ADVANCES IN ARABIC BROADCAST NEWS TRANSCRIPTION AT RWTH

David Rybach, Stefan Hahn, Christian Gollan, Ralf Schlüter, Hermann Ney

Human Language Technology and Pattern Recognition
Computer Science Department, RWTH Aachen University, Germany
{rybach,hahn,gollan,schluter,ney}@cs.rwth-aachen.de

ABSTRACT

This paper describes the RWTH speech recognition system for Arabic. Several design aspects of the system, including cross-adaptation, multiple system design and combination, are analyzed. We summarize the semi-automatic lexicon generation for Arabic using a statistical approach to grapheme-to-phoneme conversion and pronunciation statistics. Furthermore, a novel ASR-based audio segmentation algorithm is presented. Finally, we discuss practical approaches for parallelized acoustic training and memory efficient lattice rescoring. Systematic results are reported on recent GALE evaluation corpora.

Index Terms— Speech Recognition, System Combination, Cross-Adaptation, Audio Segmentation

1. INTRODUCTION

Arabic poses new challenges for research in human language technologies. The morphological complexity of Arabic and other language characteristics, like missing pronunciation information in Arabic texts [1], introduce new requirements in the design of speech recognition systems.

In this paper, we describe our automatic speech recognition (ASR) system for Arabic and present systematic results on recent evaluation corpora of the Global Autonomous Language Exploitation (GALE) project [2]. The recognition system consists of three subsystems each using a different acoustic model. We illustrate the multiple system design and analyze the effects of cross-adaptation and system combination techniques. The gain of cross-adaptation methods is also shown by improved results on English corpora of the TC-STAR Evaluation Campaign. The Arabic lexicon used is augmented using a statistical approach to grapheme-to-phoneme conversion and pronunciation statistics.

In addition, we go into some practical aspects of our system: a fast maximum likelihood training for acoustic models that is useful for processing large amounts of data and a memory efficient lattice rescoring technique required when dealing with huge language models. Furthermore, we describe a new method for audio segmentation using several features derived from the output of a speech recognizer.

The remainder of this paper is organized as follows: First, we describe the acoustic models in Section 2 and the language model together with the lexicon in Section 3. Section 4 depicts our decoding architecture. Practical aspects are covered by Section 5. Finally, we present experimental results in Section 6.

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Defense Advanced Research Projects Agency.

2. ACOUSTIC MODELS

We have three different acoustic models for our subsystems. The features used are described in the next section. The acoustic models, their training and the speaker normalization and adaptation methods applied are presented in the following sections.

2.1. Features

The baseline acoustic front end consists of Mel frequency cepstral coefficient (MFCC) features derived from a bank of 20 filters. We use 16 cepstral coefficients (including the zeroth coefficient) which are normalized using cepstral mean and variance normalization. These MFCC features are augmented with a voicedness feature [3].

The gammatone filterbank is reported to simulate the human auditory filter well. We use gammatone cepstral coefficients as described in [4], normalized like the MFCCs by cepstral mean and variance normalization. We add the voicedness feature to the gammatone cepstral features as well.

Phone posterior features estimated by a neural network (NN) are the third type of features used. The NN is trained on phoneme classes obtained by a phone alignment, which is generated using HMM models. The input to the NN are multiple time resolution features, which are based on PLP features [5]. A prior version of the NN features is described in [6], the extended implementation currently used is depicted in [7].

In order to incorporate temporal context into the acoustic features, we concatenate 9 consecutive feature vectors in a sliding window. The concatenated feature vector is projected to a feature space of lower dimension by applying a linear discriminant analysis (LDA).

Table 1. Transcribed recordings used for acoustic modelling.

Transcribed Data [h]	450.8
#Segments	209,656
#Running Words	3,256,278
Silence ratio [%]	10.3

2.2. Models

The phonemes in triphone context are modeled by 6-state hidden Markov models. Systems 1 and 3 consider triphone context across word boundaries. Classification and regression trees tie the triphone states to 4500 generalized triphone states in Systems 1 and 2, System 3 has 5000 states.

System 1 and 2 use MFCC and NN features, the resulting feature vectors have 70 and 80 components respectively. The front end of System 3 consists of gammatone and voicedness features projected by an LDA to feature vectors of 60 dimensions.

2.3. Speaker Normalization and Adaptation

Both systems using MFCC features apply vocal tract length normalization (VTLN) to the filterbank. During recognition, the warping factors are estimated by a Gaussian mixture classifier on a sliding window, which allows us to apply VTLN already in the first recognition pass and even on unsegmented data.

Speaker variations are compensated by speaker adaptive training (SAT) based on constrained maximum likelihood linear regression (CMLLR) [8].

2.4. Training

All the acoustic models are trained on the same corpus, which is described in Table 1. The roughly 450h of audio material are taken from two domains: broadcast news and broadcast conversation. Parts of the transcripts have been derived automatically or are quick transcriptions. The basic acoustic models are trained with the maximum likelihood method ("Viterbi training").

Discriminative training with the minimum phone error criterion [9] is performed to enhance the acoustic models. The word conditioned lattices required for the discriminative training are created in one pass with a unigram language model. This weak language model is important for the discriminative training because it allows for more diverse paths in the lattice, resulting in a better generalization [10]. We produce two lattice sets for the training data: one for the MFCC based Systems 1 and 2 and one for System 3, created using a decoder with the corresponding acoustic model.

3. LEXICON AND LANGUAGE MODEL

Since Arabic is a morphologically rich language, the modeling of pronunciation lexica is a challenging task. Basically, two problems have to be addressed. First, even with vocabularies consisting of 256k words, the out-of-vocabulary (OOV) rate is fairly high compared to e.g. English where we obtain an OOV rate lower than 0.7 % with a lexicon consisting of 53k words (cp. Tables 3 and 2) [11]. Second, diacritics are usually omitted in written Arabic texts. These diacritical marks are used for disambiguation and clarify the actual pronunciation. A word written without diacritics may have various meanings depending on the pronunciation. Thus, the meaning of a written word may often only be clear if the context is available.

Our Arabic pronunciation lexicon was derived from the lexicon of the LC-STAR project [12]. A statistical grapheme-to-phoneme conversion model [13] was trained based on this lexicon. This model was used to generate pronunciations for words not covered by the original lexicon. The pronunciations are constructed using a set of 34 phonemes.

For all of the experiments presented, we use the same recognition lexicon consisting of 256k words with approximately 429k distinct pronunciations. The pronunciation scores are based on relative frequencies of pronunciations calculated on a Viterbi alignment of the acoustic training data. We incorporate alignments from all of our three acoustic models to obtain more robust scores.

The two language models that we used for the recognition task are based upon a trigram backing-off LM provided by SRI. This LM was trained on parts of the FBIS, TDT4, Gigaword Arabic corpora, and data released especially for the GALE project. For word lattice rescoring, the LM contains approximately 265M multi-grams. A pruned version with about 55M multi-grams is used for lattice generation.

4. DECODING ARCHITECTURE

The recognition is performed in three passes, as depicted in Figure 1. The test corpus is segmented beforehand using the output of System 1 with a speaker independent acoustic model. The resulting segments are clustered using a generalized likelihood ratio clustering with Bayesian information criterion based stopping condition. The segment clusters act as speaker labels required by the adaptation techniques in the following steps.

System 2 performs the initial recognition pass whose output is required for the text dependent speaker adaptation in the next step. The CMLLR matrices for each system are calculated in pass two and are used for a first speaker dependent recognition. The lattices produced in this pass are rescored with an unpruned trigram language model, while the decoder uses a pruned trigram language model for lattice generation.

The second adaptation technique we use is maximum likelihood linear regression (MLLR) which is applied to the means of the acoustic models. The number of regression classes is adjusted according to the amount of available data by a regression class tree. We use cross-adaptation.

The decoding in the third pass is carried out using the MLLR transformed means and the CMLLR transformed fea-

tures. The lattices produced are again rescored with the unpruned trigram language model. Eventually, the final results can be combined to a single word sequence using ROVER.

4.1. Segmentation

The basic input to the recognition system is nearly unsegmented audio material from broadcast news recordings. Several approaches exist for splitting this audio stream in segments such that a segment includes only one speaker and (ideally) one sentence. The quality of the segmentation affects the recognition performance since segments are assumed to be spoken by a single speaker. Furthermore, the language model performs better if segment boundaries correspond to sentence boundaries.

Many segmentation methods use acoustic features directly to find segment boundaries. It has been shown that the segmentation performance can be increased by incorporating the output of a speech recognizer [14]. A simple approach to use the recognizer output for segmentation is to split the recordings at positions where silence is recognized with a duration longer than some threshold [6]. This approach makes local decisions, disregarding context, properties of surrounding segments, and speaker changes.

Our segmentation method optimizes the whole sequence of segments with respect to a context dependent segment score. The segment scoring function used incorporates several features: the segment length, variance of warping factors in the segment which corresponds to speaker homogeneity, and the word confidence of hypothesized boundary tokens.

The optimization can be expressed as a maximization over the sequence of segment boundaries $b_1^N = b_1 \dots b_N$:

$$[b_1^N]_{\text{opt}} = \underset{N,\, b_1^N:\, b_N = B}{\operatorname{argmax}} \left\{ \sum_{i=1}^{N} c(b_{i-1}, b_i) \right\}$$

where the b_i are the segment boundaries with B the recording end, and $c(b_i, b_j)$ is the segment scoring function. The optimization problem is solved with dynamic programming. A beam search has to be used in order to deal with the high number of possible segmentation hypotheses.

4.2. Speaker Adaptation

We use cross-adaptation [15] to benefit from our setup of three systems. Therefore, we combine the first best word sequences of the rescored lattices of pass 2 in such a way that for each system the results of the two other systems are used for adaptation (see Figure 1). For this cross-adaptation, each pair of systems is combined by ROVER using confidence scores.

5. PRACTICAL ASPECTS

The tasks presented in this paper require processing of large amounts of data. We describe two methods which reduce runtime and memory requirements for maximum likelihood training and language model rescoring on word lattices.

5.1. Fast Parallel Maximum Likelihood Training

Due to the morphological complexity of Arabic, a lot of different triphone contexts occur with significant frequencies. Thus, the tying of allophone states results in more mixture models. We have to process large amounts of training data to estimate these models reliably. The maximum likelihood (ML) training of the HMMs (Viterbi approximation) consists of several steps in each iteration:

1. Time alignment
2. Collection of all observations for each state
3. Estimation of the Gaussian mixture HMM parameters
4. Splitting of the mixtures (optional)

We do not count transitions here because we use fixed transition probabilities. The maximum approximation for mixture densities is used. For the estimation of a mixture model, it is therefore sufficient to assign the observations to densities of the mixture and estimate weights, means, and the covariance accordingly. We use a global pooled diagonal covariance matrix which requires the weighted accumulation of observations from all mixtures.

This algorithm is expensive in terms of computation time and memory requirements. Furthermore, it is not well suited for parallelization, because each step depends on the results of the preceding one. We can easily split up each step in small subtasks, but the next step still requires the results of all preceding subtasks.

The first approximation we use is to keep the time alignment fixed for a complete training cycle. This allows us to sort the feature vectors in caches sorted by their assigned mixture index. Thus, the accumulation of observations can be done for each mixture separately and in parallel. However, the estimation of the covariance matrix still requires a synchronization of all tasks. Consequently, we use mixture specific covariances until the splitting of mixtures is finished. Using this procedure, we can virtually train all mixture models in parallel. An implementation using a kind of the MapReduce programming model [16] is convenient. The memory requirements are small because each task keeps only a single (or a few) mixture model(s) in memory.

In order to obtain a good estimate of the global pooled variance vector after the estimation of mixture specific variances, we calculate the pooled variance vector subsequent to the splitting steps and perform a few training iterations (without further mixture splitting) using this vector. The variance vector is re-estimated after each iteration.

5.2. Memory Efficient Lattice Rescoring

Practical problems occur if we want to use n-gram language models for morphologically rich languages like Arabic. Without pruning the language model, it might consist of so many n-gram scores that it does not fit into memory anymore. However, for language model rescoring of lattices it is not necessary to load the whole language model because we can determine which multi-grams are needed beforehand. Therefore,

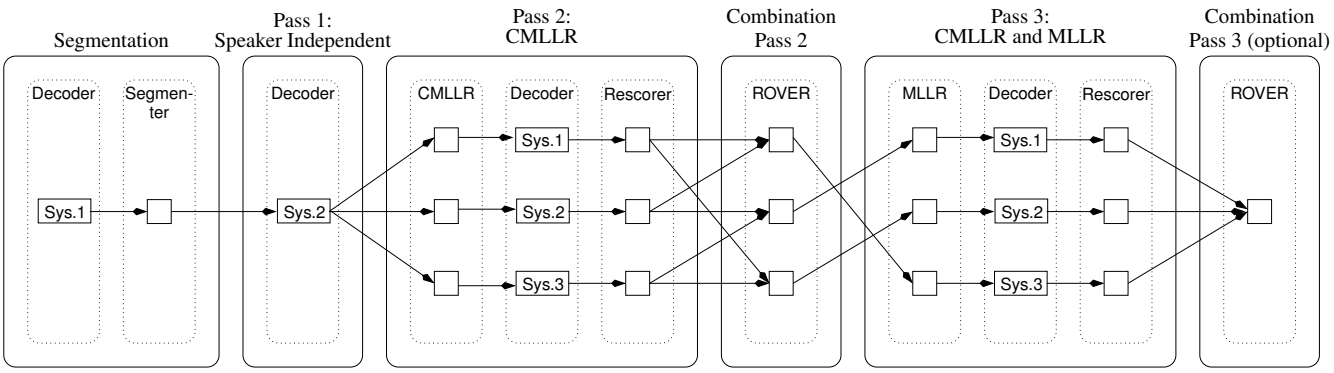

Fig. 1. Illustration of the decoding process using 3 systems.

Table 2. Statistics for the GALE Arabic corpora.

	Dev07	Eval06
Audio Data [h]	2.6	3.3
# Segments	586	815
# Running Words	18k	22k
Silence ratio [%]	10.6	11.5
OOV rate [%]	2.12	5.69

we collect all multi-grams occurring in a lattice and load only the corresponding scores of the language model. To reduce the computational overhead, we collect the multi-grams for all lattices of the corpus to transcribe. Thus, in our experiments with a trigram language model, the number of n-grams can be reduced from over 250 million to about 400,000. A similar technique for n-best lists was mentioned in [17]. The SRI Toolkit [18] offers language model filtering as well, but based on word lists instead of multi-gram lists.

6. EXPERIMENTAL RESULTS

In this section we discuss the results of the RWTH speech recognition system and describe the corpora used.

6.1. Corpora

The GALE corpora used for evaluation consist of Arabic broadcast news and broadcast conversations from several TV channels in various countries. In Table 2, statistics for the GALE development and evaluation corpora are shown. The number of segments refers to the automatically generated segmentation as described in Section 4.1. For control experiments of our decoding architecture, we performed experiments on the English EPPS task [6]. The statistics for the respective corpora are given in Table 3.

6.2. Results

First, we describe the results obtained with the fast ML training of the acoustic model described in Section 5.1. Table 4

Table 3. Statistics for the TC-STAR EPPS English corpora.

	Eval06	Eval07
Audio Data [h]	3.2	2.9
# Segments	742	644
# Running Words	30k	26k
# Speakers	41	50
Silence ratio [%]	10.3	9.7
OOV rate [%]	0.55	0.61
PP recog. LM	106.6	92.3
PP rescoring LM	96.3	87.2

Table 4. Results for different training methods (System 3, speaker independent, BNAT05 Corpus).

covariance pooling applied	WER [%]
after each split (baseline)	22.4
none	23.0
after all splits	21.8
after all training iterations	21.9

lists the results on a development corpus obtained with System 3 for different variants of covariance pooling.

The use of mixture specific covariances during recognition (without applying any smoothing or tying methods) deteriorates the recognition performance compared to the baseline training method, because of underestimated covariances. Nevertheless, the usage of mixture specific covariances for mixture *splitting* produces a better acoustic model, since the distribution of observations in a single mixture can be modelled more accurately. Pooling the covariance after all splitting steps and performing three further training iterations with a pooled covariance gives the best results.

The next issue we studied is the improvement obtained by our new segmentation. We compare a previous ASR-based segmentation that uses silence duration as sole criterion to the segmentation produced with the method described in Section 4.1. We have not evaluated the segmentation using a reference segmentation but compared the resulting recognition results directly. Table 5 shows that the new segmentation

Table 5. Results of the 2. pass of System 3 for different segmentations.

	WER [%]	
segmentation criterion	Dev07	Eval06
silence duration	18.7	34.1
multiple features	18.2	32.3

Table 6. Results of System 3 using different inputs for MLLR adaptation.

	WER [%]	
adaptation using results of system	Dev07	Eval06
3 (no cross-adapt.)	17.8	33.5
2	17.3	30.9
1	17.3	30.9
1 + 2 (ROVER)	17.1	30.6

improves the recognition performance. A segmentation obtained with the method described has been used in the Arabic speech recognition system of SRI [19] for the 2007 GALE Evaluation.

The different adaptation methods are analyzed in Table 6. Adaptation using the output of the system to be adapted yields the worst results. Using the transcription produced by another system reduces the error rate. The best results are obtained when we use the combined output of two systems.

Having shown that our segmentation and cross-adaptation methods work well, we present detailed results of the whole recognition system. Table 7 lists intermediate results produced by the subsystems of the recognition system. We observe that Systems 1 and 2 produce noticeably better results than System 3, which is caused – among other things – by the missing VTLN. The rescoring with a larger language model and both adaptation methods give consistent improvements.

The word error rates obtained for the results produced by ROVER are given in Table 8. The combination of the 3 systems decreases the error rate by 2 % relative, although the systems were already combined by cross-adaptation.

Table 8 distinguishes the error rates on the different recording conditions of the evaluation corpora. Transcription of broadcast conversations (BC) is a more difficult task than the transcription of broadcast news (BN). Professional speakers with virtually no background noise appear in the BN recordings, whereas BC are recorded in a more noisy condition. The interviews in BC with non-professional speakers lead to more frequent hesitations, false starts, and also the speech rate may vary considerably. Furthermore, the presence of more speakers and overlapping speech have impact on the recognition performance.

We applied parts of the described decoding architecture also to the English EPPS task of the 2007 TC-STAR Evaluation Campaign. We used the four subsystems as described in [6]: two MFCC based systems, a MFCC and NN based system, and a gammatone system. A comparison of results obtained with and without cross-adaptation is given in Table 9. The use of cross-adaptation improves all subsystems. We compare the effects of cross-adaptation on system combination results in Table 10. The gain of system combination without cross-adaptation is 5 % relative compared to 3 % with cross-adaptation. The results obtained with cross-adaptation are still slightly better than those without it. We translate the first best result of a single system and the system combination to Spanish using a phrase-based machine translation system [20]. The differences in BLEU scores are fairly small. However, the system combination yields slightly better results. These results differ from those in [21], although different tasks and error metrics were used.

Table 7. Intermediate results of the decoding passes 1 to 3 and LM rescorings (LMR).

		pass				
corpus	system	1	2	LMR	3	LMR
Dev07	1	–	17.4	17.1	16.4	16.2
	2	20.6	17.3	17.2	16.3	16.1
	3	–	18.2	18.0	17.1	16.7
Eval06	1	–	31.7	31.5	30.3	30.1
	2	34.7	31.3	30.8	29.9	29.8
	3	–	32.3	32.1	30.6	30.4

Table 8. Results of single systems and system combination. Divided in broadcast news (BN) and broadcast conversation (BC) recordings.

		WER [%]		
corpus	system	total	BN	BC
Dev07	1	16.2	13.8	20.7
	2	16.1	14.0	20.1
	3	16.7	14.4	20.8
	ROVER	15.7	13.5	19.7
Eval06	1	30.1	25.3	34.9
	2	29.8	25.0	34.7
	3	30.4	25.6	35.5
	ROVER	29.1	25.4	33.9

7. CONCLUSION

We described the RWTH broadcast news transcription system for Arabic. The advantages of cross-adaptation and system combination in a multiple system architecture were shown on several corpora.

The ASR-based audio segmentation algorithm presented in this paper had a considerable impact on the overall recognition performance. Furthermore, we discussed techniques for the acceleration of the acoustic training by efficient parallelization and reduced memory requirements of language model rescoring of lattices.

Table 9. Results of the RWTH TC-STAR systems for the English Eval07 corpus with and without MLLR cross-adaptation (CA). All results include language model rescoring.

		pass		
corpus	system	2	3 w/o CA	3 w. CA
Eval06	MFCC	10.1	8.7	8.4
	Gammatone	10.5	9.0	8.4
	NN	11.6	9.4	8.9
	MFCC 2	9.6	8.5	8.0
Eval07	MFCC	10.5	10.1	9.6
	Gammatone	11.2	10.7	10.1
	NN	12.2	11.7	10.4
	MFCC 2	10.4	9.7	9.3

Table 10. System combination results of the RWTH TC-STAR systems for English (Eval07) with and without MLLR cross-adaptation (CA) together with results of English to Spanish machine translation.

system	CA	WER	BLEU [%]
best single	no	9.7	38.4
	yes	9.3	38.5
combination	no	9.2	38.6
	yes	9.0	38.8

8. ACKNOWLEDGEMENTS

We thank SRI International for helping to build up the Arabic recognition system, especially Dimitra Vergyri, Wen Wang, and Andreas Stolke for their close collaboration. We thank IDIAP for contributing the NN features.

9. REFERENCES

[1] K. Kirchhoff et al., "Novel approaches to Arabic speech recognition: report from the 2002 Johns-Hopkins summer workshop," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Hong Kong, China, Apr. 2003, vol. 1, pp. 344–347.

[2] "Global autonomous language exploitation (GALE)," `http://www.arpa.mil/ipto/programs/gale/index.htm`, visited at July 19 2007.

[3] A. Zolnay, R. Schlüter, and H. Ney, "Robust speech recognition using a voiced-unvoiced feature," in *Proc. Int. Conf. on Spoken Language Processing*, Denver, CO, USA, Sept. 2002, vol. 2, pp. 1065 – 1068.

[4] R. Schlüter, I. Bezrukov, H. Wagner, and H. Ney, "Gammatone features and feature combination for large vocabulary speech recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Honolulu, HI, USA, Apr. 2007, vol. 4, pp. 649–652.

[5] H. Hermansky and P. Fousek, "Multi-resolution RASTA filtering for TANDEM-based ASR," in *Proc. European Conf. on Speech Communication and Technology*, Lisbon, Portugal, Sept. 2005, pp. 361 – 164.

[6] J. Lööf et al., "The RWTH 2007 TC-STAR evaluation system for European English and Spanish," in *Proc. Int. Conf. on Speech Communication and Technology*, Antwerp, Belgium, Aug. 2007, accepted for publication.

[7] B. Hoffmeister et al., "Development of the 2007 RWTH Mandarin GALE LVCSR system," submitted to *IEEE Automatic Speech Recognition and Understanding Workshop*, Kyoto, Japan, Dec. 2007.

[8] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, no. 2, pp. 75 – 98, Apr. 1998.

[9] D. Povey and P. C. Woodland, "Minimum phone error and i-smoothing for improved discriminative training," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Orlando, FL, May 2002, vol. 1, pp. 105 – 108.

[10] R. Schlüter, B. Müller, F. Wessel, and H. Ney, "Interdependence of language models and discriminative training," in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop*, Keystone, CO, USA, Dec. 1999, vol. 1, pp. 119–122.

[11] A. Messaoudi, J.-L. Gauvain, and L. Lamel, "Arabic broadcast news transcription using a one million word vocalized vocabulary," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Toulouse, France, May 2006, vol. 1, pp. 1093–1096.

[12] F. de Vriend et. al., "LC-STAR: XML-coded phonetic lexica and bilingual corpora for speech-to-speech translation," in *Proc. of Papillon2004, Workshop on Multilingual Lexical Databases*, Grenoble, France, Aug. 2004.

[13] M. Bisani and H. Ney, "Multigram based grapheme-to-phoneme conversion for LVCSR," in *Proc. European Conf. on Speech Communication and Technology*, Geneva, Switzerland, Sept. 2003, vol. 2, pp. 933 – 936.

[14] S. E. Tranter, K. Yu, G. Evermann, and P. C. Woodland, "Generating and evaluating segmentations for automatic speech recognition of conversational telephone speech," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004, vol. 1, pp. 753–756.

[15] D. Guiliani and F. Brugnara, "Acoustic model adaptation with multiple supervisions," in *Proc. TC-Star Workshop on Speech-to-Speech Translation*, Barcelona, Spain, June 2006, pp. 151 – 154.

[16] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," in *Proc. Symposium on Operating Systems Design and Implementation*, San Francisco, CA, USA, Dec. 2004, pp. 137–150.

[17] S. Matsoukas et al., "Advances in transcription of broadcast news and conversational telephone speech within the combined EARS BBN/LIMSI system," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 5, no. 14, pp. 1541–1556, Sept. 2006.

[18] A. Stolcke, "SRILM - An extensible language modeling toolkit," in *Proc. Int. Conf. on Spoken Language Processing*, Denver, CO, USA, Sept. 2002, pp. 901–904.

[19] A. Stolcke et al., "Recent innovations in speech-to-text transcription at SRI-ICSI-UW," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1729–1744, Sept. 2006.

[20] E. Matusov, R. Zens, D. Vilar, A. Mauser, M. Popovic, and H. Ney, "The RWTH machine translation system," in *Proc. TC-STAR Workshop on Speech-to-Speech Translation*, Barcelona, Spain, June 2006, pp. 31–36.

[21] M.J.F. Gales et al., "Speech recognition system combination for machine translation," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Honolulu, HI, USA, Apr. 2007, pp. 1277–1280.