CTM file, i.e. the concatenation of time mark records separated with a newline (Unix text file) for each hypothesized word. For English and Spanish, the word tokens may contain only upper or lowercase alphabetic characters, hyphens and apostrophes. Since the NIST scoring software does not fully support UTF-8 we will use ISO-Latin characters for scoring. (Participants can generate CTM with UTF-8 characters but they will be to be converted to ISO-Latin for scoring.) GB encoding will be used for Mandarin.

Systems are supposed to use a single standardized spelling for each language. However some filtering and mapping will be applied to the system output prior to scoring to take into account acceptable common alternate forms. Both American English and British English spelling will be allowed.

In addition to reference dictionaries, the Internet may be searched to find the most common form of a word (usually a proper name). If no form is dominant then more than one form will be allowed (cf. GLM table in scoring section).

The system may use an optional hyphen to indicate the missing (unspoken) part of a word token. Filled pause makers and non-speech markers should not be included in the system output for scoring, however participants are encouraged to provide this information which may be useful for the SLT systems.

7.3.5 Scoring

A NIST Segment Time Marked (STM) reference file will be provided for the development and test set (after the system submission). Following the NIST practice, contractions will be expanded in the STM file: i.e. the annotator will choose the single most likely expansion for each contraction. Non-scoreable regions (such as untranscribed areas and overlapping speech) will be explicitly tagged in the STM file for exclusion from scoring.

Prior to scoring, a global mapping will be performed on both the reference and system outputs via a set of rules specified in a global map (GLM) file. The GLM rules expand contractions and split compound words in the system output to all possible expanded forms.

Following NIST practice, optionally deletable tokens in the STM file may be omitted by the speech recognizer. These tokens contribute to the count of reference tokens whether or not the system outputs them.

The CTM and STM files will be aligned (using dynamic programming) to minimize the word/character error rate. Scoring will done using the NIST speech recognition scoring toolkit available at http://www.nist.gov/speech/tools. Specific filtering tables and GLM files will be developed for TC-STAR (one set per language). Scoring will be case insensitive. A hyphen within a token will be treated as a token separator.

7.3.6 Enriched system output

Participants should also provide (this is not required) a confidence score for each hypothesized word in the CTM file. This confidence score represents the system's estimate of the probability that the output token is correct. The correctness of the confidence scores will be evaluated using the normalized cross entropy score as reported by the NIST sclite tool. The confidence error rate (CER) will also be computed and reported.

In addition to the CTM file, participants are encouraged to provide n-best hypotheses and/or word graphs to be used by the SLT systems. Sites who plan to provide N-best hypotheses or word lattices must also provide these outputs for the development set in order to solve issues related to file formats, vocabulary compatibility, segmentation, and decoding parameters. As it is expected that these issues will not be solved for all pairs of providers-users, ASR and SLT participants should team to solve these issues. Interfaces between ASR and SLT are not limited to N-best hypotheses and word lattices, so participants may consider alternative solutions for within site and cross-site site integration.

7.3.7 Processing time

Even though processing speed is not a major issue for the 2005 evaluation, participants must provide information about the processing time and the resources (memory, CPU type, clock frequency) used to run the ASR systems. This should be included in each system description. Participants should report elapsed time (i.e. not real-time factor) for all steps if possible. ELDA will compute the processing speed as the ratio of the processing time to the official duration of the recorded audio data. The processing time is the total amount of elasped time used to process the data on a single CPU, including I/O and all operations performed after first accessing the test data.

8 System descriptions

8.1 ITC-irst systems

For the first official evaluation, results were submitted by ITC-Irst for English and Spanish EPPS tasks.

ITC-Irst English EPPS. The development of ITC-Irst English EPPS systems, in the *restricted* condition, included the training of a conventional system on the TCSTAR EPPS data, and the training of a system based on CMLSN normalization. Details on each submitted systems can be found in the following specific sections.

The development of the systems also included a comparison between two different lexica. The first one is a HUB4 lexicon built from different sources, all based on American English pronunciation. The second is the BEEP lexicon, that is based on British English pronunciation, and therefore should in principle be better suited for the EPPS task. Experiments showed that the latter lexicon actually performed slightly worse ($\approx$2.5% relative performance decrease) than the former. There are three possible explanations for this observation. Firstly, rules quickly adapted from the HUB4 phone set were used to build the phonetic decision tree with the BEEP phone-set, which is sub-optimal. Secondly, the overall quality of the transcriptions contained in the HUB4 lexicon may simply be better than for BEEP. Thirdly, it is possible that the European English is just a highly variable mixture of different accents of English which is no more similar to British English than to American English.

ITC-Irst Spanish EPPS. ITC-Irst Spanish recognizer basically adopts the same technology used for the English baseline. As this is the first ITC-Irst system developed for large vocabulary speech recognition in Spanish, it does not yet include adaptive training techniques. Most of the activity was related to language model and phonetic transcriptions. Text preprocessing includes numeric expansion, removal of every type of punctuation, identification of possible acronyms and proper nouns, removal of too long ($>$ 30 chars) words, various checks to identify and properly handle characters belonging to foreign languages.

Unlike the English system, in which capitalization was added in a post processing step, for Spanish a case sensitive language model was built. In order to provide capitalization, all text was normalized to lowercase, and the statistics were computed for each set of word sharing the same normalized form. Finally, the text was filtered replacing each word with the most frequent among its pool.

As there was no suitable pronunciation lexicon for Spanish available, the transcriptions were automatically generated using a set of grapheme-to-phoneme rules for Spanish. This tool can handle acronyms and multiple pronunciations; some rules were added to handle some common foreign patterns. In addition, a set of about 350 words – mostly high-frequency foreign names, found in the training data – was manually transcribed.

8.1.1 irst_eval05_epps_en_restricted_primary

Acoustic front-end. During training on the EPPS data, manual segmentation is exploited for speech segment detection and speech segment clustering is performed on a file-by-file basis by exploiting gender labels. During test, manual segmentation is exploited for speech segment detection and automatic clustering is performed on a file-by-file basis.

Acoustic features of the ITC-irst speech recognition system include 13 Mel-frequency Cepstral Coefficients (MFCCs) and their first and second order time derivatives. The MFCCs are computed every 10ms using a Hamming window of 20ms length. The filter-bank contains 24 triangular overlapping filters which are centered at frequencies between 125 and 6750 Hz.

Two recognition passes are performed. For the first and the second recognition pass two different acoustic front-ends are used. This is necessary as a supervised acoustic normalization technique is used in the second recognition pass (see ”Recognition process”, below).

- For the first pass, Cluster-based Cepstral Mean and Variance Normalization (CMVN) ensures that for each cluster the static features have zero mean and unit variance.
- For the second pass, Segment-based Cepstral Mean Normalization (CMN) is applied to the static features, adjusting the mean of each static coefficient for each segment to zero. No variance normalization is employed in this case.

Acoustic model. The EPPS training data, released within the TC-STAR Project and consisting of about 40 hours of speech, are exploited for training.

Acoustic models for the first and the second decoding pass are trained differently.

For the first decoding pass, the acoustic models are state-tied, cross-word, gender-independent, bandwidth-independent triphone HMMs. A phonetic decision tree is used for tying states and defining the context-dependent allophones. These models have about 5000 tied states and about 91000 Gaussians with diagonal covariance matrices. A standard MLE acoustic training procedure is applied on the CMVN-transformed features.

Acoustic models for the second decoding pass are trained by means of the acoustic normalization procedure described in [54, 22]

a) a set of target models is trained on untransformed, mean-normalized feature vectors. The target models are tied-states triphone HMMs with a single Gaussian density for each state.

b) for each cluster in the training data, a CMLLR [16] transform is estimated w.r.t. the target models.

c) the CMLLR transforms are applied to the feature vectors. The resulting, transformed or normalized feature vectors are supposed to contain less speaker, channel, and environment variability.

d) a conventional ML training procedure is used to initialize and train the recognition models on the normalized features, including state tying and the definition of the context-dependent allophones.

This second set of models has about 5000 tied states and about 90000 Gaussians with diagonal covariance matrices.

Language Model. A trigram language model was trained on the training text data released within the TC-STAR project (English EPPS final transcriptions, consisting of about 36 million words) and then adapted to the manual transcriptions of the acoustic EPPS data released for training of the acoustic models (consisting of about 370000 words). Perplexity and out of vocabulary rate measured on the manual transcriptions of the EPPS development set are 105.9% and 0.6%, respectively.

Recognition Lexicon. The pronunciations in the lexicon are based on a set of 45 phones. The lexicon contains 64k words. It has been generated by merging different source lexica for American English (LIMSI ’93, Cmudict, Pronlex). In addition, there is a model for silence, six models for filler words and breath noises and one for modeling out of vocabulary words (used only during training). Transcriptions for few hundreds of words were generated manually.

Recognition process. A first decoding pass is carried out with the first set of acoustic models. The recognition result (submitted for evaluation with as contrast-2) is used as supervision for the acoustic data normalization procedure and for cluster based adaptation of acoustic models to be used for the second decoding pass. The second decoding pass is carried out using the acoustic models that have been trained with the CMLSN acoustic normalization procedure after normalizing the test data and performing cluster based MLLR adaptation [30]. For MLLR, two regression classes have been determined in a data-driven manner. Mean vectors are adapted using full transformation matrices, while diagonal transformation matrices are used to adapt the variances. Three MLLR iterations are performed. A description of the cross-word decoding algorithm can be found in [4].

Execution time. The total execution time, computed on a single Intel(R) Xeon(TM) 3.00GHz processor with 1024 KB cache and 4GB memory, for producing the output after the second decoding pass is (49135 + 70286) 119421 seconds. Hyperthreading was disabled.

8.1.2 irst_eval05_epps_en_restricted_contrast-1

The acoustic front-end, acoustic models, language model and lexicon are the same as for the primary submission. The difference is only in the recognition process.

Recognition process. Unlike the primary submission, in this case the second decoding step was performed directly with the CMLSN trained models on the normalized test data, without the MLLR adaptation.

Execution time. The total execution time, computed on a single Intel(R) Xeon(TM) 3.00GHz processor with 1024 KB cache and 4GB memory, for producing the output after the second decoding pass is (49135 + 66149) 115284 seconds. Hyperthreading was disabled.

8.1.3 irst_eval05_epps_en_restricted_contrast-2

The acoustic front-end and acoustic models are as described for the first step in the primary submission. The language model and lexicon are the same as for the primary submission.

Recognition process. A single decoding pass is performed and its output is submitted for evaluation.

Execution time. The total execution time, computed on a single Intel(R) Xeon(TM) 3.00GHz processor with 1024 KB cache and 4GB memory, for producing the output after a single decoding pass is 49135 seconds. Hyperthreading was disabled.

8.1.4 irst_eval05_epps_en_restricted_contrast-3

The acoustic front-end, acoustic models, language model and lexicon are the same as for the contrast-2. The only difference is that the recognition process in this case also includes unsupervised MLLR adaptation.

Recognition process. Two decoding passes are carried out, the output of the first decoding pass (submitted for evaluation as contrast-2) is used as a supervision for adaptation of the recognition models through cluster-based MLLR adaptation. For MLLR, two regression classes have been determined in a data-driven manner. Mean vectors are adapted using full transformation matrices, while diagonal transformation matrices are used to adapt the variances. Three MLLR iterations are performed.

Execution time. The total execution time, computed on a single Intel(R) Xeon(TM) 3.00GHz processor with 1024 KB cache and 4GB memory, for producing the output after two decoding passes is (49135 + 43949) 93084 seconds. Hyperthreading was disabled.

8.1.5 irst_eval05_epps_es_restricted_primary

Acoustic front-end. During training on the EPPS data, manual segmentation is exploited for speech segment detection and clustering is performed on a file by file basis by exploiting gender labels. During test, manual segmentation is exploited for speech segment detection and automatic clustering is performed on a file by file basis. The acoustic front-end of the ITC-irst speech recognition system combines 13 Mel-frequency Cepstral Coefficients (MFCCs) and their first and second order time derivatives into a 39-dimensional feature vector. The MFCCs are computed every 10ms using a Hamming window of 20ms length. The filter-bank contains 24 triangular overlapping filters which are centered at frequencies between 125 and 6750 Hz. Cluster-based Cepstral Mean and Variance Normalization (CMVN) ensures that for each cluster the static features have zero mean and unit variance.

Acoustic model. The EPPS training data (EPPS-TD), released within the TC-STAR Project and consisting of about 44 hours of speech, is exploited for training. The acoustic models are state-tied, cross-word, gender-independent, bandwidth-independent triphone HMMs. A phonetic decision tree is used for tying states and defining the context-dependent allophones. The system has about 4100 tied states and about 65000 Gaussians with diagonal covariance matrices. Initial word segmentation of the acoustic training data was performed using a previous version of the acoustic models, trained from scratch on EPPS-TD data only. A standard MLE acoustic training procedure is applied on the CMVN-transformed features.

Language model. First a trigram language model was trained on the training text data released within the TC-STAR project: Spanish EPPS final transcriptions (EPPS-ES, 33.5 million words) plus Spanish Parliament Transcription from 1979 to 2004, (SPT, about 12.0 million words); then LM adaptation was performed using the manual transcriptions of the acoustic EPPS training data (EPPS-TD, about 338,000 words). Perplexity and out of vocabulary rate measured on the manual transcriptions of the EPPS development set are 146.0 and 0.7%, respectively.

Recognition lexicon. The lexicon contains the 50,000 most frequent words found in EPPS-ES plus SPT, plus all the words found in EPPS-TD: the total lexicon size is 52,359 words. The phonetical transcriptions are based on a set of 31 phones, and were automatically generated using a set of grapheme-to-phoneme rules for Spanish. This tool can handle acronyms and multiple pronunciations; some rules were added to handle some common foreign patterns. In addition, a set of about 350 words -mostly high-frequency foreign names, found in the training data- was manually transcribed. In addition, there is a model for silence and three models for filler words and breath noises.

Recognition process. The output of the first decoding pass (submitted for evaluation as contrast-1) is used as a supervision for MLLR adaptation of the recognition models. For MLLR, two regression classes are used which have been determined in a data-driven manner. Mean vectors are adapted using full transformation matrices, while diagonal transformation matrices are used to adapt the variances. The CTM file submitted for evaluation contains the output of the second decoding pass of the recognizer after three steps of adaptation.

Execution time. The total execution time, computed on a single Intel(R) Xeon(TM) 3.00GHz processor with 1024 KB cache and 4GB memory, for producing the output after two decoding passes is (30656 + 27432) 58088 seconds. Hyperthreading was disabled.

8.1.6 irst_eval05_epps_es_restricted_contrast-1

Acoustic front-end, acoustic model, language model and lexicon are the same as for the primary submission. The only difference is in the recognition process.

Recognition process. A single decoding pass is performed.

Execution time. The total execution time, computed on a single Intel(R) Xeon(TM) 3.00GHz processor with 1024 KB cache and 4GB memory, for producing the output after a single decoding pass is 30656 seconds. Hyperthreading was disabled.

8.2 RWTH systems

The general training setup follows the RWTH Hub4en-LVCSR-System as described in [52]. The acoustic models for the initial alignment were taken from the Spanish LVCSR-System developed for the LC-STAR project at RWTH i6 [31].

In the RWTH recognition system standard MFCC features are used. The magnitude spectrum is estimated by applying the DFT to the preemphasised and windowed audio signal each 10ms. Next the magnitude spectrum is filtered with a filter bank consisting of 16 triangular filters positioned at equidistant points on the Mel frequency axis. The logarithms of the filter outputs are cepstrally decorrelated (discrete cosine transform), resulting in 16 dimensional vectors. The MFCCs are normalised. In addition a single voicing feature is added to each vector [64]. Nine temporally consecutive vectors are fed into an LDA to obtain 33 dimensional feature vectors. Two adaption techniques are applied, first VTLN and in an second run MLLR.

The words of the vocabulary of the RWTH recognition system are modeled by position-dependent triphones with across-word contexts [53]. The triphones are represented by Hidden Markov Models (HMMs). The non-silence HMMs use a standard three states left-to-right topology, where each of the states is duplicated resulting in a six states HMM model, whereas the silence HMM consists of a single HMM state. The emission probabilities assigned to the HMM states in turn are modeled by Gaussian mixture models, sharing a single, globally pooled diagonal covariance matrix. The transition probabilities are empirically estimated. HMM states are tied using a binary decision tree (CART). During training and recognition the Viterbi approximation on the state-level is used. The acoustic model consists of 4.500 tied states and approximately 500k densities.

The RWTH recognition system uses a single n-gram language model. From each data source an n-gram was estimated by interpolating lower order backoff n-gram-models, where the backoff weights were obtained by absolute discounting (Kneser-Ney). These models were combined into a single n-gram model using linear interpolation. For estimation and interpolation the SRI language modeling toolkit was used [57].

The RWTH pronunciation lexicon for European Spanish is an augmented version of the LC-STAR Spanish Lexicon assembled in the LC-STAR project [31]. The RWTH pronunciation lexicon for European English is an augmented version of the BEEP lexicon [41]. For both lexica, missing words were phonetically transcribed using a grapheme-to-phoneme model trained on the corresponding lexicon [8, 23]. A phoneme set of size 38 was derived from the LC-STAR lexicon, and a phoneme set of size 44 was derived from the Beep lexicon. In addition a silence symbol, a symbol describing filled pauses, and five symbols describing different kinds of noises were added. Finally the lexicon for English consisted of 58.059 words, and the lexicon for Spanish consisted of 60.020 words.

The RWTH speech recognition system is gender independent and uses a beam search strategy with two pre-pruning steps. Step one is an acoustic look-ahead on a monophone base and step two a language model look-ahead using a bigram language model [52]. On an AMD Opteron with 2200Mhz and 4GB RAM the RWTH speech recognizer obtains a real-time factor of about 30 for English and of about 12 for Spanish.

8.3 LIMSI systems

The LIMSI TCSTAR EPPS English and Spanish systems, and the BN Mandarin system (a joint submission with UKA) use the same basic modeling and decoding strategy as used in the DARPA RT03 HUB-4NE 10x evaluation, with models (lexicon, acoustic models, language models) trained for the respective tasks/languages. The systems run in about 6xRT.

8.3.1 General system description

The LIMSI segmentation and clustering, used only for the BN Mandarin system, is based on an audio stream mixture model [20, 19]. First, the non-speech segments are detected and rejected using GMMs representing speech, speech over music, noisy speech, pure-music and other background conditions. An iterative maximum likelihood segmentation/clustering procedure is then applied to the speech segments. The result of the procedure is a sequence of non-overlapping segments with their associated segment cluster labels. Each segment cluster is assumed to represent one speaker in a particular acoustic environment and is modeled by a GMM. The objective function is the GMM log-likelihood penalized by the number of segments and the number of clusters, appropriately weighted. Four sets of GMMs are then used to identify telephone segments and the speaker gender. Segments longer than 30s are chopped into smaller pieces by locating the most probable pause within 15s to 30s from the previous cut.

The LIMSI speech recognizer [19] uses 39 cepstral parameters derived from a Mel frequency spectrum estimated on the 0-8kHz band (or 0-3.5kHz for telephone data) every 10ms. For each 30ms frame the Mel scale power spectrum is computed, and the cubic root taken followed by an inverse Fourier transform. Then LPC-based cepstrum coefficients are computed. The cepstral coefficients are normalized on a segment-cluster basis using cepstral mean removal and variance normalization. Thus each cepstral coefficient for each cluster has a zero mean and unity variance. The 39-component acoustic feature vector consists of 12 cepstrum coefficients and the log energy, along with the first and second order derivatives.

Each phone model is a tied-state left-to-right CD-HMM with Gaussian mixtures. The triphone-based context-dependent phone models are word-independent but position-dependent. The tied states are obtained by means of a decision tree.

Word recognition is performed in two passes, where each decoding pass generates a word lattice which is expanded with a 4-gram LM. The posterior probabilities of the lattice edges are estimated using the forward-backward algorithm. The 4-gram lattices are converted to a confusion network with posterior probabilities by iteratively merging lattice vertices and splitting lattices edges until a linear graph is obtained. This procedure gives comparable results to the edge clustering algorithm proposed in [32]. The words with the highest posterior in each confusion set are hypothesized.

Pass 1: Initial Hypothesis Generation - This step generates initial hypotheses which are then used for speaker-based acoustic model adaptation. This is done via one pass (about 1xRT) cross-word trigram decoding with gender-independent sets of position-dependent triphones and a trigram language model. The trigram lattices are rescored with a 4-gram language model.

Pass 2: Adapted decode - Unsupervised acoustic model adaptation of speaker-independent models is performed for each speaker using the CMLLR and MLLR techniques [30] with only two regression class. The lattice is generated for each segment using a bigram LM and a larger set of position-dependent triphones (32 Gaussians per state). As for pass 1, the lattices are rescored with a 4-gram language model. The execution time for this pass is about 5xRT. A neural network 4-gram language model [47] is used in this last pass during lattice rescoring for EPPS English.

8.3.2 Acoustic model training

The acoustic models are tied-state left-to-right CD-HMM with Gaussian mixtures. The number and type of context modeled depends on the language and the decoding pass. The characteristics of the acoustic models are summarized in Table 13.

Language	English	Spanish	Mandarin
Data type	EPPS	EPPS	BN
Audio data	40h	40h	27h + 170h
Transcripts	manual	manual	manual + light
P1 (#ctx/#states)	3.9k/4k	1.5k/1.7k	(UKA)
P2 (#ctx/#states)	14k/11k	5.3k/5k	24k/11.5k

Table 12: Summary of acoustic models.

English: The acoustic models were trained on about 40 hours of audio training data from the EPPS English distributed by RWTH. The acoustic models MLLT-SAT trained, tied-state position-dependent triphones, where the state-tying is obtained using a divisive decision tree based clustering algorithm.

For the restricted condition, the first pass models cover 3940 triphones with 4k tied states (32 Gaussians per state). For the public condition, the first pass models are BN English models MAP adapted with the EPPS data (37k contexts, 11k states). For the public condition, the first pass models are BN English models MAP adapted with the EPPS data (37k contexts, 11k states). The second pass models cover 14k position-dependent triphones with 11k states and 32 Gaussians per state.

Spanish: The acoustic models were trained on about 40 hours of audio training data from the EPPS Spanish distributed by UPC.

The acoustic models MLLT trained, tied-state position-dependent triphones, where the state-tying is obtained using a divisive decision tree based clustering algorithm.

For the restricted condition, the first pass models are ML trained, and cover 1523 word position dependent triphones with 1700 tied states (32 Gaussians per state). The second pass models are ML MLLT trained and cover 5275 word position dependent triphones with 5k tied states (32 Gaussians per state).

Mandarin: The acoustic models were trained on about 27 hours of Hub4-Mandarin training data (from LDC) and about 170 hours of data from the TDT2, TDT3 and TDT4 corpora. Most of these data (about 140 hours are from VOA). Since time-aligned transcripts are not available, the TDT data from the Mainland China sources (CNR, CTV and VOA) were transcribed with our recognizer using our 2003 BN eval system acoustic models [9] and source/show-specific language models estimated on the TDT closed captions for each source/show. Wide-band and band limited models were trained by pooling the Hub4 Mandarin data and the TDT data.

The acoustic models are position-dependent triphones with tied states, obtained using a divisive decision tree based clustering algorithm. Two sets of gender-dependent acoustic models were built using both MAP adaptation [21] of SI seed models for each of wideband and telephone band speech.

The English Hub4 training data was used to build the Gaussian mixture models for gender identification, and music and telephone segment detection. About 2 hours of pure music portions of the acoustic training data were used to estimate the music GMM.

8.3.3 Language model training

All systems use n-gram language models obtained by interpolation [62] of component backoff n-gram language models trained subsets of the training texts. For English and Spanish the language models are case sensitive and use the iso-latin-1 encoding. The characteristics of the language models are summarized in Table 13.

Language	English	Spanish	Mandarin
Data type	EPPS	EPPS	BN
#words	44k	65k	54k
%OOV	0.4%	0.6%	$\sim$0
Transcripts	350k	340k	460k chars
Text	32M	33M	600M
4g ppx	77	80	250

Table 13: Summary of language models.

English: For the English EPPS system component n-gram language models trained on the following 4 sources:

1. Audio transcriptions: 353k words (cut-off 0-0-0)

2. Parliamentary texts: 32M words (cut-off 0-0-1)
3. CNN caption texts [01/2000 - 15/10/2004]: 153M words (cut-off 1-1-2)
4. Broadcast news transcriptions: 293M words (cut-off 1-1-2)

The mixture weights were chosen to minimize the perplexity of the optimized development data. The 4-gram perplexity on the dev05en data is about 77. The LM contains about 2.4M bigrams, 10.2M trigrams and 3.6M fourgrams. The neural network 4-gram language model is estimated using only the restricted data (i.e., the first 2 sources listed above).

The 44k word list was selected from the same text sources so as to minimize the OOV rate on the dev05en data, and has an OOV rate of about 0.4%. The word list contains all words in manual transcriptions of the EPPS training data and words in the Parliamentary texts occurring at least twice.

For the restricted condition, only the first two text sources (audio transcripts and Parliamentary texts) are used for LM training. The 4-gram perplexity on the dev05en data with the restricted training texts is about 80.

Spanish: N-gram language models were obtained by interpolation of backoff n-gram language models trained on the following two sources:

1. Audio transcriptions: 340k words (cut-off 0-0-0)
2. Parliamentary texts: 33M words (cut-off 0-0-1)

Component LMs were trained on the two text sources and the the mixture weight was chosen to minimize the perplexity of the optimized development data. The 4-gram perplexity on the dev05en data is about 80.

The 65k word list was selected from the same text sources so as to minimize the OOV rate on the dev05es data, and has an OOV rate of about 0.6%. The word list contains all words in manual transcriptions of the EPPS training data and words in the Parliamentary texts occurring at least twice.

Mandarin: For the Mandarin system, component n-gram language models trained on the following sources available from LDC:

1. Hub4 Mandarin audio transcripts
2. China radio international 1994-1996 (87M characters)
3. People Daily newspaper (89.2M characters)
4. Xinhua news (9.9M characters)
5. TDT2,3,4 XIN (12M characters)
6. TDT2,3,4 ZBN (12M characters)
7. TDT2,3,4 VOA (2.3M characters)
8. LDC gigaword Mainland texts (367M characters)

The period of Dec98 is excluded from all text sources. Different component LMs are trained on the text sources mentioned above, with the mixture weights optimized using the transcriptions of dev03 data. The interpolation coefficients were chosen in order to minimize the perplexity a set of dev shows for which reference transcriptions were obtained by correcting the output of aligning the TDT3 closed captions with the recognizer hypotheses. The transcription correction of the six shows was shared by LIMSI and CMU (19981201_0700_0800_VOA_MAN 19981202_0900_1000_VOA_MAN 19981203_0700_0800_VOA_MAN 19981204_0900_1000_VOA_MAN 19981208_0700_0800_VOA_MAN 19981211_0900_1000_VOA_MAN).

The 54k word list was selected from the same text sources so as to minimize the OOV rate on the dev05 data. The word list includes all (about 7000) characters (i.e., there are essentially no OOV characters).

8.3.4 Recognition lexicon description

Case sensitive recognition lexica are used for English and Spanish, which also include the most frequent acronyms found in the training texts. The Mandarin lexicon contains the most frequent words in the training texts completed by >7k characters. Characteristics of three lexica are summarized in Table 14.

Language	English	Spanish	Mandarin
#words	44k	65k	54k
#phones	48	27	61
#fillers	3	3	4
#prons	55k	94k	59k

Table 14: Summary of lexica.

English: Pronunciations in the English lexicon are based on a 48 phone set (3 of them are used for silence, filler words, and breath noises). A pronunciation graph is associated with each word so as to allow for alternate pronunciations, including optional phones. The 44k vocabulary contains 43903 words including 55342 phone transcriptions. Frequent inflected forms have been verified to provide more systematic pronunciations. As done in the past, compound words for about 300 frequent word sequences subject to reduced pronunciations were included in the lexicon as well as the representation of frequent acronyms as words.

Spanish: Pronunciations are based on a 27 phone set (3 of them are used for silence, filler words, and breath noises). Pronunciations for the 65k word vocabulary are generated via letter to sound conversion rules, with a limited set of automatically derived pronunciation variants.

Mandarin: Pronunciations in the Mandarin lexicon make use of 61 phones (4 of them are reserved for silence, filler words, and breath noises). The 5 tones for the vowels are collapsed into 3 tones for each vowel (rising, flat and falling) A pronunciation graph is associated with each word so as to allow for alternate pronunciations, including optional phones. The 54k vocabulary contains 54025 words with 55377 phone transcriptions.

8.4 UKA systems

In the first TC-STAR evaluation UKA participated in the public English EPPS task and the Chinese Broadcast News task.

8.4.1 English EPPS task

For the English EPPS task UKA submitted two systems, one primary and one contrast system.

For the primary submission two different kinds of preprocessing were being used. One front-end was based on Mel-frequency scaled cepstral coefficients, using an FFT to calculate the power spectrum from which the cepstral coefficients were being calculated. For the FFT a hamming window with a window length of 16ms was applied using a window-shift of 10ms.

The second front-end utilized the warped minimum variance distortionless response (MVDR) which is a time domain technique to estimate an all-pole model using a warped short time frequency axis such as the Mel scale. The use of the MVDR eliminates the overemphasis of harmonic peaks typically seen in medium and high pitched voiced speech when spectral estimation is based on linear prediction (LP). Moreover, warping the frequency axis prior to MVDR spectral estimation ensures more parameters in the spectral model are allocated to the low, as opposed to high, frequency regions of the spectrum, thereby mimicking the human auditory system. For the short-term analysis also a 16ms hamming window was applied using a 10ms shift.

The acoustic models were trained on 48.7h of English EPPS data provided by RWTH Aachen within the TC-STAR project. Using the above data, an acoustic model using 300k Gaussians with diagonal covariances organized in 24k distributions over 6k codebooks in a 42-dimensional feature space based on MFCCs after LDA with utterance-based cepstral mean subtraction was trained. All systems made use of a global STC transformation matrix after LDA. The cluster tree, LDA Matrix and global STC transformation matrix came from the ISL Meeting Task system.

Also a as a variation one system making use of the flexible cluster tree allowing to share training data across polyphones with different center phones was trained, organizing the models in 6k distributions over 6k codebooks instead. The cluster tree was trained on the EPPS data only.

Further one system using a conventional tree, also organizing 300k Gaussians in 24k distributions and 6k codebooks as described above, was freshly trained on the EPPS data only. The LDA and STC matrix however remained unchanged.
The following systems were used during decoding:

mfcc-novtln: mfcc based frontend; merge-and-split training followed by 2 iterations of Viterbi training, no vocal tract length normalization (VTLN)

mfcc-vtln: like mfcc-novtln plus VTLN

mvdr: like mvdr based front-end; merge-and-split training followed by 2 iterations of viterbi training; with VTLN during training

mfcc-vtln.newTree: like mfcc-vtln with the standard cluster tree trained on EPPS data only.

mfcc-vtln.flexTree: like mfcc-vtln with the flexible cluster tree trained on EPPS data.

For the first recognition pass we used an n-gram language model that was interpolated between a 4-gram language model trained on 30.5 million processed words from the English side of the ”EPPS Spanish-English Parallel Text Corpus derived from the Final Text Editions” and made available by RWTH Aachen and a 3-gram model trained on 340k words of manual transcriptions of the acoustic training material from EPPS May-Sep 2004. The audio transcriptions of EPPS Oct 13-14, 2004 was used as held-out data for tuning the context-dependent LM interpolation weights. The resultant LM had the perplexity of 97.3 on the development test set transcriptions of EPPS Oct 26-28, 2004.

For all subsequent passes a 3-gram language model trained on English BN data was interpolated with the two language models mentioned above resulting into the perplexity of 95.3.

The vocabulary of the recognition system consists of 55k words. This was selected by including all the words occurring at least once in the EPPS proceedings corpus and all the words occurring in the audio transcripts of the development material. The out-of-vocabulary rate on the development test set was 0.72%. In case the words did not already exist in the pronunciation dictionary of the Meeting Task System their pronunciation was generated with the help of the Festival speech synthesis tools. The pronunciation for words containing non_ASCII characters were generated by replacing those characters with their closest ASCII character transliteration before generating the pronunciation with Festival.

The recognition process consisted of 6 passes. During the adaptation process warping factors for the VTLN are estimated and the acoustic model is adapted using maximum-likelihood linear regression (MLLR) and feature-space constrained MLLR. For pass i) the window shift of the preprocessing is set to 10ms as in the training, for all other passes to 8ms.

i) Unadapted pass: Decoding with the mfcc-novtln system

iia) Adaptation of mfcc-vtln.newTree on i) and decoding

iib) Adaptation of mfcc-vtln.flexTree on i) and decoding

iic) Adaptation of mvdr on i) and decoding

iii) Adaptation of mfcc-vtln on iia) and decoding

iv) Confusion network combination of iia), iib), iic), and iii)

The execution time of the different passes is tabulated below.

i)	66203s
iia)	8107s + 30020s = 38127s
iib)	10138s + 34621s = 44759s
iic)	12056s + 38213s = 50269s
iii)	8484s + 35690s = 44174
iv)	1407s

Thus the total execution time equals 244939 seconds.

For the contrast submission the recognition process was shortened and consisted of only 4 passes:

i) Unadapted pass: Decoding with the mfcc-novtln system

ii) Adaptation of mfcc-vtln.flexTree on i) and decoding

iii) Adaptation of mfcc-vtln on ii) and decoding

iv) Confusion network generation from iii)

The total execution time for the contrast condition is 160994 seconds.

8.4.2 Chinese Broadcast News task

The ISL Mandarin Broadcast News evaluation system uses the JANUS speech recognition toolkit.

The frontend processing is based on 13 MFCC features using a context window of 15 frames. Cepstral mean and variance compensation for each clusters was followed by an LDA transform, giving the final feature vector of 42 components. Vocal tract length normalization was performed on a cluster basis.

Two sets of gender-independent acoustic models were applied: one using an initial-final (IF) lexicon and another using a phone-level lexicon. The IF system has 3000 clustered triphone states and a total of 168k Gaussians; the phone system has 3000 tied septaphone states with a total of 169k Gaussians. Tonal information was used in decision trees such that a single tree was used for all tonal variants of the same phone.

Maximum likelihood training was used for both sets of models. The mixtures were grown incrementally over several iterations. A single global semi-tied covariances (STC) are employed. The acoustic models were trained in a cluster adaptive way, making use of cluster based feature space transforms (FSA-SAT). Speaker adaptation during testing was carried out on the features (FSA), means (MLLR).

The partition strategy consists of four components: speech/non-speech segmentation, music detection, foreign language detection, and speaker clustering. It proceeds in the following steps:

1) initial segmentation using energy-based speech/non-speech detection (CMU segmenter, CMUseg_0.5 package);
2) Gaussian mixture model based music/non-music classification: Music segments were subsequently discarded;
3) Language identification: We use a phonetic language modeling approach to detect English segments in a Chinese show. A open-loop Chinese phone recognizer is used to decode both Chinese BN shows and English BN shows. The output phone sequence is used to train a ngram phonetic language model, one for Chinese and one for English. During testing, each speech segment is first decoded by the Chinese phone recognizer. Then, the output phone sequence is compared to both the Chinese phonetic language model and the English phonetic language model. The likelihood ratio is used to determine the language identity of the segment. The Chinese phonetic language model is trained on a 2-hour subset of the 1997 Hub4 Mandarin training data. The English phonetic language model is trained on a 5-hour subset of the 1996 BN English training data. Bigram phonetic language model is used in both cases.
4) speaker clustering: All speech segments are clustered using a hierarchical, agglomerative clustering algorithm, which employs a tied-GMM based distance measure and a BIC based stopping criteria.

The Ibis single pass decoder was used to decode the evaluation data. Cross-adaptation between the two sets of acoustic models was performed to progressively refine the hypotheses. A 4-gram language model was further used to rescore lattices from earlier stages. We then applied confusion network combination.

The acoustic models were trained on:

a. 27 hours of manually transcribed Broadcast News data released by LDC (LDC98S73)
b. 69 hours of quickly transcribed TDT4 Mandarin data (LDC2003E21)

The language models were trained on:

* Mandarin Chinese News Text Corpus
  * China Radio 1994-1996
  * People's Daily 1991-1996
  * Xinhua News 1994-1996
* TDT2 and TDT3
* TDT4 (excluding text data preceding the last test epoch (Feb 2001))
* Mandarin Gigaword corpus
  * Xinhua News 1990 - 2002 (excluding text data preceding the last test epoch (Feb 2001))
* HUB4m 1997 training transcript
* RFA (web-crawled) from 2001 (excluding text data preceding the last test epoch (Feb 2001)) to Nov 2003
* NTDTV (web-crawled) from 2002 to Nov 2003

The language model training data was modified in order to observe the black-out period for the evaluation.

We incorporated the LDC name entity list into our text segmenter's wordlist and then segmented the text data. Then we derived the word vocabulary from the segmented text. We added the Chinese character set of size 6.7k to the vocabulary. The size of the vocabulary is around 63k. We employed count-mixing approach to train the word trigram and 4-gram LMs. The mixing weight for HUB4m 1997 transcript is set to 6 while the mixing weight for other text sources are set to 1. We used the SRI LM toolkit to train the LM. The LMs were smoothed using Kneser-Ney smoothing scheme. We pruned word trigram and word 4-gram counts by applying count cutoff. The minimum counts of word trigram and 4-gram are 3 and 5 respectively.

The lexicon contains 84K entries derived from the LDC CallHome Mandarin lexicon (LDC96L15). We used a maximal matching technique to generate pronunciations for words not in the LDC lexicon. There are 23 Initials and 34 Finals in the initial-final model, and 38 phonemes in the phone-based models.

Eight additional phonemes are used for noises and silence.

Processing of the test data took about 26 times real-time on a 3.2G Pentium4 single CPU Linux box. Process size was about 600MB during decoding.

8.5 IBM systems

8.5.1 ibm_eval05_epps_en_public_primary

Acoustic front-end. In the IBM English system, the features used to represent the acoustic signal for recognition are 40 dimensional feature vectors obtained from an LDA projection. The source space for the LDA projection is 117 dimensional and obtained by stacking 9 temporally consecutive 13 dimensional acoustic observation vectors. The perceptual linear prediction (PLP) feature vectors are computed at a rate of 100 frames per second from a Hamming windowed speech segment of 25 ms. The vectors contain 13 cepstral parameters obtained from an LPC analysis of the cubic root of the inverse DCT of the log outputs of a 24 band, triangular filter bank. The filters are positioned at equidistant points on the Mel-frequency scale between 0 and 8 kHz. The cepstral parameters are mean and variance normalized on a per cluster (see section 6 for details on how clusters were obtained) basis.

Acoustic model.

In the IBM English system, the acoustic model consists of 5 sets of HMMs all of them trained on the 41 hours of acoustic material available for training as released by RWTH for this project.

a) Speaker Independent (SI) model:
This model uses continuous density left-to-right HMMs using Gaussian mixture emission distributions and uniform transition probabilities. If the number of observations for a tied state s is denoted as Cs, the number of mixtures for that state was determined as 4*Cs^{0}.2. The final mixture distributions are obtained as the result of a mix-up procedure, starting with single Gaussian distributions. Intermediate mixture distribution estimates are obtained using the Expectation-Maximization (EM) algorithm updating the mixture weights, means and covariances. In addition, the model uses a global Semi-Tied Covariance (STC)[16, 44] linear transformation which is also updated at every EM training stage. The sizes of the mixtures are increased in steps interspersed with EM updates until the final model complexity is reached. Each HMM has 3 states except for the silence HMM which is a single state model. The system uses 45 phones, 42 speech phones, 1 silence phone and 2 noise phones. The speech HMMs use 6123 context dependent tied state distributions obtained by decision tree clustering of triphone statistics using context questions based on 73 phonetic classes.

b) Voicing Model:
The evaluation system employs Vocal Tract Length Normalization (VTLN)[60, 42]. The frequency warping is piecewise linear using a breakpoint at 6500Hz. The most likely frequency warping is estimated from among 21 candidate warping factors ranging from 0.8 to 1.2 using a step of 0.02. Warping likelihoods are estimated using a voicing model. This model uses the same state tying as the SI model, however, states model emissions using Gaussian full covariance distributions and the model is based on the 13 dimensional PLP features.

c) VTLN model:
The VTLN model used in VTLN rescoring passes is trained on features in a VTLN warped space. VTLN warping factors are estimated on a per speaker basis for all data in the training set using the voicing model. In that feature space, a new LDA transformation is estimated and a new VTLN model is obtained using the same state tying and mixture component allocation as the SI model but is the result of a new mix-up procedure.

d) Speaker Adaptive Training (SAT) model:
The SAT model used in Constrained Model-space Adaptation (CMA) [16, 44] rescoring passes is trained on features in a linearly transformed feature space resulting from applying CMA transforms to the VTLN normalized features. CMA transforms are computed like the VTLN warping factors on a per speaker basis for all speakers in the training set. In contrast to the VTLN model, the SAT model is obtained by performing a one-pass retraining update from the VTLN space to the VTLN+SAT space. As a result, this model also has the same state tying and mixture allocation as the VTLN and SI models.

e) Minimum Phone Error (MPE) model:
The MPE model used in the MPE rescoring passes is trained on features obtained from a feature-space minimum phone error (fMPE) transformation. The fMPE projection uses 400 Gaussians obtained from clustering the Gaussian components in the SAT model. Posterior probabilities are then computed for these 400 Gaussians for each frame and time spliced vectors of these posterior probabilities are the foundation for the features that are subjected to the fMPE transformation. The fMPE transformation maps the high dimensional posterior-based observation space to a 40-dimensional fMPE feature space [38]. The MPE model is then trained in this feature space using 3 iterations of training using a Minimum Phone Error criterion [39].

Language Model. All decoding passes used a 4-gram Katz back-off model using Good-Turing discounting to reserve probability mass for unseen events and was built using the SRI LM toolkit [57]. One model was trained on the training transcripts and another on the 36M word text corpus released by RWTH for this project. A perplexity minimizing mixing factor was computed using the Dev set reference text. The mixed model was then pruned using an entropy based criterion [56] to 58k unigrams, 2M 2-grams, 2M 3-grams and 1M 4-grams. The language model, lexicon and HMM components were then used to build a static decoding graph of 39M states and 48M arcs.

A second model is a 4-way interpolated back-off 4-gram model using Kneser-Ney smoothing. One of the language models (LM1) was trained on the acoustic transcripts and contains 0.7M n-grams. Another language model (LM2) was trained on the 36M word text corpus released by RWTH for this project and contains 28M n-grams. The other two language models were trained on out-of-domain text sources. One of those language models (LM3) containing 80M n-grams was trained on 525M words of web data released by the University of Washington and the other (LM4) containing 39M n-grams was built on 204M words of Broadcast News. The interpolation weights were (LM1:0.25, LM2:0.55, LM3: 0.1, LM4:0.1). This model was used for lattice rescoring.

Recognition Lexicon. The 56k lexicon was obtained by taking all words occurring at least twice in the joint corpus of 36M word text corpus and the 375k acoustic transcripts. Pronunciations are based on a 45 phone set (42 speech, 1 silence phone and 2 noise phones). Pronunciations were obtained from the Pronlex lexicon and augmented with manual pronunciations.

Recognition process. The first processing step of the IBM English system is to cluster the audio segments into clusters. This clustering is a 3 stage process. First 12 dimensional Mel-frequency cepstral feature vectors augmented with a raw signal energy parameter are computed. Then, on a per segment basis, the 12 dimensional cepstral coefficients are retained for those frames with an energy within 15dB of the maximum energy seen within the segment. In the second stage, text-independent Gaussian mixture models (TIGMMs) are computed for each segment using only the frames selected using the energy selection criterion. The TIGMMs are obtained by a mixing up procedure, starting with a single Gaussian. Mixture weights, means and variances are estimated using the Expectation-Maximization (EM) algorithm up to a complexity of 4 component mixtures. The mix-up process is then continued updating only the weights and means until 32 component TIGMMs are obtained. In the third and final step of the clustering process, the TIGMMs are clustered agglomeratively. The distance metric between segments is based on the log-likelihood loss of scoring the mixture means given the model itself versus the model of the candidate merge. In addition, the distance metric uses a proximity weight increasing the distance for segments that occur farther away in the recording. After no more merges are possibly for a given distance threshold, segments less than 15 seconds in length were merged with the closest cluster. After the clustered are defined, the final system output is obtained in 6 passes:

a) The SI pass uses the SI model and the LDA projected PLP features.

b) Using the transcript from a) as supervision, warp factors are estimated for each cluster using the voicing model and a new transcript is obtained by decoding using the VTLN model and VTLN warped features.

c) Using the transcript from b) as supervision, CMA transforms are estimated for each cluster using the SAT model. A new transcript is obtained by decoding using the SAT model and the CMA transformed VTLN features.

d) The VTLN features after applying the CMA transforms are subjected to the fMPE transform and a new transcript is obtained by decoding using the MPE model and the fMPE features.

e) Using the transcript from d) as supervision, MLLR transforms are estimated for each cluster using the SAT model. A new transcript is obtained by decoding using the MLLR adapted SAT model and the fMPE features.

f) The lattices resulting from e) are rescored using the 4-way interpolated language model.

Execution time.

Recognition experiments were run on a Pentium 4, 2.8GHz Xeon processor with 512kB cache and 2Gb memory.

Execution time and memory use:

1.	segment clustering	780	$<$ 20Mb
2.	SI decode	47436	530Mb
3.	VTLN estimation	2880	$<$ 20Mb
4.	VTLN decode	46564	530Mb
5.	CMA estimation	2520	200Mb
6.	CMA decode	43140	530Mb
7.	fMPE computation	1319	$<$ 20Mb
8.	MPE decode	46132	530Mb
9.	MLLR estimation	1440	100Mb
10.	MLLR decode	46552	540Mb
11.	LM rescoring	6281	2Gb
total for 12562.2 seconds:		245044 (19.5 xRt)	

8.5.2 ibm_eval05_epps_es_restricted_primary

Acoustic front-end. The features used to represent the acoustic signal for recognition are 42 dimensional feature vectors obtained from an LDA projection. The source space for the LDA projection is 117 dimensional and obtained by stacking 9 temporally consecutive 13 dimensional acoustic observation vectors. The feature vectors are computed at a rate of 100 frames per second from a Hamming windowed speech segment of 25 ms. The vectors contain 12 cepstral parameters obtained from an inverse DCT of the log outputs of a 24 band, triangular filter bank. The filters are positioned at equidistant points on the Mel-frequency scale between 0 and 8 kHz. In addition to the 12 cepstral parameters, the vectors contain a raw frame energy parameter. The ceptral parameters are mean and variance normalized on a per cluster basis. (see section 6 for details on how clusters were obtained).

Acoustic model. The acoustic model used in the system consists of 2 sets of HMMs all of them trained on the 41 hours of acoustic material available for training as released by RWTH for this project.

a) Speaker Independent (SI) model:
This model uses continuous density left-to-right HMMs using Gaussian mixture emission distributions and uniform transition probabilities. If the number of observations for a tied state s is denoted as Cs, the number of mixtures for that state was determined as 4*Cs^{0}.2. The final mixture distributions are obtained as the result of a mix-up procedure, starting with single Gaussian distributions. Intermediate mixture distribution estimates are obtained using the Expectation-Maximization (EM) algorithm updating the mixture weights, means and covariances. In addition, the model uses a global Semi-Tied Covariance (STC)[16, 44] linear transformation which is also updated at every EM training stage. The sizes of the mixtures are increased in steps interspersed with EM updates until the final model complexity is reached. Each HMM has 3 states except for the silence HMM which is a single state model. The system uses 54 phones, 49 speech phones, 1 silence phone and 4 noise phones. The speech HMMs use 5414 context dependent tied state distributions obtained by decision tree clustering of triphone statistics using context questions based on 100 phonetic classes.

b) Speaker Adaptive Training (SAT) model:
The SAT model used in Constrained Model-space Adaptation (CMA) [16, 44] rescoring passes is trained on features in a linearly transformed feature space resulting from applying CMA transforms to the SI features. CMA transforms are computed on a per speaker basis for all speakers in the training set. The SAT model is obtained by performing a one-pass retraining update from the SI space to the SAT space. As a result, this model also has the same state tying and mixture allocation as the SI model.

Language Model. Two language models were used, a 3-gram model for acoustic decoding and a 4-gram model for lattice rescoring. For each of them 3 LMs were interpolated with mixing factors that minimized the perplexity on the development set reference text using the SRI LM toolkit [57]. The 3 LMs were build from the training transcripts and the 33M EU Parliament and 43M Spanish Parliament word text corpora released by RWTH for this project. The mixing weights were found as (0.44 0.40 0.16) for the 3-gram and (0.42 0.40 0.18) for the 4-gram model. The mixed 3-gram model was then pruned using an entropy based criterion [56] to 52k unigrams, 2.1M 2-grams and 3.4M 3-grams. The language model, lexicon and HMM components were then used to build a static decoding graph. The 4-gram model for lattice rescoring was interpolated from 0.5M, 29M and 43M n-gram LMs.

Recognition Lexicon. The 52k lexicon was obtained by taking all words occurring in the acoustic transcripts and at least 8 times in either one of the Parliament texts. The resulting OOV rate was about 1% on the development test set. Pronunciations are based on a 54 phone set (49 IBM specific Spanish phones with stress information, 1 silence phone and 4 noise phones).

Recognition process. The first processing step of the system is to cluster the audio segments into clusters. This clustering is a 3 stage process. First 12 dimensional Mel-frequency cepstral feature vectors augmented with a raw signal energy parameter are computed. Then, on a per segment basis, the 12 dimensional cepstral coefficients are retained for those frames with an energy within 15dB of the maximum energy seen within the segment. In the second stage, text-independent Gaussian mixture models (TIGMMs) are computed for each segment using only the frames selected using the energy selection criterion. The TIGMMs are obtained by a mixing up procedure, starting with a single Gaussian. Mixture weights, means and variances are estimated using the Expectation-Maximization (EM) algorithm up to a complexity of 4 component mixtures. The mix-up process is then continued updating only the weights and means until 32 component TIGMMs are obtained. In the third and final step of the clustering process, the TIGMMs are clustered agglomeratively. The distance metric between segments is based on the log-likelihood loss of scoring the mixture means given the model itself versus the model of the candidate merge. In addition, the distance metric uses a proximity weight increasing the distance for segments that occur farther away in the recording. After no more merges are possibly for a given distance threshold, segments less than 15 seconds in length were merged with the closest cluster.

After the clustered are defined, the final system output is obtained in 4 passes:

a) The SI pass uses the SI model and the LDA projected MFCC features. Output indicated in the Exp-ID as: <SYSTEM> = contrast-si
b) Using the transcript from a) as supervision, CMA transforms are estimated for each cluster using the SAT model. A new transcript is obtained by decoding using the SAT model and the CMA transformed features. Output indicated in the Exp-ID as: <SYSTEM> = contrast-sat
c) Using the transcript from b) as supervision, MLLR transforms [3] are estimated for each cluster using the SAT model. A new transcript is obtained by decoding using the MLLR adapted SAT model. Output indicated in the Exp-ID as: <SYSTEM> = contrast-mllr
d) The lattices resulting from c) are rescored using the interpolated 4-gram language models. Output indicated in the Exp-ID as: <SYSTEM> = primary

Execution time.

Recognition experiments were run on a Pentium 4, 2.4GHz Xeon processor with 512kB cache and 3.7Gb memory. Elapsed time for the decoding steps:
Execution time and memory use:

1.	SI-Decoding	164995
2.	CMA-Estimation	9989
3.	CMA-Decoding	126850
4.	MLLR-Estimation	3013
5.	MLLR-Decoding	116011
6.	LM-Rescoring	9865
	total time in seconds	430723

8.6 NOKIA system

Nokia in collaboration with LIMSI participated in the first evaluation campaign. Nokia's primary system is a context dependent phoneme based recognizer. To evaluate the performance of low complexity acoustic modeling approaches, Nokia submitted contrast systems with Scalar quantized and Subspace distribution clustered acoustic models. The results are based on the best hypothesis obtained from the lattices generated by LIMSI.

Acoustic front-end. The waveform is downsampled to 8KHz. Then 13 cepstral coefficients, their first and second derivatives are computed. The zero-th cepstral coefficient and its derivatives are normalised to zero mean and unit variance [59].

Acoustic models. Acoustic models are trained on 194 hours of combined WSJ and Speecon US databases. The subphoneme acoustic units used are context-dependent (CD) cross-word phonemes. Each CD phoneme is modeled by 3 state left to right HMMs. The pdfs in each state are parameterised by 16 component Gaussian Mixture Models. The states are tied using a decision tree based clustering algorithm. The final model set has about 58,000 Gaussians.

Scalar Quantized HMMs. Each mean and inverse-standard deviation components of the trained models are quantised using two Lloyd-Max quantisers [58]. The quantisers were stopped after 500 iterations.

Subspace HMM. Subspace HMMs are derived from continuous density HMMs by first mapping the Gaussians in the model set into disjoint subspaces (streams) and then clustering these subspace Gaussians into a small number of prototype subspace Gaussians [5]. These prototypes are then used to quantize the

Gaussians in the original model set. The subspaces or streams are obtained by clustering mean-variance pair of each feature vector component. This resulted in 39 streams.Subspace HMMs differ from qHMMs in that each stream has its own codebook instead of the single codebook used with qHMMs. Also, qHMMs use a separate codebook for the mean and variance values whereas for subspace HMMs, the codebook elements are made up of a mean-variance pair.

Memory Foot print for quantised acoustic models.

Unquantised HMM: 18 Mega bytes	
Subspace clustered HMM	
4 bits (16 centroids):	1.58 Mega bytes
2 bits (4 centroids):	0.80 Mega bytes
Scalar quantised HMM	
5 bits mean + 3 bits variance:	2.66 Mega bytes
3 bits mean + 1 bit variance:	1.40 Mega bytes

Language Model. The language model scores are obtained from the lattices generated by LIMSI. The lattices were generated using BN LM by LIMSI.

Lexicon. Pronunciations are based on a 42 phone set including silence and interword short pause. The pronunciations are generated using a text to phoneme mapping algorithm. The vocabulary is generated from the list of words in the lattices. There are about 15K words in the vocabulary.

Recognition process. The lattices were rescored using a modified version of HTK software [63].

8.7 SONY system

The Sony system currently used for the TC-STAR project was originally based on the Janus 3 Speech Recognition Toolkit, which was developed conjointly at the Universitt Karlsruhe and Carnegie Mellon University [61]. Sony is continuously extending this original version by integrating its own signal processing library and speech recognition algorithms. This system is mainly used as a research and development tool for applications related to speech and music analysis.

Other speech recognition systems have been developed at Sony for more specific applications and platforms, like the AIBO personal robot, which is using a totally different hardware and decoder. Sony is also developing its own LVCSR decoder, but for research activities the Janus decoder is preferred due to its integration with the Tcl/Tk language scripting capabilities thus allowing a faster implementation of new development ideas and an easier adaptation to new domain and tasks.

Acoustic Front End

The current SONY BN baseline speech recognizer uses 32 LDA coefficients computed on top of a 38 MFCC parameters consisting of 12 cepstrum coefficients, along with the first and second order derivatives and first and second order derivatives of the log energy. Previously the Mel frequency power spectrum is estimated on the 0Hz-8kHz band every 10ms. For each 16ms frame, the signal is first preemphasized with a first order IIR filter (1-0.97*z-1), a Hamming Window is applied and the Mel-scaled FFT power spectrum is computed. The frequency band used by the triangular filter bank is finally 80Hz-7500Hz. The cepstral coefficients are normalized on a segment-cluster basis using cepstral mean removal and variance normalization.

Acoustic model

Each phone model is a tied-state left-to-right CD-HMM with Gaussian mixtures. The triphone-based context-dependent phone models are word-independent but position-dependent. The tied states are obtained by means of a decision tree.

Language model

The trigram language model used for the 2005 evaluation (epps.46k.arpabo.3) was kindly provided by UKA.

The number of N-grams is the following: 1-gram=46265, 2-gram=2165000, 3-gram=8948764.

Recognition lexicon description

The corresponding 46k words dictionary (En.voc46k.dict) was also provided by UKA. It contains 46414 words with 46470 phonetic transcriptions.

Recognition process

The recognition is performed mainly in three steps:

1) Tree Pass, Time synchronous search without word copies on a tree organized dictionary
2) Flat Pass Time synchronous search on a rolled out dictionary: poor-man's trigrams.
3a) Lattice generation
3b) Lattice rescoring: Using the full trigram

Execution time

The execution time in seconds and other information are given in the following table. The execution time includes also the total estimation for the lattice rescoring with the following values of language modeling penalty and weight:
For the tree and fast forward pass: Lz: 26, lp: 12

For the lattice rescoring:

Lz: 20 26 28 30 32
Lp: -10 -5 0 5 10 15 20 25

The results provided in the CTM file are those optimized for the dev05 task with lz: 32 lp: 5.

File name	Processing time	treeFwd	flatFwd	Lattice	Rescoring	Number utterances	Length utterances
20041115_1705_1735	22584	11429	6598	589	4326	135	1019,49
20041116_1505_1800	24922	12164	7846	393	4540	238	2301,28
20041117_0905_1240	52940	29405	14503	845	8244	292	2669,41
20041117_1500_1835	74316	39835	21291	1245	12168	464	3714,3
20041118_1000_1225	36933	19758	10470	585	6242	268	2333,54
20041118_1500_1600	7540	3618	2308	122	1492	60	524,31
Total	219235	116209	63016	3779	37012	1457	12562,33

9 Conclusions

The most significant Automatic Speech Recognition activity in the first year has been the development of performant ASR systems for the European Parliament Plenary Sessions (EPPS). The WP2 partners have been quite successful at reducing the word error rate from 32% for the best baseline system to 10% for the best systems submitted to the TC-STAR Mar'05 evaluation. This significant error reduction has made the speech to text translation of the EPPS data quite a viable task considerably limiting the impact of ASR errors on the overall speech to text Machine Translation process.

The main achievements are the following:

- Development of acoustic models, language models and pronunciation lexicons for European English and Castilian Spanish (as opposed to American English and American Spanish)
- Development of case sensitive vocabularies and language models for English and Spanish
- Improved acoustic modeling for Chinese Mandarin by using lightly supervised training techniques and cross-system adaptation methods
- Development of discriminatively trained acoustic models (e.g. fMPE method) resulting in significant error rate reductions. A number of discriminative training approaches have been studied and experimented with.
- Development of continuous space neural network language models for the EPPS data, allowing significant word error reductions when only limited amounts of on-task training data is available.
- Using multi-site system combination based on a voting scheme or cross adaptation to further improve the ASR accuracy over the best performing systems
- Generation of Enriched transcription for better integration with the MT systems. This enriched transcription includes: capitalization, confidence measures, sentence breaks, word lattices and confusion networks. Experiments have been carried out to demonstrate the importance of word lattices to improve the speech to text MT results.
- Development of small footprint acoustic models and low complexity decoders.
- Development of noise robust speech methods accounting for background noise and reverberation noise.
- Development of complete ASR systems for the EPPS data in Spanish and English, and for broadcast news data in Chinese Mandarin. A total of 30 systems were submitted to the TC-STAR Mar'05 ASR evaluation with participation of all the WP2 partners.

The WP2 activities have been carried out in an efficient manner, respecting the planned schedule and progressing faster than expected due to the early availability of some EPPS data.

10 References

[1] A. M. Abdelatty, J.V. der Spiegel, P. Mueller, “Robust Auditory-Based Speech Processing Using the Average Localized Synchrony Detection.” In *IEEE Trans. on speech and audio processing*, vol.10, no.5., pp.279-292, July 2002.

[2] T. R. Anderson, “A comparison of auditory models for speaker-independent phoneme recognition,” In *Proc. IEEE ICASSP*, pp.231-234, 1993.

[3] J. Blauert, “Spatial Hearing, the psychophysics of human sound localization,” In *the MIT Press, Massachusetts Institute of Technology*, Cambridge, 1996.

[4] F. Brugnara, ”Context-Dependent Search in a Context-Independent Network,” In *Proc. of ICASSP*, 360-363, Hong Kong, April 2003.

[5] E. Bocchieri, and B.K.-W. Mak, “Subspace distribution clustering hidden Markov model, Speech and Audio Processing,” In *IEEE Transactions on ASSP*, Volume: 9, Issue: 3, pp. 264-275, March 2001.

[6] Y. Cao, S. Sridharan and M. Moody, “Co-talker separation using the cocktail party effect,” In *J. Audio Eng. Soc.*, Vol.44, No.12, pp.1084-1096, Dec. 1996.

[7] M. Bisani and H. Ney, “Investigations on joint-multigram models for grapheme-to-phoneme conversion,” in *Proc. Int. Conf. on Spoken Language Processing*, Vol. 1, pp. 105 – 108, Denver (CO), USA, Sep. 2002..

[8] M. Bisani, H. Ney. ”Multigram-based Grapheme-to-Phoneme Conversion for LVCSR”. In Proc. European Conference on Speech Communication and Technology, Vol 2, pp. 933-936, Geneva, Switzerland, September 2003.

[9] M. Bisani and H. Ney, “Open Vocabulary Speech Recognition with Flat Hybrid Models,” submitted to *Proc. European Conf. on Speech Communication and Technology*, Lisbon, Portugal, Sep. 2005.

[10] M. Bodden, “Binaurale Signalverarbeitung: Modellierung der Richtungserkennung und des Cocktail-Party-Effektes,” In *Fortschritt-Berichte VDI*, Nr.85, Reihe 17, 1992.

[11] M. Bodden and T.R. Anderson, “A Binaural selectivity model for speech recognition,” In *Proc. Eurospeech*, pp.127-130, 1995.

[12] S. Deligne, F. Yvon, and F. Bimbot, “Variable-length sequence matching for phonetic transcription using joint multigrams,” in *Proc. European Conf. on Speech Communication and Technology*, pp. 2243–2246, Madrid, Spain, Sept. 1995.

[13] M. Kleinschmidt, J. Tchorz, T. Wittkop, V. Hohmann und B. Kollmeier, “Robuste Spracherkennung durch binaurale Richtungsfilterung und gehrgerechte Vorverarbeitung,” In *Fortschritte der Akustik*, DAGA 98.

[14] L. Chen, J.L. Gauvain, L. Lamel and G. Adda, “Dynamic Language Modeling for Broadcast News,” *Proc. ICSLP’04*, Jeju, October 2004.

[15] J. Fiscus, “A post-processing sys- tem to yield reduced word error rates: Recogniser output voting error reduction (ROVER),” *Proc. IEEE ASRU Workshop*, pp. 347352, Santa Barbara, 1997.

[16] M.J.F. Gales, ”Maximum Likelihood Linear Transformations for HMM-based Speech Recognition”, Computer Speech & Language, Vol. 12, pp.75-98, 1998

[17] Y. Gao, T. Huang, S. Chen and J.P. Haton, “Auditory model based speech processing,” In *Proc. ICSLP*, Vol.1, pp.73-76, Oct. 1992.

[18] Y. Gao, T. Huang and J.P. Haton, “Central auditory model for spectral processing,” In *Proc. IEEE ICASSP*, Vol.2, pp.704-707, 1993.

[19] J.L. Gauvain, L. Lamel and G. Adda, “The LIMSI Broadcast News Transcription System,” *Speech Communication*, **37**(1-2):89-108, May 2002.

[20] J.L. Gauvain, L. Lamel and G. Adda, “Partitioning and Transcription of Broadcast News Data,” *ICSLP’98*, 1335-1338, Sydney, Dec. 1998.

[21] J.L. Gauvain and C.H. Lee, “Maximum A Posteriori for Multivariate Gaussain Mixture Observation of Markov Chains,” *IEEE Trans. on Speech and Audio Processing*, **2**(2):291-298, April 1994.

[22] D. Giuliani, M. Gerosa, F. Brugnara, ”Speaker normalization through constrained MLLR based transforms” In *Proc. of INTERSPEECH*, Jeju Island, Korea, Oct. 2004 vol. 4, pp. 2893-2897

[23] C. Gollan, M. Bisani, S. Kanthak, R. Schlter, H. Ney: “Cross Domain Automatic Transcription on the TC-STAR EPPS Corpus,” Proc. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Philadelphia, PA, March 2005.

[24] H. Hermansky, “Auditory modeling in automatic recognition of speech,” In *Proc. of the first European conference on signal analysis and prediction*, pp.17-21, 1997.

[25] M. J. Hunt and C. Lefbvre, “Speech recognition using an auditory model with pitch-synchronous analysis,” In *in Proc. IEEE ICASSP*, pp. 813-816, 1988.

[26] C. R. Jankowski Jr., H.-D. H. Vo, R. P. Lippmann, “A comparison of signal processing front ends for automatic word recognition,” In *IEEE Trans. Speech Audio Processing*, vol. 3, pp. 286 - 293, July 1995.

[27] S. Kanthak, H. Ney: “FSA: An Efficient and Flexible C++ Toolkit for Finite State Automata using On-demand Computation,” Proc. *42nd Annual Meeting of the ACL*, pp. 510–517, Barcelona, Spain, 2004.

[28] D.-S. Kim, S.-Y. Lee and R.M. Kil, “Auditory processing of speech signals for robust speech recognition in real-world noisy environments,” In *IEEE Trans. on speech and audio processing*, Vol. 7, no.1, pp. 55-69, Jan. 1999.

[29] D. Kocharov, A. Zolnay, R. Schlüter, and H. Ney, “Articulatory Motivated Acoustic Features for Speech Recognition,” submitted to *Proc. European Conf. on Speech Communication and Technology*, Lisbon, Portugal, Sep. 2005.

[30] C.J. Leggetter and P.C. Woodland, ”Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models,” Computer Speech & Language, Vol. 9, pp. 171-185, 1995.

[31] Lexica and Corpora for Speech to Speech Translation Components (LC-STAR), Project funded by the European Commission, No. IST-2001-32216, http://www.lc-star.com

[32] L. Mangu, E. Brill and A. Stolke, “Finding Consensus Among Words: Lattice-Based Word Error Minimization,” *Eurospeeech’99*, 495-498, Budapest, Sep. 1999.

[33] E. Matusov, S. Kanthak, and H. Ney, “On the Integration of Speech Recognition and Statistical Machine Translation,” submitted to *Proc. European Conf. on Speech Communication and Technology*, Lisbon, Portugal, Sep. 2005.

[34] W. Macherey, R. Schlüter, and H. Ney, "Discriminative Training with Tied Covariance Matrices" Proc. of the *8th International Conference on Spoken Language Processing (ICSLP 2004)*, Jeju Island, Korea, Vol. 1, pp. 681-684, October 2004.

[35] W. Macherey, L. Haferkamp, R. Schlüter, H. Ney, "Investigations on Error Minimizing Training Criteria for Discriminative Training in Automatic Speech Recognition," submitted to *Proc. European Conf. on Speech Communication and Technology*, Lisbon, Portugal, Sep. 2005.

[36] H. Ney, "Speech Translation: Coupling of Recognition and Translation," Proc. *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 1149-1152, Phoenix, AZ, 1999.

[37] D. Povey, "Discriminative Training for Large Voculabulary Speech Recognition," *PhD Thesis, Cambridge University*, 2004.

[38] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "fMPE: Discriminatively Trained Features for Speech Recognition," in *Proc. of ICASSP*, 2005.

[39] D. Povey and P.C. Woodland, "Minimum Phone Error and I-smoothing for Improved Discriminative Training," in *Proc. of ICASSP*, 2002.

[40] K. Rateitscheck, "Ein binauraler Signalverarbeitungsansatz zur robusten maschinellen Spracherkennung in lrmerfllter Umgebung," In *Fortschritt-Berichte VDI*, Nr.566, Reihe 10, 1998.

[41] T. Robinson, "British English Example Pronunciations (BEEP), version 1.0", Cambridge University Engineering Department, Cambridge, UK, 1996, ftp://svr-ftp.eng.cam.ac.uk/pub/comp.speech/dictionaries/beep.tar.gz

[42] G. Saon, M. Padmanabhan and R. Gopinath, "Eliminating Inter-Speaker Variability Prior To Discriminant Transforms," in *Proc. of ASRU*, 2001.

[43] G. Saon, G. Zweig, B Kingsbury, L. Mangu, U. Chaudhari, "An Architecture for Rapid Decoding of Large Vocabulary Conversational Speech," in *Proc. of Eurospeech*, 2002.

[44] G. Saon, G. Zweig and M. Padmanabhan, "Linear Feature Space Transformations for Speaker Adaptation," in *Proceedings International Conference on Acoustics Speech and Signal Processing*, Salt Lake City, 2001.

[45] R. Schlüter, T. Scharrenbach, H. Ney, "*Bayes* Risk Minimization using Metric Loss Functions," submitted to *Proc. European Conf. on Speech Communication and Technology*, Lisbon, Portugal, Sep. 2005.

[46] H. Schwenk and J.L. Gauvain, "Connectionist language modeling for large vocabulary continuous speech recognition," In *Proc. IEEE ICASSP*, pp. I: 765–768, 2002.

[47] H. Schwenk and J.L. Gauvain, "Neural network language models for conversational speech recognition," In *Proc. ICSLP* , pp. 1215–1218, 2004.

[48] H. Schwenk and J.L. Gauvain, "Building Continuous Space Language Models for Transcribing European Languages," Submitted to *Eurospeech*, Sep 2005.

[49] S. Seneff, "Pitch and spectral analysis of speech based on an auditory synchrony model," In *Ph.D Thesis, Massachusetts Institute of Technology*, Cambridge, January 1985

[50] S. Seneff, "Vowel recognition based on line-formants derived from an auditory-based spectral representation," In *Proceedings of the Eleventh International Congress of Phonetic Sciences*, Tallinn, Estonia, 1987.

[51] S. Seneff, “A joint synchrony/mean-rate model of auditory speech processing,” In *Journal of Phonetics*, Vol. 16, pp.55-76, 1998.

[52] A. Sixtus, S. Molau, S. Kanthak, R. Schlüter, H. Ney, “Recent Improvements of the RWTH Large Vocabulary Speech Recognition System on Spontaneous Speech,” In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1671-1674, Istanbul, Turkey, June 2000.

[53] A. Sixtus. ”Across-Word Phoneme Models for Large Vocabulary Continuous Speech Recognition”. Dissertation, Aachen, Germany, January 2003.

[54] G. Stemmer, F. Brugnara, and D. Giuliani, ”Adaptive Training Using Simple Target Models” In *Proc. of ICASSP*, Philadelphia, 2005, vol. 1, pp. 997-1000

[55] H. Soltau, B. Kingsbury, L. Mangu, D. Povey, G. Saon, G. Zweig, “The IBM 2004 Conversational Telephony System for Rich Transcription in EARS,” in *Proc. of ICASSP*, 2005.

[56] A. Stolcke, “Entropy-based pruning of backoff languge models,” in *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, pp. 270-274, Lansdowne, VA, Feb. 1998, Morgan Kauffmann.

[57] A. Stolcke. ”SRILM - An Extensible Language Modeling Toolkit”. In Proc. International Conference on Spoken Language Processing, Denver, CO, Sept. 2002.

[58] M. Vasilache, “Speech recognition using hmms with quantized parameters,” In *Proc. of ICSLP*. 2000.

[59] O. Viikki, D. Bye, and K. Laurila, “A recursive feature vector normalization approach for robust speech recognition in noise,” In *ICASSP98*, pages 733-736, Seattle, USA, May 1998.

[60] S. Wegman, D. McAllaster, J. Orloff and B. Peskin, “Speaker Normalization on Conversational Telephone Speech,” in *International Conference on Acoustics Speech and Signal Processing*, Atlanta, 1996.

[61] M. Woszczyna, “Fast speaker independent large vocabulary continuous speech recognition,” In *Ph.D Thesis, Universitt Karlsruhe*, pp.31-32, Feb. 1998.

[62] P.C. Woodland, T. Neieler, E. Whittaker, “Language Modeling in the HTK Hub5 LVCSR,” presented at the 1998 Hub5E Workshop, Sep. 1998.

[63] S.J. Young, “The HTK Hidden Markov Model Toolkit: Design and Philosophy.” In *Tech. Rep. 153*, Cambridge University, UK, 1993.

[64] A. Zolnay, R. Schlüter, H. Ney: “Acoustic Feature Combination for Robust Speech Recognition,” Proc. *IEEE Intern. Conf. on Acoustics, Speech, and Signal Processing*, Philadelphia, PA, March 2005.