The ISL TC-STAR Spring 2006 ASR Evaluation Systems

Sebastian Stüker*, Christian Fügen*, Roger Hsiao†, Shajith Ikbal*, Qin Jin†, Florian Kraft*
Matthias Paulik†, Martin Raab*, Yik-Cheung Tam†, Matthias Wölfel*
*Institut für Theoretische Informatik
Universität Karlsruhe (TH), Karlsruhe, Germany
{stueker, fuegen, shajith, fkraft, mraab, wolfel}@ira.uka.de
† Interactive Systems Laboratories
Carnegie Mellon University, Pittsburgh, PA, USA
{wrhsiao, paulik, qjin, yct}@cs.cmu.edu

Abstract

The project *Technology and Corpora for Speech to Speech Translation* (TC-STAR) aims at making a break-through in speech-to-speech translation research, significantly reducing the gap between the performance of machines and humans at this task. Technological and scientific progress is driven by periodic, competitive evaluations within the project. For automatic speech recognition the Interactive Systems Laboratories participated in the English European Parliamentary Sessions (EPPS) and Mandarin Chinese Broadcast News task within these evaluations. In this paper we present our evaluation systems with which we participated in the TC-STAR Spring 2006 evaluation for the two tasks mentioned.

1. Introduction

TC-STAR - Technology and Corpora for Speech to Speech Translation is a three year integrated project financed by the European Commission within the Sixth Framework Programme. The aim of TC-STAR is to advance research in all core technologies for speech-to-speech translation (SST) in order to reduce the gap in performance between machines and human translators. To foster significant advances in all SST technologies, periodic competitive evaluations are conducted within TC-STAR for all components involved, including automatic speech recognition (ASR) research, as well as end-to-end systems. In the spring of 2006 ASR evaluation the Interactive Systems Laboratories have participated in the English European Parliamentary Plenary Sessions (EPPS) (Gollan et al., 2005) and the Mandarin Chinese Broadcast News evaluation.

Our systems were mainly developed and experiments performed with the help of our own Janus Recognition Toolkit (JRTk) which features the Ibis single pass decoder (Soltau et al., 2001).

2. English EPPS

In this section we describe our English evaluation system for the EPPS task. The models trained are centered around a cross-adaptation and system combination scheme which makes use of models that vary in preprocessing, cluster tree for context-dependent models, and phoneme set. The decoding scheme leads to a word error rate of 10.0% on the official 2006 evaluation set. Further we describe how we produced case-sensitive output that was enriched with punctuation, a new requirement in this year's evaluation.

2.1. Segmentation and Clustering

In last year's evaluation the audio stream had already been manually segmented into utterance like segments before being shipped to the participants. This year's evaluation only gave large blocks of monolingual data without any further segmentation. In order to process the data we first cut it down into shorter segments, satisfying durational constraints, while not throwing away any part of the available speech. Then those segments were clustered for the purpose of unsupervised speaker adaptation during decoding.

Segmentation is performed using speech class posteriors computed for each frame with a multi-layer perceptron (MLP), with a frame size of 32ms and a frame shift of 10ms. The MLP has been trained to classify each frame into speech or non-speech. The audio signal is being preprocessed by calculating 13 mel-frequency cepstral coefficients (MFCC), their deltas and delta-deltas for each frame. Nine consecutive frames are stacked together as input feature to the MLP (current frame, four frames to the left, and four frames to the right); thus the total number of MLP input nodes is 351. The number of hidden nodes used is 1000, and the number of output nodes is 2, one corresponding to speech and the other corresponding to non-speech. The output from the node corresponding to speech is taken as the speech class posterior. After computing posteriors for all the frames, using a threshold value of 0.5 regions of non-speech are identified along with confidence values assigned to them. The confidence for a non-speech region is computed based on its duration and average posterior value. Then, the point of highest confidence in non-speech is searched for in an interval between 0.5 and 20.0 seconds after the starting from the last segment boundary; a first segment break is made at that point. Then with that point as the new starting point further points of segment breaks are found in a similar manner until the end point is reached. It is assumed that the points of high-confidence non-speech would correspond to points of actual sentence breaks and that there will be only one speaker in each segment.

The clustering process aims at grouping the speech segments into several clusters, with each cluster, in the ideal case, corresponding to one individual speaker. A hierarchical, agglomerative clustering technique is used. It is based on TGMM-GLR distance measurement and the Bayesian Information Criterion (BIC) stopping criteria (Jin and Schultz, 2004). A Tied Gaussian Mixture Model

(TGMM) θ was built based on the entire speech segments. Then, one GMM for each speech segment was created via adapting θ to each segment. The Generalized Likelihood Ratio (GLR) distance between two segments Seg_a and Seg_b is defined as

$$D(Seg_a, Seg_b) = -log(\frac{P(X_a \cup X_b|\theta_c)}{P(X_a|\theta_a)P(X_b|\theta_b)}) \quad (1)$$

where X_a, X_b are feature vectors extracted from Seg_a and Seg_b, respectively. θ_a, θ_b, and θ_c are statistical models built on X_a, X_b, and $X_a \cup X_b$ respectively. A symmetric distance matrix is computed corresponding to the pairwise distances between any two segments. At each clustering step, the two segments with the smallest distance are merged, and the distance matrix is updated after each merging. BIC is used as the stopping criterion.

2.2. Preprocessing

For the evaluation system we used four different kinds of acoustic front-ends: *MFCC-I*, *MFCC-II*, *MVDR-I*, and *MVDR-II*. Two are based on the traditional Mel-frequency Cepstral Coefficients (MFCC) and two are based on the warped minimum variance distortionless response (MVDR). The second front-end replaces the Fourier transformation by a warped MVDR spectral envelope (Wölfel and McDonough, 2005), 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.

For training, both front-ends have provided features every 10 ms. During adaptation and decoding this was sometimes changed to 8 ms. In training and decoding, the features were obtained either by the Fourier transformation followed by a Mel-filterbank or the warped MVDR spectral envelope.

For the MVDR-I front-end we used a model order of 80. The resulting 129 spectral coefficients were then reduced to 30 with the help of a linear filterbank. Since the warped MVDR already provides the properties of the Mel-filterbank, namely warping to the Mel-frequency and smoothing, a filterbank has not been used for the MVDR-II front-end and the model order was just 22. The advantage of this approach is an increase in resolution in low frequency regions which cannot be attained with traditionally used Mel-filterbanks. Furthermore, with the MVDR we apply an unequal modeling of spectral peaks and valleys that improves noise robustness, due to the fact that noise is mainly present in low energy regions.

When vocal tract length normalization (VTLN) (Zhan and Westphal, 1997) is applied this is either done in the linear domain for MFCC-I and MFCC-II, or in the warped frequency domain for MVDR-I and MVDR-II. The front-ends use 13 cepstral coefficients with the exception of the MVDR-II front-end which uses 15. The mean and variance of the cepstral coefficients were normalized on a per-utterance basis. In the case of MFCC-I, MVDR-I, and MVDR II, seven adjacent frames were combined into one single feature vector. For MFCC-II the cepstral coefficients were combined with normalized signal energy, approximations of the first and second derivative, and zero crossing rate. For MFCC-I, MVDR-I, and MVDR-II, the resulting feature vectors were then reduced to 42 dimensions using *linear discriminant analysis* (LDA). LDA was also applied to the 43 dimensional MFCC-II feature vectors but without reducing the feature vectors' dimensionality.

AM	# Models	# Gaussians	front-end	P Set
Ia	6k quinphones	23k	MFCC-I	P1
Ib	6k quinphones	23k	MVDR-II	P1
IIa	3k triphones	30k	MFCC-I	P1
IIb	3k quinphones	30k	MVDR-I	P1
IIc	3k quinphones	60k	MVDR-II	P1
III	16k quinphones	18k	MFCC-I	P2
IVa	3k quinphones	60k	MVDR-II	P1
IVb	3k quinphones	60k	MFCC-II	P1

Table 1: Overview of the trained acoustic models (AM), giving the number of models, number of Gaussians, the acoustic front-end used and the phoneme set (P Set) used

2.3. Acoustic Model Training

We trained a variety of phoneme based acoustic models for the final evaluation system. All of them are left-right hidden markov models (HMMs) without state skipping with three HMM states per phoneme. All models were trained on the same approx. 80h of English EPPS data provided by RWTH Aachen within the TC-STAR project.

We trained acoustic models in different sizes for two different kinds of phoneme sets, referred to as *P1* and *P2*, and combined them with the acoustic front-ends introduced in 2.2. P1 is a version of the Pronlex phoneme set which consists of 44 phonemes and allophones while P2 is a version of the phoneme set used by the CMU dictionary that consists of 45 phonemes and allophones. Table 1 gives an overview of the acoustic models trained.

For the models based on P1, at first context independent acoustic models were initialized by taking the global mean over all training data. Several iterations of Viterbi training were then applied in order to train the models. During training cluster based Cepstral Mean Subtraction (CMS) and Variance Normalization (CVN) was applied. Then context dependent models of different sizes were obtained by using our standard top-down clustering approach. All models were trained using incremental splitting of Gaussians training, followed by 2 iterations of Viterbi training. For all models we used one global semi-tied covariance (STC) matrix after LDA (Gales, 1998). In addition to that feature space speaker adaptive training (FSA-SAT) was applied on top for models IVa and IVb.

For the acoustic model III, the only one based on phoneme set P2, forced alignments for the EPPS training data were obtained in the same way as for the ISL-Meeting task system (Metze et al., 2004). A legacy, fully continuous cluster tree trained for meeting and lecture recognition was used. With that, fully continuous models using merge-and-split training as well as two iterations of Viterbi training were created. Thereafter the cluster tree was extended into a semi-continuous cluster tree with 16000 distributions over

corpus	words	weight
EPPS transcripts	750k	0.35
EPPS final texts	33M	0.54
Hub4 BN	130M	0.09
English UN	41M	0.02

Table 2: Overview of the different LM training corpora and their interpolation weights.

4000 codebooks. FSA-SAT training was applied to the semi-continuous models.

2.4. Language Model and Test Dictionary

For language model (LM) training we used data from the following corpora: the EPPS transcriptions, the EPPS final text editions, Hub4 Broadcast News data, and the English part of the UN Parallel Text Corpus v1.0. The specifications for the evaluation demanded the output to be in British English (BE) spelling. However, many of the training corpora are either in American English (AE) spelling or mixed. In order to compensate for that we substituted American English spellings by their British English counterparts in all corpora by using respell, version 0.1 (http://membled.com/work/apps/respell/), which is based on mapping tables build from ispell dictionaries. With the help of the SRI Language Modeling Toolkit (Stolcke, 2002) we build separate 4-gram LMs with modified Kneser-Ney smoothing on each of the corpora and interpolated them together by tuning the interpolation weights on the 2005 EPPS development data. The computed interpolation weights and the size of the different corpora can be seen in Table 2. Thereby we reached a perplexity of 93 on the 2006 EPPS development data.
The British English vocabulary was built by using all words from the EPPS transcripts and all words with more than three occurrences from the EPPS final text editions, which resulted in a case sensitive OOV rate of 0.43% on the 2006 EPPS development data, whereas hyphenated words were split into their constituent parts.
For P1 the initial version of the recognition lexicon was a merger of the callhome_english_lexicon_97061 dictionary and the LIMSI SI-284 training dictionary. Frequently missing words were added manually, all other missing words were generated automatically with the help of the grapheme-to-phoneme conversion tool written by Bill Fisher (Fisher, 1999). The resulting dictionary contained 50k pronunciations.
For P2 the pronunciation dictionary was generated by taking all pronunciation variants of known words from an internal dictionary and generating pronunciation variants for new words with the help of Festival (Black, 1997) resulting in an overall size of 45k pronunciations. After that, we added around 2000 bi- and tri-gram multi word pronunciations.

2.5. Capitalization and Punctuation

The capitalization of the recognition output was done in a post-processing step after the actual decoding procedure which produces case-insensitive output. The decision about a word being upper or lower case is made with the help of a case-sensitive 4-gram language model which is an interpolation of models trained on the EPPS transcriptions and the final text editions (see also 2.4.) Words are being capitalized from left to right, the language model score being the decision criteria.
After capitalization the output was enriched with punctuation with the help of a case-insensitive 4-gram language model and hard coded rules based on pause duration information. Punctuation marks (or in our terminology boundary marks) that were inserted are full stop, comma, question mark, and word boundary (WB), i.e. no punctuation mark. The language model was computed on the EPPS final text editions and the EPPS transcripts, while the pause duration information was extracted from the ASR output by computing the gap between the end time and the start time of two successive words. The hard coded rules based on the pause information were used to define the set of possible types of boundaries that can be inserted after a word. Following rules were used:

- if pause $>$ 0.7 sec $then$ full stop or question mark
- if 0.03 sec $<$ pause $<$ 0.7 sec $then$ full stop, question mark, comma, or WB
- if pause $<$ 0.03 sec $then$ WB

The final decision on which boundary mark to use was made with the help of the language model scores. They were computed by using a sliding window of width seven such that the type of the current boundary B_0 was estimated based on the two words and the one boundary before and after the current boundary.

$$w_{-2}B_{-1}w_{-1}B_0w_{+1}B_{+1}w_{+2}$$

For boundary B_{-1} the punctuation mark type estimated in the previous step was used. For B_{+1} all punctuation marks allowed by the above rules were considered. The difference in WER on the reference transcriptions of the development set for the case insensitive case and for counting punctuation marks as words was approximately 1.0% absolute better when automatically inserting punctuation marks compared to inserting no punctuation marks at all.

2.6. Decoding Strategy and Results

Decoding is organized in five stages. Each stage consists of several decoding passes whose outputs are combined with the help of confusion network combination (CNC) (Mangu et al., 2000). One decoding pass consists of a decoding with lattice generation and rescoring of that lattice with a different language model weight and word penalty. The decoding passes in the first stage are performed without speaker adaptation; all other passes are adapted on the output from a previous stage using Maximum Likelihood Linear Regression (MLLR) (Leggetter and Woodland, 1995), Vocal Tract Length Normalization (VTLN) (Zhan and Westphal, 1997), and feature-space constrained MLLR (fMLLR) (Gales, 1997). Adaptation and decoding of the systems was either done with the 10ms frame shift used in training or an 8ms frame shift. In the end the output of several CNCs was combined using ROVER (Fiscus, 1997).

Table 3 summarizes the stages with their respective decoding passes, describing which acoustic model, as given in 2.3., was used and which frame shift. It also gives the word error rates achieved on the official TC-STAR 2006 development and evaluation sets for the different stages and passes.

3. Mandarin Broadcast News System

In this section we describe our effort on the development of the Mandarin Broadcast News transcription system. We first introduce the acoustic model in Section 3.1., followed by the language model in Section 3.2.. We describe the development of language model adaptation in Section 3.3. followed by the recognition results in Section 3.4..

3.1. Acoustic Modeling

The feature extraction of the Mandarin Chinese system is similar to MFCC-I described in section 2.2.. The window size is 16ms and a feature vector is extracted every 10ms. The feature vectors are concatenated with 15 adjacent frames, and LDA is performed to extract the final 42-dimension feature vectors. CMS, CVN, and VTLN are applied on a per speaker/cluster basis.

Two acoustic models were built with different modeling units: initial-finals (I-F) and phones. Both models are context dependent and clustered using decision trees. The I-F system has 3000 clustered triphone states and a total of 168k Gaussians; the phone system has 3000 septaphone states with a total of 169k Gaussians. Tonal information was incorporated 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 covariance matrix (STC) was employed (Gales, 1997). The acoustic models were trained in a cluster adaptive way, which uses cluster base feature space transforms (FSA-SAT). During testing, speaker adaptation was carried out on the features (FSA), and the Gaussian means (MLLR).

The data for acoustic models consists of two data sets: 27 hours of manually transcribed Mandarin Broadcast News Data released by Linguistic Data Consortium (LDC), and 69 hours of quickly transcribed TDT4 Mandarin data. The CMU segmenter is used to produce the initial segmentation (Siegler et al., 1997). The TDT4 data does not have noise annotations and may include minor transcription errors. The TDT4 segments in the original transcripts may have more than one speaker per segment. They were resegmented at major silences located through forced alignment.

3.2. Language Modeling

Several corpora were used for our LM development: Mandarin Chinese News Text, TDT{2, 3, 4}, the Mandarin Gigaword corpus, the HUB4m 1997 training transcript and some web-crawled data from RFA and NTDTV. Any text falling into the black out period defined in the evaluation guidelines was removed.

The Chinese text data was first preprocessed to normalize for ASCII numbers, ASCII strings and punctuations. Heuristic rules were devised in combination with a Maximum Entropy (Maxent) classifier to normalize the numbers. The classifier identifies whether a number is a digit string (e.g. telephone number) or a quantity by using the surrounding text. English words were mapped to a special token "+english+", and human noises (such as breath and cough) to "+human_noise+". Environmental noises were removed from the HUB4m training transcript. Punctuations provided word boundary information for word segmentation, and they were removed after word segmentation.

We incorporated the LDC Name-Entity (NE) list into our text segmenter's word list. The NE list has different semantic categories such as organization, company, person and location names. This addition to segmenter's word list improved segmentation quality, which leads to more accurate predictions in the LM.

The word vocabulary was derived from the segmented text based on frequency count. The commonly used Chinese character set of size 6.7k was added to the vocabulary. The size of the resulting vocabulary is around 63k. We employed the 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 modified Kneser-Ney smoothing scheme. We pruned word trigram and word 4-gram counts by applying count cutoffs. The minimum counts of word trigram and 4-gram are 3 and 5 respectively.

3.3. Language Model Adaptation

We applied dynamic language model adaptation from our previous work (Tam and Schultz, 2005) using the Latent Dirichlet Allocation model (Blei et al., 2003). The Latent Dirichlet Allocation (LDA) model is a Bayesian model for Latent Semantic Analysis (LSA) which tries to capture the latent topics of a document corpus. In broadcast news, a document usually refers to a piece of news story within which the latent topics are consistent. One view of the LSA model is a Bayesian extension of a mixture of unigram LMs where the topic mixture weight vector θ is drawn from a prior Dirichlet distribution:

$$f(\theta;\alpha) \quad \propto \quad \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \tag{2}$$

where $\alpha = \{\alpha_1, ..., \alpha_K\}$ represents the prior observation count of the K latent topics and $\alpha_k > 0$. As a "bag-of-word" generative model, the LDA model assigns probability to a document $w_1^n = w_1 w_2 ... w_n$ as follows:

$$Pr(w_1^n) \quad = \quad \int_\theta \left(\prod_{i=1}^{n} \sum_{k=1}^{K} \beta_{w_i k} \cdot \theta_k \right) f(\theta;\alpha) d\theta \tag{3}$$

where $\beta_{w_i k}$ denotes the probability of a word w_i given the k-th latent topic. Optimizing the exact likelihood is computationally intractable. One alternative is to optimize the lower-bound of the log likelihood. It turns out that the lower bound of the log likelihood has the following form:

$$Q(\Lambda, \Gamma) \quad = \quad E_q[\log \frac{f(\theta, w_1^n, z_1^n; \Lambda)}{q(\theta, z_1^n; \Gamma)}] \tag{4}$$

stage/pass	technique	AM/pass	adapted on	frame shift	Dev06	Eval06
1a	Decoding	Ia	–	10ms	19.6%	15.9%
1b	Decoding	Ib	–	10ms	19.4%	16.0%
1	CNC	1a,1b	–	–	17.8%	14.9%
2a	Decoding	IIa	1	8ms	15.6%	12.7%
2b	Decoding	IIb	1	8ms	15.2%	12.5%
2c	Decoding	IIc	1	8ms	15.5%	12.7%
2	CNC	2a,2b,2c	–	–	14.8%	12.2%
3a	Decoding	III	2	10ms	14.0%	11.0%
3b	Decoding	III	2	8ms	13.8%	10.9%
3	CNC	3a,3b	–	–	13.4%	10.5%
4a	Decoding	IVa	3b	8ms	13.5%	10.6%
4b	Decoding	IVb	3b	8ms	13.4%	11.0%
4i	CNC	4a,4b	–	–	13.0%	10.3%
4ii	CNC	4a,4b,3a,3b	–	–	12.8%	10.0%
5	ROVER	4i,4ii,3	–	–	**12.7%**	**10.0%**
5.c	Casing		–	–	–.-%	11.1%
5.p	Punctuation		–	–	–.-%	16.7%
5.p+c	Punctuation + Casing		–	–	–.-%	17.7%

Table 3: Overview of the decoding scheme with the individual stages and passes, giving the acoustic model (AM) used, or combined decodings respectively, stages adapted on, frame shift used, and word error rate on the 2006 development and evaluation sets

where $q(\theta, z_1^n)$ is an approximate posterior distribution over the latent variables, the topic mixture weights θ and the latent topic sequence z_1^n given an observed document. In Variational Bayes inference (Jordan et al., 1999), the distribution is factorisable and parameterized by Γ:

$$q(\theta, z_1^n; \Gamma) \quad = \quad q(\theta) \cdot \prod_{i=1}^{n} q(z_i) \tag{5}$$

where $q(\theta)$ is a Dirichlet distribution over topic mixture weights, and $\{q(z_i)\}$ is a set of multinomial distributions over topic indices. Optimizing the auxiliary function Q(.) can be performed using the VB-EM algorithm. The E-step determines the parameters Γ of variational posteriors q(.) and the M-step uses q(.) to re-weight the observations to estimate the model parameters Λ. We only show the results of the parameter estimations of a single document. Complete derivations can be found in (Blei et al., 2003).
E-Step:

$$\gamma_k \quad = \quad \lambda \cdot \alpha_k + \sum_{i=1}^{n} q(z_i = k) \tag{6}$$

$$q(z_i = k) \quad \propto \quad \beta_{w_i k} \cdot e^{E_q[\log \theta_k]} \tag{7}$$

Eqn 6 and Eqn 7 are applied iteratively until convergence.
M-Step:

$$\beta_{vk} \quad \propto \quad \sum_{i=1}^{n} q(z_i = k)\delta(w_i, v) \tag{8}$$

where $\delta(.)$ is the Kronecker Delta function. Parameters of the Dirichlet prior $\{\alpha_k\}$ can be determined using the Newton-Raphson algorithm or gradient ascent procedure.
In dynamic LM adaptation, the idea is to have an adaptive unigram LM in which the topic mixture weights are adapted according to the decoded word hypotheses from the previous speech utterances. The topic mixture weights can be estimated by first running the E-step and normalizing the topic posterior counts γ_k of each k-th topic:

stage/pass	w/o LSA	w/LSA
[1] SI (I-F)	19.6%	19.5%
[2] SAT (I-F)	15.8	15.1
[3] SAT (phone)	14.8	14.4
[4] SI (I-F)	14.1	14.1
[5] SAT (phone)	14.3	-
[5.1] SAT (I-F,phone,8ms)	{14.4,14.7}	-
CNC	13.9	13.8
[6] x-adapt (LIMSI)	12.4	12.0

Table 4: Character Error Rates (%) of the Mandarin BN system on the dev06 set.

$$Pr_{lsa}(w) \quad = \quad \sum_{k=1}^{K} \beta_{wk} \cdot \hat{\theta}_k \tag{9}$$

$$\text{where } \hat{\theta}_k \quad = \quad \frac{\gamma_k}{\sum_{k=1}^{K} \gamma_k} \quad (k = 1...K) \tag{10}$$

The adaptive unigram LM is interpolated with the background N-gram LM in an on line fashion. The adaptive unigram LM works like a cache-based LM. But instead of caching the word counts in the history, we cache the latent topic counts in the history. Since word hypotheses contain recognition errors, caching the latent topic counts may be less susceptible to reinforcing recognition errors back to the LM. In our setting, we adapted the LSA-unigram LM using the past 20-word window. The decaying factor λ is set to 0.4 in the E-step to compensate for the topic switching in a BN show. We determined these tuning parameters by minimizing the perplexity of the RT04 development set.

3.4. Decoding Strategy and Results

We employed a multi-pass decoding strategy by cross-adapting our I-F system with our phone-based system. We performed the first-pass decoding using the speaker independent I-F models. Then we used the decoded hypotheses to adapt the speaker-adaptive I-F models and performed the second-pass decoding. The speaker-adaptive decoding was performed with VTLN, constrained MLLR and model MLLR similar to the decoding strategy employed in our English system. In our third-pass decoding, we applied the speaker-adaptive phone models, followed by the constrained decoding using the speaker-adaptive I-F model. We then combined the lattices from the second to fourth decoding passes using CNC. Finally, we exchanged our word hypotheses after CNC with LIMSI's Mandarin BN system to cross-adapt our speaker-adaptive I-F models. Table 4 shows the stage wise recognition results in character error rates (CER) on the dev06 development set. The second column in the table shows the results from year 2005 without LM adaptation and the third column shows the results from year 2006 with LM adaptation using LSA. Results showed that we successfully reduced the number of decoding passes without degradation in recognition performance. LM adaptation helps for the second and third decoding passes, but we observed no performance gain at the fourth decoding pass. However, when we combined the word lattices generated from the speaker-adaptive decoding passes using CNC, we observed a slight performance gain. On the other hand, when we cross-adapted our I-F system using LIMSI's hypotheses, we achieved 0.4% absolute CER reduction due to LM adaptation.

4. Conclusion

In this paper we have described our speech recognition systems with which we have participated in the second TC-STAR evaluation campaign in spring of 2006. In this campaign the Interactive Systems Laboratories have participated in the English EPPS evaluation, obtaining a word error rate of 10.0% and in the Mandarin Chinese Broadcast News evaluation, achieving a word error rate of 9.8% in cooperation with LIMSI.

5. Acknowledgments

This work has been funded by the European Union under the integrated project TC-Star - Technology and Corpora for Speech to Speech Translation - (IST-2002-FP6-506738, http://www.tc-star.org). The authors would like to thank Susanne Burger for her help in the preparation of the phoneme sets and Sharath Rao for his help in developing the punctuation procedure.

6. References

Alan Black 1997. The festival speech synthesis system: System docmunation. Technical report, Human Communication Research Centre, University of Edingburgh, Edingburgh, Scotland, United Kingdom.

D. Blei, A. Ng, and M. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*.

Jonathan Fiscus. 1997. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover). In *ASRU*, Santa Barbara, CA, USA.

W.M. Fisher. 1999. A statistical text-to-phone function using ngrams and rules. In *ICASSP*, Phoenix, AZ, USA.

M.J.F. Gales 1997. Maximum likelihood linear transformations for hmm-based speech recognition. Technical report, Cambridge University, Engineering Department.

M.J.F. Gales 1998. Semi-tied covariance matrices for hidden markov models. Technical report, Cambridge University, Engineering Department, February.

Christian Gollan, Maximilian Bisani, Stephan Kanthak, Ralf Schlüter, and Hermann Ney. 2005. Cross domain automatic transcription on the tc-star epps corpus. In *International Conference on Acoustics, Speech, and Signal Processing*, Philadelphia, PA, USE.

Qin Jin and Tanja Schultz. 2004. Speaker segmentation and clustering in meetings. In *Proceedings of the 8th International Conference on Spoken Language Processing*, Jeju Island, Korea, October.

M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. 1999. An introduction to variational methods for graphical models. *Journal of Machine Learning Research*, pages 183–233.

C.J. Leggetter and P.C. Woodland. 1995. Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer Speech and Language*, 9:171–185.

L. Mangu, E. Brill, and A. Stolcke. 2000. Finding Consensus in Speech Recognition: Word Error Minimization and Other Applications of Confusion Networks. *Computer Speech and Language*, 14(4):373–400, Oktober.

F. Metze, Q. Jin, C. Fügen, K. Laskowski, Y. Pan, and T. Schultz. 2004. Issues in meeting transcription - the ISL meeting transcription system. In *Proceedings of the 8th International Conference on Spoken Language Processing*, Jeju Island, Korea, October.

M. Siegler, U. Jain, B. Raj, and R. Stern. 1997. Automatic segmentation, classification and clustering of broadcast news audio. In *DARPA Speech Recognition Workshop*, Chantilly, VA, USA, February.

H. Soltau, F. Metze, C. Fügen, and A. Waibel. 2001. A one pass-decoder based on polymorphic linguistic context assignment. In *ASRU*, Madonna di Campiglio Trento, Italy, December.

A. Stolcke. 2002. SRILM – An Extensible Language Modeling Toolkit. Denver, Colorado, USA.

Y. C. Tam and Tanja Schultz. 2005. Dynamic language model adaptation using variational bayes inference. In *Proceedings of the 9th European Conference on Speech Communication and Technology*, Lisbon, Portugal, September.

M.C. Wölfel and J.W. McDonough. 2005. Minimum variance distortionless response spectralestimation, review and refinements. *IEEE Signal Processing Magazine*, pages 117–126.

Puming Zhan and Martin Westphal. 1997. Speaker normalization based on frequency warping. In *ICASSP*, Munich, Germany.