Recent Innovations in Speech-to-Text Transcription at SRI-ICSI-UW

Andreas Stolcke, *Senior Member, IEEE,* Barry Chen, Horacio Franco, *Member, IEEE,* Venkata Ramana Rao Gadde, Martin Graciarena, *Member, IEEE,* Mei-Yuh Hwang, Katrin Kirchhoff, *Member, IEEE,* Arindam Mandal, Nelson Morgan, *Fellow, IEEE,* Xin Lei, Tim Ng, Mari Ostendorf, *Fellow, IEEE,* Kemal Sönmez, *Member, IEEE,* Anand Venkataraman, Dimitra Vergyri, *Member, IEEE,* Wen Wang, *Member, IEEE,* Jing Zheng, *Member, IEEE,* Qifeng Zhu, *Member, IEEE,*

***Abstract*— We summarize recent progress in automatic speech-to-text transcription at SRI, ICSI, and the University of Washington. The work encompasses all components of speech modeling found in a state-of-the-art recognition system, from acoustic features, to acoustic modeling and adaptation, to language modeling. In the front end, we experimented with nonstandard features, including various measures of voicing, discriminative phone posterior features estimated by multilayer perceptrons, and a novel phone-level macro-averaging for cepstral normalization. Acoustic modeling was improved with combinations of front ends operating at multiple frame rates, as well as by modifications to the standard methods for discriminative Gaussian estimation. We show that acoustic adaptation can be improved by predicting the optimal regression class complexity for a given speaker. Language modeling innovations include the use of a syntax-motivated almost-parsing language model, as well as principled vocabulary-selection techniques. Finally, we address portability issues, such as the use of imperfect training transcripts, and language-specific adjustments required for recognition of Arabic and Mandarin.**

***Index Terms*— Speech-to-text, conversational telephone speech, broadcast news**

I. Introduction

ACCURATE transcription of speech into text (speech-to-text, or STT) is a prerequisite for virtually all other natural language applications operating on audio sources. Supported by the DARPA EARS program, a team of researchers at SRI International, the International Computer Science Institute (ICSI), and the University of Washington (UW) developed a system to produce "rich transcripts" from conversational telephone speech (CTS) and broadcast news (BN) sources, that is, transcripts containing not just streams of words, but also structural information corresponding to sentence boundaries and disfluencies. The methods used to recover information "beyond the words" are described in a companion paper for this special issue [1]. This article focuses on the prerequisite for rich transcription, namely, accurate word recognition. We describe a series of new techniques that were developed and tested on CTS and BN data, spanning the major trainable components of a speech recognition system: feature extraction front end, acoustic models, and language models. In the front end, we experimented with nonstandard features, including various measures of voicing, discriminative phone posterior features estimated by multilayer perceptrons, and a novel phone-level macro-averaging for cepstral normalization. Acoustic modeling was improved with combinations of front ends operating at multiple frame rates, as well as by modifications to the standard methods for discriminative Gaussian estimation. We show that acoustic adaptation can be improved by predicting the optimal regression class complexity for a given speaker. Language modeling innovations include the use of a syntax-motivated almost-parsing language model, as well as principled vocabulary selection techniques. Finally, we address portability issues, such as the use of imperfect training transcripts, and language-specific changes required for recognition of Arabic and Mandarin.

II. Recognition system

To provide the necessary background, we give a brief description of SRI's CTS recognition system, which served as the basis of most of the work described here. It is depicted in Fig. 1. An "upper" (in the figure) tier of decoding steps is based on Mel-frequency cepstral coefficient (MFCC) features;

Manuscript received October 16, 2005; revised May 30, 2006. This work was supported by the Defense Advanced Research Projects Agency (DARPA) under Contract MDA972-02-C-0038 and Grant MDA972-02-1-0024 (approved for public release, distribution unlimited). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Alex Acero.

A. Stolcke is with SRI International, Menlo Park, CA 94025 USA, and with the International Computer Science Institute, Berkeley, CA 94704 USA (e-mail: stolcke@speech.sri.com).

B. Chen was with the International Computer Science Institute, Berkeley, CA 94704 USA. He is now with the Lawrence Livermore National Laboratory, Livermore, CA 94550 USA. (e-mail: byc@icsi.berkeley.edu)

H. Franco, V. R. R. Gadde, M. Graciarena, K. Sömez, A. Venkataraman, D. Vergyri, W. Wang, and J. Zheng are with SRI International, Menlo Park, CA 94025 USA (e-mail hef@speech.sri.com; rao@speech.sri.com; martin@speech.sri.com; kemal@speech.sri.com; anand@speech.sri.com; dverg@speech.sri.com; wwang@speech.sri.com; zj@speech.sri.com).

M.-Y. Hwang, K. Kirchhoff, A. Mandal, X. Lei, and M. Ostendorff are with the Signal, Speech, and Language Interpretation (SSLI) Laboratory, Department of Electrical Engineering, University of Washington, Seattle, WA 98195-2500 USA (e-mail: mhwang@ee.washington.edu; katrin@ee.washington.edu; marindam@ssli.ee.washington.edu; leixin@ee.washington.edu; mo@ssli.ee.washington.edu).

N. Morgan is with the International Computer Science Institute, Berkeley, CA 94704 USA (e-mail: morgan@icsi.berkeley.edu).

T. Ng was with the Signal, Speech, and Language Interpretation (SSLI) Laboratory, Department of Electrical Engineering, University of Washington, Seattle, WA 98195-2500 USA. He is now with the Speech Group, BBN Technologies, Cambridge, MA 02138 USA. (e-mail: tng@ssli.ee.washington.edu)

Q. Zhu was with the International Computer Science Insitute, Berkeley, CA 94704 USA. He is now with Texas Instruments, Dallas, TX 75243 USA. (qifeng@icsi.berkeley.edu)

Digital Object Identifier 10.1109/TASL.2006.879807

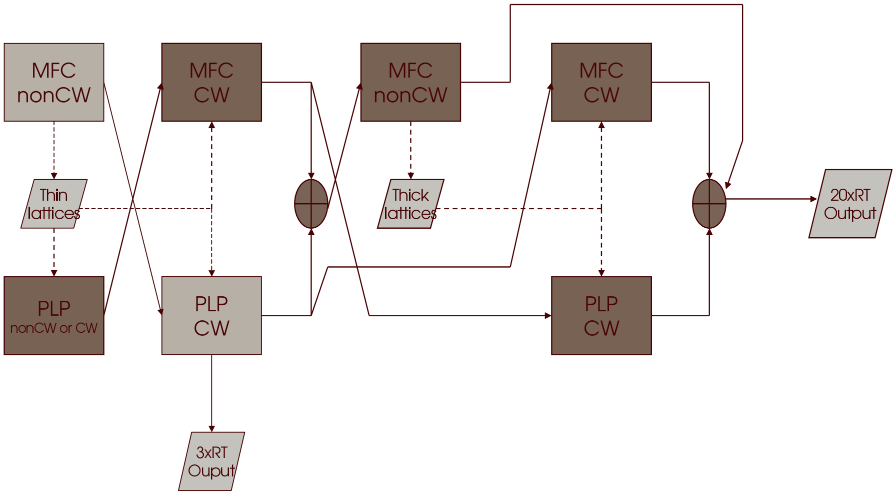


Fig. 1. SRI CTS recognition system. Rectangles represent decoding steps. Parallelograms represent decoding output (lattices or 1-best hypotheses). Solid arrows denote passing of hypotheses for adaptation or output. Dashed lines denote generation or use of word lattices for decoding. Crossed ovals denote confusion network system combination. The two decoding steps in light gray can be run by themselves to obtain a "fast" system using about 3xRT runtime.

a parallel "lower" tier of decoding steps uses perceptual linear prediction (PLP) features [2]. The outputs from these two tiers are combined twice using word confusion networks (denoted by crossed ovals in the figure). Except for the initial decodings, the acoustic models are "cross-adapted" to the output of a previous step from the respective other tier using maximum-likelihood linear regression (MLLR) [3]. The initial decoding steps in each tier also use MLLR, though with a phone-loop model as reference.

Lattices are generated initially to speed up subsequent decoding steps. The lattices are regenerated once later to improve their accuracy, after adapting to the outputs of the first combination step. The lattice generation steps use noncross-word (nonCW) triphone models, while decoding from lattices uses crossword (CW) models. The final output is the result of a three-way system combination of MFCC-nonCW, MFCC-CW, and PLP-CW models. Each box in the diagram corresponds to a complex recognition step involving a decoding run to generate either lattices or N-best lists, followed by a rescoring of these outputs with higher-order language models, duration models, and a pause language model [4].

The acoustic models used in decoding use standard normalization techniques: cepstral mean and variance normalization, vocal tract length normalization (VTLN) [5], heteroscedastic linear discriminant analysis (HLDA) [6], [7], and speaker-adaptive training based on constrained MLLR [8]. All acoustic models are trained discriminatively using the minimum phone error (MPE) criterion [9] or variants thereof (as described below). The baseline language models (LMs) are bigrams (for lattice generation), trigrams (for lattice decoding), and 4-gram LMs (for lattice and N-best rescoring). The CTS in-domain training materials are augmented with data harvested from the web, using a search engine to select data that is matched for both style and content [10].

The entire system runs in under 20 times real time (20xRT) on a 3.4 GHz Intel Xeon processor. For many scenarios it is useful to use a "fast" subset of the full system consisting of just two decoding steps (the light-shaded boxes in Fig. 1); this fast system runs in 3xRT and exercises all the key elements of the full system except for confusion network combination. The baseline system structure is the result of a heuristic optimization (which took place over several years) that aims to obtain maximal benefit from system combination and cross-adaptation, while staying within the 20xRT runtime constraint imposed by the DARPA CTS STT evaluation.

For BN recognition the system was further simplified to run in under 10xRT. In this case only two recognition stages are used (nonCW and CW), and both are based on a PLP front end. Final LM rescoring uses 5-gram LMs.

III. Feature Extraction Front End

A. Voicing Features

Our first strategy to improve the front end is to augment the cepstral feature representation with phonetic features computed using independent front ends. The parameters from each front end specific to a phonetic feature are optimized to improve recognition accuracy. While this is a general framework for multiple phonetic features, our present approach explores the use of just voicing features, since voicing is highly relevant for phone discrimination.

The first voicing feature used in this paper is the traditional normalized peak autocorrelation. The second voicing feature used is a newly defined entropy of the high-order cepstrum. For the time-windowed signal $x(t)$ of duration T the high-order cepstrum is defined as

$$C = \mathrm{IDFT}(\log(|\mathrm{DFT}(w(t) \cdot x(t))|^2)) \quad (1)$$

where $w(t)$ is the Hamming window of duration T. Zero padding is used prior to the computation of the DFT. The entropy of the high-order cepstrum is computed as follows:

$$H(C) = -\sum_r P(C(r)) \log(P(C(r))) \quad (2)$$

$$P(C(r)) = \frac{C(r)}{\sum_{r'} C(r')} \quad (3)$$

where the indices r and r' correspond to a pitch region from 80 Hz to 450 Hz. For robust voicing detection, both voicing features are used together, since they have complementary strengths at different pitch values [11]

We explored several alternatives for integrating the voicing features into the CTS recognition system [11]. In an initial system, we first concatenated the voicing features with the standard Mel cepstral features and optimized the temporal window duration for the voicing feature front end. We extended the window duration beyond the traditional 25 ms, and found that the voicing activity was captured more reliably with a longer time span.

We then explored the integration of the voicing features in a more complex recognition system with Mel cepstral features augmented with third differential features, reducing the dimensionality with HLDA. Different integration approaches were evaluated in this system, revealing the usefulness of a multiframe window of voicing features. We found a five-frame window to be optimal in recognition.

A two-stage system similar to the "fast" version of the system in Fig. 1 was designed to evaluate the effect of voicing features on word error rate (WER). Both stages used nonCW gender-dependent triphone models trained with maximum likelihood on approximately 400 h of the Switchboard, CallHome English, and Switchboard Cellular databases. We then tested this system on the NIST RT-02 CTS database, which contains approximately five hours of speech from 120 speakers. The WER results after key decoding steps pass, with and without voicing features, are presented in Table I. We see that voicing features give a relative gain of around 2%, and a similar gain is preserved after rescoring and MLLR.

In another experiment, we used the complete CTS evaluation system and tested the effect of voicing features just prior to the final N-best rescoring stage. The acoustic models in this case are CW-word triphone models trained with maximum mutual information estimation (MMIE). The relative WER reduction using the voicing features, from 25.6% to 25.1%, was again around 2%.

B. Discriminative Features Estimated by Multilayer Perceptrons

Many researchers have found that incorporating certain types of acoustic information from longer time ranges than the typical short-time (25 ms or so) analysis window can be helpful for automatic speech recognition; examples include cepstral mean subtraction [12] or RASTA [13]. It was shown that the incorporation of long-time (as long as 1 s) acoustic information directly as observed features for a hidden Markov model (HMM) could lead to substantial improvements on speech recognition for small vocabulary tasks [14], where the multilayer perceptron (MLP) was the key workhorse methodology, performing nonlinear discriminant transformations of the time-frequency plane. In the work reported here, we extended these approaches to large-vocabulary speech recognition. Here we found that we could also achieve substantial gains with these features, despite potential overlap with other techniques incorporated in the full system, such as linear discriminative transformations and discriminative training of the HMMs.

Our intent was to do system combination at the feature level (in addition to another system combination being done at the word level later in the process). We have found that, in contrast to the requirements for high-level system combination, feature combination can benefit from comparatively weak components if they are sufficiently complementary. For the purpose of this system, then, we incorporated three feature components:

1) a temporally oriented set of long-time (500 ms) MLP-based features, derived from log critical band energies;
2) a set of moderate-time (100 ms) MLP-based features, derived from 9 frames of PLP cepstra and 2 derivatives;
3) PLP or MFCC features and 3 derivatives, transformed and dimensionally reduced by HLDA.

The first two features were combined at the level of the MLP outputs, which could be interpreted as posteriors due to their training with 1/0 targets indicating the phonetic labels that were obtained from a previous forced alignment [15]. The posteriors were combined additively, with weights derived from the inverse of the entropy function computed from the posteriors themselves; thus, MLP "decisions" with strong certainty were more heavily weighted [16]. During our development, we found this method to be roughly comparable to more straightforward approaches (e.g., summing weighted log probabilities with empirically determined weights), but it was both automatic and more reliable in cases where one of the feature streams was badly damaged. The logs of the combined posteriors were then processed with principal component analysis (PCA) for orthogonalization and dimensionality reduction, and then appended to the more traditional PLP or MFCC features.

The 500 ms features used techniques based on the original development called TempoRal Patterns (TRAPs) [14]. In the original approach, MLPs were trained on log critical energies with phonetic targets, and then further combined with a larger MLP that was also trained on phonetic targets. In the variant developed for our task, we used the critical band training to derive input-to-hidden weights for the MLPs, and then combined the hidden layer outputs with the broadband MLP as before. We named this modified version of the features Hidden Activation TRAPs, or HATs. The motivation for this modification was that the individual critical bands were only marginally effective in 46-category phonetic discrimination; on the other hand, we noted that the input-hidden connections appeared to be learning common temporal patterns as part of each full MLP's attempt to discriminate between the phones [17]. We further determined that the critical band networks could work well with relatively small hidden layers (40 to 60 hidden units), while the combining networks benefited

TABLE I

RECOGNITION WERS WITH AND WITHOUT VOICING FEATURES, TESTED ON EVAL2002 CTS TEST SET. RELATIVE PERCENTAGE REDUCTIONS ARE GIVEN IN PARENTHESES.

Step	No voicing features	With voicing features
Phone-loop adapted, bigram LM	38.6%	37.8% (-2.1%)
4-gram and duration rescored	33.6%	32.5% (-3.3%)
MLLR to first recognition output	30.6%	30.0% (-2.0%)

TABLE II

RESULTS WITH MLP FEATURES ON RT-04F CTS DEVELOPMENT AND EVALUATION SETS

	RT-04F Dev			RT-04F Eval		
System	Male	Female	All	Male	Female	All
Baseline	18.1	16.2	17.2	20.2	20.4	20.3
w/MLP feats.	16.8	14.2	15.5	19.0	17.7	18.3
Rel. change (%)	-7.2	-12.3	-9.9	-5.9	-13.2	-9.9

from a large number of parameters, particularly for the task incorporating 2000 h of speech.

For the larger task, four full-band MLPs needed to be trained: for each gender, there was a HATs combining network, and a 9-frame network incorporating PLP-based inputs. Each of these nets had roughly 8 million weights (with a large hidden layer, typically 10,000 to 20,000 units wide, where each hidden unit incorporated a sigmoidal nonlinearity). Given the large number (360 million) of frames used for each training, the computational load presented a huge practical problem. Straightforward extrapolation from previous smaller experiments suggested a runtime of 18 months for the full training. This results from an almost quadratic complexity, since it requires a roughly linear growth in the number of parameters to optimally benefit from the increase in the size of the training set. The complexity is not quite quadratic because there is a modest decrease in the number of training epochs required for convergence (as assessed on an independent cross-validation set). Fortunately, we were able to significantly reduce training time by a combination of software speedup and algorithm heuristics. The most important heuristic was to do early training epochs with fewer patterns and fairly large learning rates. The learning rates were gradually decreased, while the amount of data used was gradually increased.

After solving the practical issues with MLP training, we tested the features in the full CTS system. We investigated various options for augmenting the various models used by the system with MLP features. A detailed account of these investigations can be found in [18]. The outcome was that the best strategy is to add MLP features only to the MFCC-based models used in the architecture depicted in Fig. 1, and to leave the PLP-based stages unchanged. This makes sense in that the resulting subsystems are more differentiated, and therefore give better results upon combination.

Table II summarizes the results with and without MLP features on the RT-04F CTS development and evaluation test sets. (The system including MLP features represented the official submission by the SRI/ICSI/UW team to the RT-04F CTS evaluations.) The overall relative WER reduction on both testsets is identical, 9.9% (2.0% absolute on the evaluation set). However, we also observe that the improvement is almost twice as big for female speakers as for males. This imbalance needs further investigation and points to a possible improvement of the system (by improving accuracy specifically on male speakers).

C. Macro Normalization of Features

Current recognition systems perform a variety of feature normalizations to reduce the variations in speech due to differences in channel, speaker, and environment. Examples of these normalizations include feature mean removal [19], feature variance normalization [20], vocal tract length normalization [5], and model-based normalizations like HLDA [6], [7] and speaker-adaptive training (SAT) [8]. While these normalization techniques produce significant gains in recognition accuracy, they suffer from one weakness: the estimates of the normalization parameters are affected by the relative distribution of the speech frames among phone classes. To redress this, we investigated a new approach to feature normalization that estimates the parameters independent of the distribution of the speech among the phone classes.

1) Algorithm: Consider an utterance from a speaker represented by a sequence of feature vectors x_i. Let N be the total number of such features extracted from the utterance(s). Assume that we have an alignment for the utterance associating the features and phone classes (produced by a first recognition pass). Let C be the number of phone classes. We then estimate the feature mean and variance (for the purposes of normalization) as

$$\begin{aligned} \mu_c &= \tfrac{1}{N_c}\sum_{i=1}^{N_c} x_{ic}; & \sigma_c{}^2 &= \tfrac{1}{N_c}\sum_{i=1}^{N_c}(x_{ic}-\mu_c)^2 \\ \mu &= \tfrac{1}{C}\sum_{c=1}^{C}\mu_c; & \sigma^2 &= \tfrac{1}{C}\sum_{c=1}^{C}(\sigma_c{}^2+(\mu_c-\mu)^2). \end{aligned}$$

It can be seen that our technique results in estimates that are independent of the distribution of the features among the different phone classes. The traditional (micro-averaging) estimation corresponds to the case of a single class.

One problem with our technique is that if a class has no frames assigned to it, it is not used in the estimation. In other words, the estimates are not robust to missing classes (classes with no frames). To overcome this, we compute global estimates on the training data for each class and use these estimates to smooth the speaker-level estimates for the same class. We use linear interpolation to smooth the speaker estimates with the global estimate for the same class.

TABLE III
WER RESULTS WITH AND WITHOUT MACRO-NORMALIZATION

Step	Word Error Rate		
	Baseline	Macro-normed models	
		Unsmoothed	Smoothed
Unadapted Recog	34.3	33.7	33.5
HYP Adapt+Recog	32.2	31.7	31.9

2) Experiments: To evaluate the performance of our new feature mean/variance normalization technique, we performed speech recognition experiments using the SPINE (Speech in Noisy Environments) databases and the Fisher CTS corpus available from the Linguistic Data Consortium (LDC). The SPINE corpus experiments were focused on finding the optimal phone classes to use for estimating the normalization parameters. To find the optimal phone classes, we computed feature mean and variance normalizations using 1, 2, 3, 7, 11, and 47 phone classes. We then trained acoustic models and tested on the SPINE 2001 evaluation set. We used the same setup as used in the first recognition step in our 2001 SPINE evaluation system [21]. For both training and testing, we used the same number of phone classes. We observed that the best performance improvements were obtained for the 47-phone class case (which corresponds to a single phone per class).

The Fisher corpus is substantially larger than the SPINE corpus and has over 2000 h of acoustic CTS data. To train acoustic models, we used a subset of approximately 180 h of the data containing about 203 K utterances for 2589 male speakers. For testing, we used the male portion of the RT-04F development testset, containing 1474 utterances from 37 speakers. MFCC features augmented with voicing features (as described above) and reduced by HLDA to 39 dimensions were used. A bigram LM was used in recognition.

3) Results: Table III compares the baseline, macro-normed, and smoothed macro-normed models. We show the results for the first recognition pass with unadapted models, and recognition results after adaptation with MLLR. For all models, the recognition hypotheses from the first recognition pass with baseline models were used as adaptation references. We observe that our normalization technique outperforms the baseline before and after adaptation. We find that the improvement from smoothing with global estimates is small. This may be because the Fisher corpus has more data per speaker, resulting in less smoothing. We also find that adaptation reduced the performance gains from macro-normalization.

The results of our experiments show that our new approach based on estimating feature normalization parameters from macro-averages results in a reduction in WER. Smoothing the class estimates for a speaker with global estimates for that class reduces the WER further. We also find that this WER reduction drops significantly after adaptation. As current speech recognition systems employ multiple adaptation steps, one can argue that the technique may produce only marginal improvement in the overall system performance. We believe that the performance loss occurs primarily because the adaptation algorithms rely on statistics that weight all frames equally (micro statistics), thereby negating the compensation done in our feature mean and variance estimation (using macro statistics). In line with our approach, we are modifying the adaptation algorithms to utilize macro statistics (instead of micro statistics). Preliminary results with macro-normalized HLDA support this.

IV. ACOUSTIC MODELING

A. Multirate Recognition Models for Phone-Class-Dependent N-best List Rescoring

Modeling of speech with fixed-rate front ends and HMMs implies a constant rate of information accumulation. In this framework, frames of a fixed length are scored uniformly to compute the likelihood that a given sequence of feature vectors is produced by the model. The common fixed frame length of 25 to 30 ms represents a fundamental time-frequency tradeoff in the speech representation. For example, vowels can result in a relatively stationary harmonic structure that can be sustained for hundreds of milliseconds, whereas stop consonants can have landmark transients that last no more than 10 ms. In a constant frame length front end, transient phenomena are blended with the context, decreasing the sharpness of the models that account for information-bearing discontinuities. Therefore, frame scores with particularly relevant information, such as those of stops, are washed out in the statistics of phone scores that span many more frames, such as those of vowels. Incorporation of information from acoustical phenomena taking place at different rates has received significant attention in the speech recognition literature; a brief overview can be found in [22].

We present a method that aims to incorporate information from multiple time-frequency tradeoffs by projecting the variable frame problem at the front end to the back end through rescoring of N-best lists generated by a fixed-rate recognizer with a normalized rate-dependent score. In our approach, the hypotheses generated by the fixed-rate recognition engine are in effect used to parse the incoming speech into phones, which subsequently determine the most likely rate model through the definition of a mapping from phone classes to the available set of multiple rate models. The final scores are obtained through rescoring of N-best lists by phone-dependent multiple rate model scores, a common way of incorporating other information sources. The scoring has two important aspects that differentiate our approach: (i) the normalization with respect to dynamic range of scores of models at different rates, which is carried out by normalizing with the likelihood of all the phones in the same phone context at the same resolution, and (ii) averaging of the frame-level scores to produce a single score for each phone state in the hypotheses. Resulting phone-class-dependent scores are treated as knowledge sources and combined into a linear model, parameters of which are optimized to minimize the WER.

1) Approach: Our technique involves choosing a small set of rates and training acoustic models at those rates. After generating N-best lists using a standard rate model, we score the N-best hypotheses using different rate models and combine the scores to minimize the WER.

The likelihoods computed using different acoustic models cannot be combined, as they use different features. The solution we propose is to use a normalized phone class likelihood ratio for frame-level scores. Specifically, the normalized score for feature vector x_i at triphone-state $(p_{-1}pp_{+1})$ is computed by

$$\hat{S}(x_i, p) = \log \frac{P(x_i|p_{-1}pp_{+1})}{\sum_k P(x_i|p_{-1}p_kp_{+1})} \tag{4}$$

where p represents the center phone and p_{-1} and p_{+1} represent the preceding and succeeding phone contexts. Given the normalization in Eq. 4, each frame score can now be regarded as independent of the rate of the model by which it was generated. With the normalized scores, we compute sentence-level scores for each phone class, P_k, using Eq. 5.

$$\tilde{S}(P_k) = \sum_i \hat{S}(x_i, p) I[p \in P_k]. \tag{5}$$

Finally, we combine the phone-class dependent scores with the baseline acoustic and language model scores through a linear combiner and optimize the linear combination weights to directly minimize the WER. Details may be found in [22].

2) Experiments: In our experiments, we used the three models:

1) a baseline model at standard rate (100 fps, 25.6-ms window)l
2) a slower rate model at $2/3$ of the baseline (15 ms shift, 38.4-ms window);
3) a faster rate model at twice the baseline (5 ms shift, 12.8-ms window).

The acoustic training data were the male subset of our RT-02 CTS training set (about 140 h). The features were 13 MFCCs (including C_0) and their first and second time derivatives. We trained nonCW triphone models containing 2400 genones (state clusters) with 64 Gaussians per genone. As in the first stage of our full system, the models were adapted to a phone-loop (using MLLR) before recognition.

3) Results: We used the NIST RT-02 male testset (3083 utterances) in our experiments. This set was partitioned into two parts; a tuning set containing about 1400 utterances and a heldout set containing the rest. The tuning set was used to optimize the weights for different scores. These weights were then applied to the heldout set and the WER was computed. Table IV shows the results from different model combinations.

The results on RT-02 confirm that our rescoring approach results in a significant reduction in the WER, with the best reduction of 1.0% absolute for the slower rate model. Increasing the number of classes reduces the WER for the tuning set but not on the heldout set. For finer phone sets, it seems that we may need a larger tuning set to properly estimate the weights.

B. Improved Discriminative Training

Discriminative training criteria, such as maximum mutual information (MMI) [23] and minimum phone error (MPE) [24], have shown great advantage over the traditional maximum likelihood (ML) training in large-vocabulary speech recognition. Our contribution to this work addresses the challenge brought by vast amounts of training data, and to obtain accuracy gains over the standard MPE and MMI training.

TABLE IV
RESULTS WITH MULTIRATE PHONE-CLASS-BASED RESCORING ON THE NIST RT-02 CTS TEST SET

Model	WER(%) for #Phone Class					
	1		3		7	
	Tune	Held	Tune	Held	Tune	Held
Baseline	39.9	39.9	39.9	39.9	39.9	39.9
+2.0 rate	39.2	39.4	38.8	39.1	38.6	39.2
+0.67 rate	39.3	39.1	38.9	39.0	38.9	38.9
+0.67 +2.0	39.3	39.4	38.7	39.1	38.6	38.9

1) Phone-Lattice-Based Discriminative Training: The standard MMI and MPE training procedures use word lattices to represent competing hypotheses. Phone boundaries are marked in the word hypotheses to constrain and speed up search during training. Alternatively, in an *implicit-lattice* method for MMI training [25], lattices are generated on the fly by decoding a highly compact decoding graph compiled from a pronunciation dictionary. This approach can save a lot of disk space for lattice storage at the cost of increased computation.

In this work, we aim to speed up both the lattice generation and statistics collection procedures, and therefore propose phone-lattice-based discriminative training, which is applicable to both MMI and MPE. Similar to implicit-lattice MMI, we compile all dictionary pronunciations into a determinized and minimized [26] finite-state phone network, with both pronunciation and unigram language model probabilities embedded. Using this finite-state network, we generate phone lattices in a very fast decoding pass, using an algorithm similar to that described in [27]. In a phone lattice, each real arc represents a phone, with start and end time information. Null arcs are introduced to reduce the number of real arcs, and thus size and computation. With a forward-backward search pass constrained by the timing of the phone arcs, statistics for both MMI and MPE can be collected in the standard manner [9]. Compared to word lattices, phone lattices are much faster to generate, as the decoding graph is much more compact without word information. Phone lattices are also more efficient in representing hypotheses with different phones (which is all that matters in phone-based HMM training), and as a result need less storage space.

Based on the phone lattices, we can easily apply both MMI and MPE training. Recent research showed that I-smoothing with different prior models can help boost the effectiveness of discriminative training [9]. We found that alternating MMI and MPE criteria during discriminative training can help to reach the best model accuracy with the least number of training iterations. For odd-numbered iterations, estimate the MPE model with MMI prior; for even-numbered iterations, estimate the MMI model with MPE prior. The prior models themselves are I-smoothed with ML models. We call this approach MPE+MMI.

2) Minimum Phone Frame Error (MPFE) Criterion: In the standard MPE criterion, the correctness measure,

TABLE V
EFFECT OF DIFFERENT DISCRIMINATIVE TRAINING CRITERIA ON VARIOUS CTS EVALUATION TEST SETS (WER%)

	RT-04F (Eval)	RT-03	RT-02	Hub-5 2001
MLE	24.5	25.3	26.7	26.1
MPE+MMI	22.6	23.7	24.9	24.4
MPFE+MMI	22.4	23.1	24.5	24.0

$PhoneAcc(q)$, of a phone hypothesis q is defined as

$$PhoneAcc(q) = \begin{cases} -1 + 2e & \text{if } q \text{ is correct in label} \\ -1 + e & \text{otherwise} \end{cases} \quad (6)$$

where e is the overlap ratio between q and its corresponding phone in the reference transcription. We propose a different phone accuracy definition $PhoneFrameAcc(q)$:

$$PhoneFrameAcc(q) = \sum_{t=start(q)}^{end(q)} P(S_t \in S(q)|W, O) \quad (7)$$

where q is the phone hypothesis under study; $S(q)$ denotes the set of HMM states associated with this phone; $start(q)$ and $end(q)$ represent the start and end times of q in frame units; $P(S_t \in S(q)|W, O)$ is the posterior probability of the HMM state belonging to $S(q)$ at time t given observations O and transcription W, which can be obtained with the standard forward-backward algorithm that is widely used in HMM training.

Substituting the $PhoneFrameAcc(q)$ for $PhoneAcc(q)$ in the MPE criterion, we obtain the MPFE criterion. Because of the similarity in definition, MPFE can use the same algorithm as MPE except for the difference of measuring the hypothesis accuracy. Since all the competing hypotheses in a lattice have the same number of frames, MPFE does not have a systematic bias favoring deletion error. We also observed that the MPFE occupancies have values similar to those of MMI occupancies. This may make MPFE more robust than MPE when dealing with a small amount of data.

Table V compares two English CTS crossword gender-dependent models trained on about 1400 h of Switchboard and Fisher data, with MPE+MMI and MPFE+MMI, respectively. A 39-dimensional feature vector obtained from MFCCs and voicing features by HLDA projection and SAT transformation was used for training. A bigram language model was first used to generate, and then a 4-gram language model to rescore lattices. Final hypotheses were generated from consensus decoding [28], [29]. LM weight, word penalties, and so on were optimized in the RT-04F development test set, and applied to the NIST 2001 Hub-5, RT-02, RT-03, and RT-04F evaluation test sets. As can be seen, the MPFE+MMI approach has a small but consistent advantage over MPE+MMI on different test sets, ranging from 0.2% to 0.6% absolute.

C. Speaker-Dependent Variation of Adaptation

Next we describe an automatic procedure for online complexity control of the acoustic adaptation parameters. The idea is to choose the best number of MLLR regression classes to use for a speaker by using speaker-level acoustic and recognizer attributes [30]. This procedure improved system performance compared to the popular approach where only the amount of adaptation data is used to control adaptation complexity [31], [3], [32].

TABLE VI
RESULTS (WER) FOR SPEAKER-DEPENDENT MLLR REGRESSION CLASS PREDICTION, ON NIST RT-04F CTS EVALUATION TEST SET

Default	Rec. independent	Rec. dependent	All	Oracle
18.6	18.3	18.2	18.3	17.4

The motivation for this approach is based on analysis of system performance in offline experiments for six different sizes of regression class trees that included the possibility of using unadapted speaker-independent models. By choosing the oracle regression class tree size for each speaker, we observed that for the recent NIST CTS test sets (from 1998 through 2004), we could achieve a 1% absolute improvement in WER on average.

1) Prediction of Tree Sizes: To capitalize on this observation, we developed an automatic procedure that classified each speaker into one of six possible regression class tree sizes using standard statistical learners that were trained on acoustic and speaker-level information observed in adaptation data. Speaker-level features that were investigated include both recognizer-independent features (seconds of adaptation speech, VTLN factor, a normalized energy measure, and rate of speech) and features that would depend on the recognition output (pre-adaptation acoustic scores and average word-based confidence scores). Using an n-fold cross-validation training paradigm and these speaker-level features, an ensemble of classifiers was designed and combined by averaging their class posteriors to form a stacked learner. Decision trees were found to perform best, though not significantly better compared to support vector machines, k-nearest neighbor and multinomial neural network classifiers. The overall classification error obtained was in the range of 55% to 64%.

2) Recognition Experiments: Various system configurations and feature subset combinations were evaluated in predicting the regression class tree size for individual speakers in the NIST RT-04F test set. The main results, using the full CTS recognition system, are shown in Table VI, with more detailed results and analysis in [30]. Recognizer-dependent and recognizer-independent feature subsets gave similar performance gains, but no additional gain was observed by combining them. The result for the oracle case shows that there is still much room for improvement.

V. LANGUAGE MODELING

A. SuperARV LM

Structured language models (LMs) have recently been shown to give significant improvements in large-vocabulary recognition relative to traditional word N-gram models. In [33], we developed an almost-parsing language model based on the constraint dependency grammar (CDG) formalism.

Here we summarize the basics of the LM and the approach we developed for adapting and applying the LM in SRI's English BN and CTS STT systems.

The SuperARV LM [33] is a highly lexicalized probabilistic LM based on syntactic information expressed in CDGs [34]. It tightly integrates multiple knowledge sources, such as word identity, morphological features, lexical features that have synergy with syntactic analyses (e.g., gap, mood), and syntactic and semantic constraints at both the *knowledge representation level* and *model level*.

At the knowledge representation level, integration was achieved by introducing a linguistic structure, called a super abstract role value (*SuperARV*), to encode multiple knowledge sources in a uniform representation that is much more fine grained than parts-of-speech (POS). A SuperARV is an abstraction of the joint assignment of dependencies for a word, which provides a mechanism for lexicalizing CDG parse rules. A SuperARV is formally defined as a four-tuple for a word, $\langle C, F, (R, L, UC, MC)+, DC\rangle$, where C is the lexical category of the word, $F = \{Fname_1 = Fvalue_1, \ldots, Fname_f = Fvalue_f\}$ is a feature vector ($Fname_i$ is the name of a feature and $Fvalue_i$ is its corresponding value), $(R, L, UC, MC)+$ is a list of one or more four-tuples, each representing an abstraction of a role value assignment, where R is a role variable (e.g., governor), L is a functionality label (e.g., np), UC represents the relative position relation of a word and its dependent (i.e., modifiee), MC is the lexical category of the modifiee for this dependency relation, and DC represents the relative ordering of the positions of a word and all of its modifiees. Hence, the SuperARV structure for a word provides an explicit way to combine information about its lexical features with one consistent set of dependency links for the word that can be directly derived from its parse assignments, providing admissibility constraints on syntactic and lexical environments in which a word may be used.

Model-level integration was accomplished by jointly estimating the probabilities of a sequence of words w_1^N and their SuperARV memberships t_1^N:

$$\begin{aligned} P(w_1^N t_1^N) &= \prod_{i=1}^{N} P(w_i t_i | w_1^{i-1} t_1^{i-1}) \\ &= \prod_{i=1}^{N} P(t_i | w_1^{i-1} t_1^{i-1}) \cdot P(w_i | w_1^{i-1} t_1^{i}). \end{aligned}$$

Note that the SuperARV LM is fundamentally a class-based LM using SuperARVs as classes. Because the SuperARV LM models the joint distribution of classes and words, data sparseness is an important issue just as for standard word N-gram LMs. In [35], we evaluated several smoothing algorithms and how to interpolate with, or backoff to, lower-order N-gram probability estimates using a combination of heuristics and mutual information criteria to globally determine the lower-order N-grams to include in the interpolation, as well as their ordering [35].

In the process of adapting the SuperARV LM technique originally developed on newswire text to the BN and CTS STT tasks, we explored three major research issues. First, the SuperARV LM must be trained on a corpus of CDG parses. Due to the lack of large CDG treebanks, we have developed a methodology to automatically transform context-free grammar (CFG) constituent bracketing into CDG annotations [35]. In addition to generating dependency structures by headword percolation, our transformer utilizes rule-based methods to determine lexical features and need role values for the words in a parse. Although these procedures are effective, they cannot guarantee that the CDG annotations generated are completely correct. In [36], the impact of errorful training data on the SuperARV LM was investigated on the Hub4 BN STT task. Two state-of-the-art parsers were chosen based on accuracy, robustness, and mutual consistency to generate CFG parses [36]. The resulting CFG treebank was then transformed to CDG parses for training the SuperARV LM. We found that the SuperARV LM was effective even when trained on inconsistent and errorful training data. In spite of these results, we improved the CFG parser accuracy and investigated the effect that had on SuperARV LM performance.

The second research issue for SuperARVs was the tradeoff between generality and selectivity. To this end, we investigated the effect of tightening the constraints by adding lexical features related to the dependents, so-called "modifiee constraints" (which are in addition to the SuperARV structure shown in [33]).

The third research issue is the implementation of an efficient approach for integrating the SuperARV LM into SRI's multipass BN and CTS decoding systems. Full evaluation of the SuperARV would be computationally expensive and difficult for lattice rescoring purposes, given that a dynamic programming algorithm has to be carried out from the start of the sentence.

We therefore opted for an approximation whereby SuperARV probabilities are only computed over the span of a standard N-gram LM, and the conditional SuperARV probabilities for a limited set of N-grams are encoded in a standard LM, and used in the standard manner in lattice and N-best rescoring. For this approximation to be effective the selection of the N-gram set is critical. Initially, we experimented with a static selection method, whereby a large number of N-grams selected based on training data is included in the approximate SuperARV N-gram LM [37]. However, we found that this approach does not always generalize well; for example, we found that it worked quite well on the RT-04F CTS development test set, but gave no improvement on the evaluation set. Consequently, we adopted a dynamic N-gram selection method for the results reported here. After generating the first set of lattices (see Fig. 1), N-grams with the highest posterior expected counts are extracted, and an approximate SuperARV N-gram LM is constructed specifically for the given test set. That LM is then used just as a standard LM in all subsequent decoding steps.

Tables VII and VIII show recognition results with dynamically approximated SuperARV LMs on CTS and BN data, respectively. The baseline results correspond to a standard backoff LM with modified Kneser-Ney smoothing [38]. For CTS (Table VII) the effects of improved parse quality and added modifiee constraints are also demonstrated. Both factors translate into WER reductions, and are partly additive. Overall, we observe between 3% and 8% relative error reduction with dynamic SuperARV approximation.[1]

B. Vocabulary Selection

The vocabulary of a speech recognition system is a significant factor in determining its performance. We thus investigated the problem of finding an optimal way to select the *right* words. The problem can be briefly summarized as follows. We wish to estimate the true vocabulary counts of a partially visible corpus of in-domain text (which we call the heldout set) when a number of other fully visible corpora, possibly from different domains, are available on which to train. The reason for learning the in-domain counts x_i of words w_i is that the words may be ranked in order of priority, enabling us to plot a curve relating a given vocabulary size to its out-of-vocabulary (OOV) rate on the heldout corpus. Therefore, it is sufficient to learn some monotonic function of x_i in place of the actual x_i. Let x_i be some function Φ of the known counts $n_{i,j}$ of words w_i, for $1 \leq j \leq m$ for each of m corpora. Then, the problem can be restated as one of learning Φ from a set of examples where

$$x_i = \Phi(n_{i,1}, \cdots, n_{i,m}).$$

For simplicity, let Φ be a linear function of the $n_{i,j}$ and independent of the particular word, w_i. Then, we can write

$$\Phi(n_{i,1}, \cdots, n_{i,m}) = \sum_j \lambda_j n_{ij}. \tag{8}$$

The problem transforms into one of learning the λ_j. Our investigations showed that a maximum likelihood based count estimation procedure was optimal in terms of selecting the best vocabulary for a domain given limited visibility into its test corpora. In ML count estimation, we simply interpret the normalized counts n_{ij} as probability estimates of w_i given corpus j and the λ_j as mixture coefficients for a linear interpolation. We try to choose the λ_j that maximize the probability of the in-domain corpus. The iterative procedure used to compute the λ_j is shown below.

$$\lambda_j \leftarrow \frac{1}{m} \tag{9}$$

$$\lambda'_j \leftarrow \frac{\lambda_j \prod_{i=1}^{|V|} P(w_i|j)^{C(w_i)}}{\sum_k \lambda_k \prod_{i=1}^{|V|} P(w_i|k)^{C(w_i)}} \tag{10}$$

$$\delta \leftarrow \lambda'_j - \lambda_j \tag{11}$$

$$\lambda_j \leftarrow \lambda'_j \tag{12}$$

$$\text{Repeat from (10) if } \delta > \text{some threshold.} \tag{13}$$

The λ_j are reestimated at each iteration until a convergence criterion determined by some threshold of incremental change is met. The likelihood of the heldout corpus increases monotonically until a local minimum has been reached. The iterative procedure is effective in rapidly computing the values of the λ_j.

We compared the ML vocabulary selection technique against the baseline of counting each word occurrence equally regardless of source, as well as several other more sophisticated techniques, as described in [39]. We evaluated the OOV rates on heldout BN and CTS data sets as a function of vocabulary size. A pilot study was conducted on the English BN task, where a small amount of hand-corrected closed-captioned data, amounting to just under 3 h (about 25,000 words), drawn from six half-hour broadcast news segments from January 2001, was used as the *partially visible* heldout data to estimate the two mixture weights λ_1 and λ_2. This heldout data is part of the corpus released by the LDC for the DARPA-sponsored English topic detection and tracking (TDT-4) task. There are two distinct corpora for training: an 18.5-million-word corpus of English newswire data covering the period of July 1994 through July 1995, and a 2.5-million-word corpus of closed captioned transcripts from the period of November through December 2000 from segments of the TDT-4 dataset released by the LDC, a closer match to the target domain. On testing, we found that for small vocabularies there exist obvious differences in the performance of a number of different vocabulary selection methods including the one introduced herein. But for large vocabularies, all methods yield about the same OOV rates.

On the English CTS task, we conducted a full evaluation, by using all the available LM training data resources and an unseen heldout data set, the RT-04F English CTS development test set. We observed that the ML method outperformed all other methods with a prominent margin, for vocabularies of size 1,000 to 90,000 words.

VI. Portability Issues

A. Flexible Alignment for Broadcast News

The majority of the English BN acoustic training data consisted of quickly transcribed (QT) speech. QT speech, typically closed captioned data from television broadcasts, usually has a significant number of deletions and misspellings, and has a characteristic absence of disfluencies such as filled pauses (such as *um* and *uh*). Errors of these kinds lead to inaccurate or failed alignments. At best, the erroneous utterance is discarded and does not benefit the training procedure. At worst, it could misalign and end up sabotaging the models. We developed a procedure called flexible alignment that aims to *cleanse* quick transcriptions so that they align better with the acoustic evidence and thus yield better acoustic models.

Our approach is characterized by a rapid alignment of the acoustic signal to specially designed word lattices that allow for the possibility of either skipping erroneously transcribed or untranscribed words in either the transcript or the acoustic signal, and/or the insertion of an optional disfluency before the onset of every word. During the flexible alignment, we process every transcript to generate a hypothesis search graph that has the following properties: every word is made optional; every word is preceded by either an optional *garbage* word, which we call the @reject@ word, or one of a certain number of

[1] The mechanics of the parse processing, SuperARV extraction, model training and evaluation are quite complex and nontrivial to reproduce. We have therefore made a software toolkit and documentation for these steps available for download; see http://www.speech.sri.com/people/wwang/html/software.html.

TABLE VII

WER (%) ON THE RT-04F CTS DEVELOPMENT AND EVALUATION TEST SETS FOR BASELINE SYSTEM AND DYNAMICALLY APPROXIMATED SUPERARV LMS. THE VALUES IN PARENTHESES ARE THE ABSOLUTE WER REDUCTIONS OVER THE BASELINE.

	WER (%)(absolute reduction)			
	dev04		eval04	
SARV factors	Baseline	SARV	Baseline	SARV
standard	15.5	14.6 (-0.9)	18.4	17.8 (-0.6)
+ better CFG trees	-	14.5 (-1.0)	-	17.6 (-0.8)
+ modifiee constraints	-	14.6 (-0.9)	-	17.8 (-0.6)
+ better CFG trees + modifiee constraints	-	14.3 (-1.2)	-	17.5 (-0.9)

TABLE VIII

WER (%) ON THE RT-04F BN DEV04, LDC-DEV04, AND EVAL04 TEST SETS FOR BASELINE AND DYNAMICALLY APPROXIMATED SUPERARV LMS. THE VALUES IN PARENTHESES ARE THE ABSOLUTE WER REDUCTIONS OVER THE BASELINE.

	WER (%)(absolute reduction)					
	dev04		ldc-dev04		eval04	
factors	Baseline	SARV	Baseline	SARV	Baseline	SARV
standard	13.0	12.3 (-0.7)	18.4	17.7 (-0.7)	15.2	14.8 (-0.4)
standard + better CFG trees + modifiee constraints	-	12.1 (-0.9)	-	17.6 (-0.8)	-	14.7 (-0.5)

TABLE IX

COMPARISON OF WERS WITH DIFFERENT METHODS FOR TRAINING ON QUICKLY TRANSCRIBED BN SPEECH

Method	TDT data (h)	dev2003	eval2003	dev2004
Baseline	0	17.8	14.9	
LDC-raw	249	16.8	14.7	18.9
LDC-hand-corrected	184	15.9	13.9	18.1
CUED-recognized	245	15.9	14.0	18.2
Flexalign	248	15.8	14.4	18.0

Baseline: only Hub4 transcripts (no TDT-4 data); LDC-Raw: closed-captioned transcripts as is, with unalignable portions discarded; LDC-hand-corrected: transcripts corrected by human transcribers CUED-recognized: transcripts from biased recognizer developed at Cambridge University.

disfluencies; and every word is followed by an optional pause of variable length.

Our experiments were based on BN data from the TDT-4 collection released by LDC. The LDC baseline transcripts came from closed-captioned television shows. More recently, the LDC has released a manually corrected subset of these transcripts. These were used to gauge the improvement that can be obtained with automatic QT cleanup procedures, in spite of the fact that only about 73% of the TDT had been hand corrected. As a further point of reference, we also tested TDT-4 transcripts that were automatically generated by Cambridge University's BN recognition system using a biased LM trained on the closed captions. These automatic transcriptions were generated by a fast, stripped-down version of the regular Cambridge STT system that had the best performance in the NIST RT-03 BN STT evaluations [40].

To measure the quality of these four sets of transcripts, we trained acoustic models (of identical size) with them and evaluated the performance of resulting models on three different evaluation data sets, namely, the 2003 and 2004 TDT-4 development test sets defined by the DARPA EARS program participants (denoted dev2003 and dev2004) and the NIST RT-03 BN STT evaluation data (denoted eval2003). Besides the TDT-4 reference and acoustic data, the data used for acoustic model training includes 1996 and 1997 Hub-4 English BN speech (146 h). The recognition system used was the first stage of the full BN system (decode and LM rescore). Details of the experiment can be found in [41].

As Table IX shows, the flexible alignment model produced the lowest WER after first-pass decoding on both the Hub4 Broadcast News 2003 and 2004 TDT-4 development test set and the lowest WER on all test sets. On the eval2003 test set, the performance of the Flexalign model is still competitive with the performance of the two best models. Considering that the flexible alignment approach represents the fastest of the methods for generating suitable training transcripts (short of using the original QT transcripts), these results make our method quite attractive for large-scale use.

B. Porting the CTS System to Levantine Arabic

We found that most of the techniques developed for English could be ported to the development of a Levantine Conversational Arabic (LCA) system. We used, with few changes, the same architecture as the English system. However, because the training transcripts were in Arabic script and the amount of data was limited, some techniques did not have the expected effect. Here we describe the language-specific issues for this task and the effects these had on our modeling techniques.

1) Data: We used a corpus of LCA data provided by the LDC, consisting of 440 conversations (70 h of speech with about 500 K words). The training corpus vocabulary consists of 37.5 K words including 2.5 K word fragments and 8 nonspeech tokens. About 20 K words were singletons. The development (dev04) testset consists of 24 conversations (3 h of speech, about 16 K words). The OOV token rate for this set based on the training set vocabulary was 5.6%. The test set used for the RT-04 evaluations (eval04) consists of 12 conversations (1.5 h of speech, 8 K words).

The data was transcribed in Arabic script orthography, which is phonetically deficient in that it omits short vowels and other pronunciation information. The pronunciation lexicon was obtained by directly mapping the graphemes to phones and applying certain pronunciation rules such as assimilation of the "sun" letters and insertion of the appropriate short vowel in the presence of hamzas. All the rest of the short vowels were missing from the resulting pronunciations. We also experimented with techniques to automatically insert the missing vowels in the transcription and train vowelized acoustic models, as described below.

2) Acoustic Modeling: Due to the lack of short vowels in the grapheme-based lexicon, each acoustic model implicitly models a long vowel or a consonant with optional adjacent short vowels. NonCW and CW MFCC and PLP models (using HLDA, but without MLP features) were trained with decision-tree-based state clustering [42], resulting in 650 state clusters with 128 Gaussians for each cluster.

3) Discriminative Training: We found that the effect of the discriminative training procedure MPFE (described in Section IV-B) was smaller than in English. MPFE training for this task produced only a 2% relative improvement in the first iteration, while subsequent iterations increased WER. It is likely that grapheme models cannot substantially benefit from the discriminative training procedure since each grapheme represents a class of heterogeneous acoustic models rather than one single model. Also, the high WER and the numerous inconsistencies in the transcriptions can limit the effect of the MPFE procedure, especially since it relies on accurate phone alignments for discrimination.

4) Crossword Grapheme-Based Models: We found that the performance of CW models was worse than that of nonCW models, unless the word-boundary information was used during state clustering [43]. In English, word-boundary information improves the CW models, but in conversational Arabic it turns out to be critical. This could be attributed to the fact that the nature of the hidden short vowels is different at word boundaries compared to the within-word location.

5) Modeling of Short Vowels: Since previous work on Egyptian Colloquial Arabic (ECA) [44] has shown a significant benefit from using vowelized models versus grapheme-based ones, we attempted to do the same for the LCA system. Unlike in our previous work, no vowelized data or lexicon was available for this task.

In our first effort to use vowels in the LCA system we generated word pronunciation networks that included one optional generic vowel phone in all possible positions in the pronunciation. The possible positions were determined using the Buckwalter analyzer from LDC. For the words where it failed (24 K out of 37 K vocabulary), we included pronunciations that allowed an optional vowel between every consonant pair.

In our second approach we manually added the vowels on a small subset of the training data (about 40 K words), which was selected to have high vocabulary coverage. We trained a 4-gram character-based language model on this data, which was used as a hidden tag model to predict the missing vowels in all training data transcripts. That produced a vowelized text with about 7% character error rate.

When compared with the baseline grapheme models, the vowelized models did not show any significant improvement, possibly because of the inaccuracy in the vowelization procedure. Nevertheless, these systems contribute to significant improvements when combined with grapheme-based models as we will show in Section VI-B.7.

6) Language Modeling: We used two different types of language models in our system: standard word-based LMs and factored language models (FLMs) [45]. FLMs are based on a representation of words as feature vectors and a generalized parallel backoff scheme that utilizes the word features for more robust probability estimation. The word features can be the stem, root, affixes, or a tag that represents the morphological properties of the word. The structure of the model, that is, the set of features to be used and the combination of partial probability estimates from features, is optimized using a genetic algorithm [46]. By leveraging morphological structure, FLMs improve generalization over standard word-based LMs, which is especially important given the scarcity of in-domain training data, and the general lack of written LM training material in dialectal Arabic.

In our previous work on ECA, where the morphological features of each word were hand-annotated, factored language models yielded an improvement of as much as 2% absolute, for a baseline with approximately 40% word error rate [47]. For our present LCA system, word morphological information was not available and had to be inferred by other means. Since automatic morphological analyzers do not currently exist for dialectal Arabic, we used a simple script and knowledge of Levantine Arabic morphology to identify affixes and subsets of the parts of speech from the surface script forms. We also applied a morphological analyzer developed for Modern Standard Arabic [48] to obtain the roots of the script forms. Those forms that could not be analyzed retained the original script form as factors. It was found that this type of decomposition, although error-prone, yielded better results than using data-driven word classes. On the development set the perplexity was reduced from 223 to 212.

7) Evaluation System: The processing stages of the full system submitted for the RT-04 evaluation follow the setup of the English 20xRT CTS system describe in Section II, except that we used the PLP models for the lattice generation stages, and no SuperARV LM was used. Instead, an FLM was used for lattice rescoring (at the second lattice generation stage only). The N-best lists generated from these lattices used adapted PLP and MFCC graphemic models.

In Table X we show the contribution of vowelized models in this system. First we replaced the nonCW graphemic MFCC model with one that used the generic vowel approach, getting 0.6% to 0.4% WER improvements. Then we added the MFCC phonemic models that used the automatically vowelized data. We generated a third set of lattices using the vowelized LM, which were used to obtain the vowelized MFCC N-best lists with nonCW and CW models. These models improve the final performance by 0.8% to 1.0% absolute over the grapheme-

TABLE X
EFFECT OF THE VOWELIZED MODELS ON THE LEVANTINE ARABIC CTS SYSTEM WER

	dev04	eval04
grapheme	43.1	47.3
+ generic-vowel non-cw MFCC	42.5 (-0.6)	46.9 (-0.4)
+ auto-vowel MFCC models	42.1 (-1.0)	46.5 (-0.8)

TABLE XI
EFFECT OF THE FACTORED LM ON THE LEVANTINE ARABIC CTS SYSTEM WER

	dev04	eval04
No FLM	42.7	47.0
N-best rescoring FLM1	42.5	46.9
lattice rescoring FLM2	42.1 (-0.6)	46.7 (-0.3)

based system.

Table XI shows the contribution of the FLM to the final system's performance. To keep the runtime within 20xRT we used only the generic vowel MFCC model for these experiments. We see that including a bigram FLM only for final N-best rescoring improves the result by 0.2% and 0.1% on the two testsets. Using a trigram FLM for all lattice rescoring steps, we obtained improvements of 0.6% and 0.3% absolute, respectively.

C. Porting the CTS System to Mandarin

Porting to Mandarin required again very minor changes to the CTS system [49]. The core engine (the acoustic and language training paradigms and the decoding structure) remained the same. The main differences were in the addition of tone modeling with a tonal phone set, pitch extraction as part of the feature representation, word tokenization, and in the details of the web data collection.

1) Data: Three speech corpora were used for training acoustic models, all from LDC: Mandarin CallHome (CH), Mandarin CallFriend (CF), and the 58 h collected by Hong Kong University of Science and Technology (HKUST) in 2004. CH and CF together comprise 46 h (479 K words), including silence.

The transcriptions associated with these three corpora were all used to train word-based N-gram LMs. Because of the small size of the corpora, we also harvested web data as supplemental LM training data.

2) Language Modeling: As there are no spaces between written Chinese characters, there is no clear definition of words in Chinese and thus character error rate (CER) is usually measured when evaluating systems. Using single-character words in the vocabulary is not a good idea because of the short acoustic context and the high perplexity. Therefore, most Chinese STT systems define words as their vocabulary. We used the word tokenizer from New Mexico State University [50] to segment our text data into word units, resulting in 22.5 K unique words in CH+CF+HKUST training data. The HKUST corpus consisted of 251 phone calls. We randomly selected 25 phone calls as a heldout set to tune parameters in N-gram modeling. The rest of 226 phone calls (398 K words) were named subHKUST. CH+CF were used to train one trigram LM, subHKUST another. They were interpolated to create the baseline LM, LM_0, with interpolation weights to maximize the probability of generating the heldout set. No higher-order N-gram LMs were trained.

TABLE XII
WORD PERPLEXITY AND CER ON RT-04 CTS DATA (AUTOMATICALLY TOKENIZED)

	LM Weight					
	subHKUST	CH+CF	Web$_c$	Web$_t$	PPL	CER
LM$_0$	0.87	0.13	-	-	269.3	38.8%
LM$_1$	0.65	0.05	0.30	-	202.2	36.4%
LM$_2$	0.64	0.04	0.16	0.16	192.6	36.1%
LM$_3$	0.66	0.05	-	0.29	193.5	36.1%

In addition, we harvested two separate corpora from the World Wide Web to augment LM training. Key differences relative to our English web text collection method are in the text cleaning and normalization, and in the method for topic-dependent text collection [51]. The first batch was to fetch data in the style of general conversations by submitting the 8800 most frequent 4-grams from HKUST data to the Google search engine. The fetched data were then cleaned by removing pages with corrupted codes, removing HTML markers, converting Arabic digits into spoken words, and so on. Finally, pages with high perplexity computed by the baseline LM were filtered such that 60% (100 M) of the total number of words of the entire retrieved documents were retained. LM_1 was created by three-way interpolation of CH+CF, subHKUST, and conversational web data (Web_c), again maximizing the probability of the heldout set.

The second batch of web data collection focused on the 40 topics given in the HKUST collection. We defined 3-word key phrases for each particular topic, t, as word sequence $w_1w_2w_3$ if

$$\frac{C(w_1w_2w_3|t)/\alpha_t}{\Sigma_{j=1}^{40} C(w_1w_2w_3|j)/\alpha_j} > 0.3$$

and if there are enough training data in subHKUST in topic t. For rare topics, we manually designed key phrases based on the brief descriptions that were provided to the subjects as part of the data collection protocol. After the key phrases were defined for all 40 topics, we then queried Google for 40 collections of web pages. These 40 collections (a total of 244 M words), Web_t, were cleaned and filtered in the same way as Web_c and were combined to train a word-based N-gram, to be used in the 4-way interpolation.

The LM interpolation weights for these training corpora are indicated in Table XII. As one can see easily, subHKUST matched the heldout set strongly as they were from the same data collection effort. Additionally, the significant weights given to web data and the perplexity reduction show that our web query algorithm was effective, and that the web collection better matches the target task than other conversational speech that is not topically matched. Due to the lack of a Mandarin treebank and other resources, we did not build a Chinese

TABLE XIII
MANDARIN CERS ON THE MANUALLY SEGMENTED DEV04 CTS TEST SET

Acoustic Model	SI	SA
(1) PLP nonCW MLE, no pitch	41.5%	36.4%
(2) PLP nonCW MLE	40.4%	35.5%
(3) PLP CW MLE	39.5%	34.5%
(4) PLP CW SAT MLE	36.8%	34.0%
(5) PLP CW SAT MPFE	35.3%	32.9%
(6) MFCC CW SAT MPFE	36.2%	33.4%
(7) MFCC nonCW MPFE	40.0%	33.6%
Rover (5)+(6)+(7)	-	31.7%

SuperARV language model.

Recognition experiments were conducted on the 2-hour Dev04 set, collected again by HKUST. For fast turnaround, these experiments were conducted using simpler acoustic models and the 3xRT decoding architecture shown in Fig. 1. From Table XII, we chose the 4-way interpolated model, LM_2 as our final model for evaluation.

3) Acoustic Modeling: We obtained our initial pronunciation lexicon from BBN and reduced the phone set to 62, plus additional phones for silence, noise, and laughter. The phone set is similar to IBM's main vowel idea [52]. Tones are associated only with vowels and the /ng/ phone.

Similar to the English system, the front end features include both fixed frame rate MFCC and PLP static, delta, and double delta, with cepstra mean and variance removal, and VTLN. We did not find HLDA helpful in our early systems and therefore decided not to incorporate it into the final Mandarin CTS system. In addition, we used the Entropic *get_f0* program to compute pitch features, reduced pitch halving and doubling with lognormal tied mixture modeling [53], and then smoothed pitch values for unvoiced sections in a way similar to [52]. Along with pitch delta and double delta, our Mandarin feature vector was 42 dimensional. Neither voicing features nor any MLP related features were incorporated because of time constraints. Both nonCW and CW triphones with decision-tree clustered states were trained with MLE and MMIE+MPFE training. Questions for the decision tree clustering included those related to tone. Different state locations of the same phone and different tones of the same toneless phone were allowed to be clustered. CW models were further SAT transformed in the feature domain using 1-class constrained MLLR. The model size was 3500 clustered states with 32 Gaussians per cluster.

4) Results: To understand the incremental improvement with various technologies, we ran 1-pass decoding on top of the thick lattices in Fig. 1, which contained LM_2 trigram scores. We tested both speaker-independent (SI) models and MLLR unsupervised speaker-adapted (SA) performance. The results are shown in Table XIII.

The final evaluation made use of the full 20xRT decoding architecture, using MMIE+MPFE trained nonCW and CW triphone models with decision tree based state clustering, and trigram LM_2. The final CER was 29.7% on the RT-04 evaluation set.

VII. CONCLUSIONS

We surveyed a number of recent innovations in feature extraction, acoustic modeling, and language modeling, as used in the speech-to-text component of the SRI-ICSI-UW Rich Transcription system. Substantial improvements were obtained through trainable, discriminative phone-posterior features estimated by multilayer perceptrons, and modeling of syntactic word relationships in the SuperARV almost-parsing language model framework. Smaller gains were achieved by adding a novel combination of voicing features to the front end, by improving MPE-based discriminative training, and by predicting the speaker-dependent optimal number of regression classes in adaptation. Other techniques, while yet to be incorporated into our full system, show promise, such as the macro-averaging of frame-level statistics for more robust normalization, and the combination of multiple front ends at different frame rates.

We also developed an efficient flexible alignment technique for correcting and filtering imperfect, close-caption-style speech transcripts, for acoustic training purposes. Finally, we gained experience in porting STT technologies first developed for English to other languages, specifically, Arabic and Mandarin. We found that most techniques carry over, although language-specific issues have to be dealt with. In Arabic, the lack of detailed phonetic information encoded in the script form of the language has to be overcome, and the morphological complexity can be accounted for with factored language modeling. Mandarin Chinese, on the other hand, requires automatic word segmentation, and modifications to both feature extraction and dictionary to enable effective recognition of lexical tone.

VIII. ACKNOWLEDGMENT

The authors would like to thank their colleagues at IDIAP for their prior work on discriminant longer-term features, principally, H. Hermansky and H. Bourlard. The authors would also like to thank P. Woodland and Cambridge collaborators for making their training transcripts available for the authors' work, and to BBN for sharing its BN training transcripts and Mandarin dictionary.

REFERENCES

[1] Y. Liu, E. Shriberg, A. Stolcke, D. Hillard, M. Ostendorf, and M. Harper, "Enriching speech recognition with automatic detection of sentence boundaries and disfluencies", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, Sep. 2006, Special Issue on Progress in Rich Transcription.

[2] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech", *J. Acoust. Soc. Am.*, vol. 87, pp. 1738–1752, Apr. 1990.

[3] C. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of HMMs", *Computer Speech and Language*, vol. 9, pp. 171–186, 1995.

[4] D. Vergyri, A. Stolcke, V. R. R. Gadde, L. Ferrer, and E. Shriberg, "Prosodic knowledge sources for automatic speech recognition", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 208–211, Hong Kong, Apr. 2003.

[5] S. Wegmann, D. McAllaster, J. Orloff, and B. Peskin, "Speaker normalization on conversational telephone speech", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 339–341, Atlanta, May 1996.

[6] N. Kumar, *Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition*, PhD thesis, Johns Hopkins University, Baltimore, 1997.

[7] M. J. Gales, "Maximum likelihood multiple subspace projections for hidden Markov models", *IEEE Transactions on Speech and Audio Processing*, vol. 10, pp. 37–47, 2002.

[8] H. Jin, S. Matsoukas, R. Schwartz, and F. Kubala, "Fast robust inverse transform SAT and multi-stage adaptation", *in Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, pp. 105–109, Lansdowne, VA, Feb. 1998. Morgan Kaufmann.

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

[10] I. Bulyko, M. Ostendorf, and A. Stolcke, "Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures", in M. Hearst and M. Ostendorf, editors, *Proceedings of HLT-NAACL 2003, Conference of the North American Chapter of the Association of Computational Linguistics*, vol. 2, pp. 7–9, Edmonton, Alberta, Canada, Mar. 2003. Association for Computational Linguistics.

[11] M. Graciarena, H. Franco, J. Zheng, D. Vergyri, and A. Stolcke, "Voicing feature integration in SRI's Decipher LVCSR system", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 921–924, Montreal, May 2004.

[12] B. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification", *Journal of the Acoustical Society of America*, vol. 55, pp. 1304–1312, 1974.

[13] H. Hermansky and N. Morgan, "RASTA processing of speech", *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 578–589, Oct. 1994.

[14] H. Hermansky and S. Sharma, "Temporal patterns (TRAPs) in ASR of noisy speech", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 289–292, Phoenix, AZ, Mar. 1999.

[15] H. Bourlard and C. Wellekens, "Multilayer perceptrons and automatic speech recognition", *in Proc. of the First Intl. Conf. on Neural Networks*, vol. IV, pp. 407–416, San Diego, CA, 1987.

[16] H. Misra, H. Bourlard, and V. Tyagi, "New entropy based combination rules in HMM/ANN multi-stream ASR", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 741–744, Hong Kong, Apr. 2003.

[17] B. Y. Chen, *Learning Discriminant Narrow-Band Temporal Patterns for Automatic Recognition of Conversational Telephone Speech*, PhD thesis, University of California at Berkeley, 2005.

[18] Q. Zhu, A. Stolcke, B. Y. Chen, and N. Morgan, "Using MLP features in SRI's conversational speech recognition system", *in Proceedings of the 9th European Conference on Speech Communication and Technology*, pp. 2141–2144, Lisbon, Sep. 2005.

[19] S. Furui, "Cepstral analysis technique for automatic speaker verification", *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 29, pp. 254–272, 1981.

[20] J. L. Gauvain, G. Adda, L. Lamel, and M. Adda-Decker, "Transcribing broadcast news: The LIMSI Nov96 Hub4 system", *in Proceedings DARPA Speech Recognition Workshop*, pp. 56–63, Chantilly, VA, Feb. 1997. Morgan Kaufmann.

[21] V. R. R. Gadde, A. Stolcke, D. Vergyri, J. Zheng, K. Sonmez, and A. Venkataraman, "Building an ASR system for noisy environments: SRI's 2001 SPINE evaluation system", in J. H. L. Hansen and B. Pellom, editors, *Proceedings of the International Conference on Spoken Language Processing*, vol. 3, pp. 1577–1580, Denver, Sep. 2002.

[22] V. R. Gadde, K. Sonmez, and H. Franco, "Multirate ASR models for phone-class dependent n-best list rescoring", *in Proceedings IEEE Workshop on Speech Recognition and Understanding*, San Juan, Puerto Rico, Nov. 2005, To appear.

[23] L. R. Bahl, P. F. Brown, P. V. de Souza, and R. L. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition", *in Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 49–52, Tokyo, Apr. 1986.

[24] P. C. Woodland and D. Povey, "Large scale discriminative training of hidden Markov models of speech recognition", *Computer Speech and Language*, vol. 16, pp. 25–47, 2002.

[25] J. Huang, B. Kingsbury, L. Mangu, G. Saon, R. Sarikaya, and G. Zweig, "Improvements to the IBM Hub-5E system", *in NIST Rich Transcription Workshop*, Vienna, VA, May 2002.

[26] M. Mohri, "Finite-state transducers in language and speech processing", *Computational Linguistics*, vol. 23, pp. 269–311, 1997.

[27] A. Ljolje, F. Pereira, and M. Riley, "Efficient general lattice generation and rescoring", *in Proceedings of the 6th European Conference on Speech Communication and Technology*, vol. 3, pp. 1251–1254, Budapest, Sep. 1999.

[28] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: Word error minimization and other applications of confusion networks", *Computer Speech and Language*, vol. 14, pp. 373–400, Oct. 2000.

[29] G. Evermann and P. Woodland, "Posterior probability decoding, confidence estimation, and system combination", *in Proceedings NIST Speech Transcription Workshop*, College Park, MD, May 2000.

[30] A. Mandal, M. Ostendorf, and A. Stolcke, "Leveraging speaker-dependent variation of adaptation", *in Proceedings of the 9th European Conference on Speech Communication and Technology*, pp. 1793–1796, Lisbon, Sep. 2005.

[31] C. J. Leggetter and P. C. Woodland, "Flexible speaker adaptation using maximum likelihood linear regression", *in Proc. ARPA Spoken Language Technology Workshop*, pp. 104–109, 1995.

[32] M. J. Gales, "The generation and use of regression class trees for MLLR adaptation", Technical Report CUED/F-INFENG/TR263, Cambridge University, 1996.

[33] W. Wang and M. Harper, "The SuperARV language model: Investigating the effectiveness of tightly integrating multiple knowledge sources", *in Proceedings of Conference of Empirical Methods in Natural Language Processing*, pp. 238–247, 2002.

[34] H. Maruyama, "Structural disambiguation with constraints propagation", *in Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pp. 31–38, Pittsburgh, PA, June 1990. Association for Computational Linguistics.

[35] W. Wang, *Statistical Parsing and Language Modeling Based On Constraint Dependency Grammar*, PhD thesis, Purdue Univ., West Lafayette, IN, 2003.

[36] W. Wang, M. P. Harper, and A. Stolcke, "The robustness of an almost-parsing language model given errorful training data", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 240–243, Hong Kong, Apr. 2003.

[37] W. Wang, A. Stolcke, and M. P. Harper, "The use of a linguistically motivated language model in conversational speech recognition", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 261–264, Montreal, May 2004.

[38] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling", Technical Report TR-10-98, Computer Science Group, Harvard University, Aug. 1998.

[39] A. Venkataraman and W. Wang, "Techniques for effective vocabulary selection", *in Proceedings of the 8th European Conference on Speech Communication and Technology*, pp. 245–248, Geneva, Sep. 2003.

[40] M. J. F. Gales, D. Y. Kim, P. C. Woodland, H. Y. Chan, D. Mrva, R. Sinha, and S. E. Tranter, "Progress in the CU-HTK Broadcast News transcription system", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, Sep. 2006, Special Issue on Progress in Rich Transcription.

[41] A. Venkataraman, A. Stolcke, W. Wang, D. Vergyri, V. R. R. Gadde, and J. Zheng, "An efficient repair procedure for quick transcriptions", in S. H. Kim and D. H. Youn, editors, *Proceedings of the International Conference on Spoken Language Processing*, pp. 1961–1964, Jeju, Korea, Oct. 2004.

[42] S. Young, J. Odell, and P. Woodland, "Tree-based state tying for high accuracy acoustic modelling", *in Proceedings ARPA Workshop on Human Language*, pp. 307–312, 1994.

[43] P. Woodland, C. Leggetter, J. Odell, V. Valtchev, and S. Young, "The 1994 HTK large vocabulary speech recognition system", *in Proc. of ICASSP*, Detroit, 1995.

[44] K. Kirchhoff et al., "Novel approaches to Arabic speech recognition - final report from the JHU summer workshop 2002", Technical report, John-Hopkins University, 2002.

[45] J. Bilmes and K. Kirchhoff, "Factored language models and generalized parallel backoff", *in Proceedings of HLT/NACCL*, pp. 4–6, 2003.

[46] K. Duh and K. Kirchhoff, "Automatic learning of language model structure", *in Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pp. 148–154, 2004.

[47] D. Vergyri, K. Kirchhoff, K. Duh, and A. Stolcke, "Morphology-based language modeling for Arabic speech recognition", in S. H. Kim and D. H. Youn, editors, *Proceedings of the International Conference on Spoken Language Processing*, pp. 2245–2248, Jeju, Korea, Oct. 2004.

[48] K. Darwish, "Building a shallow Arabic morphological analyser in one day", *in Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pp. 47–54, Philadelphia, PA, 2002.

[49] M. Hwang, X. Lei, T. Ng, M. Ostendorf, A. Stolcke, W. Wang, J. Zheng, and V. Gadde, "Porting Decipher from English to Mandarin", Technical Report UWEETR-2006-0013, University of Washington Electrical Engineering Department, Seattle, WA, Presented at the NIST RT-04 Fall Workshop, 2004.

[50] W. Jin, "Chinese segmentation and its diambiguation", Technical Report MCCS-92-227, New Mexico State University, 1992.

[51] T. Ng, M. Ostendorf, M.-Y. Hwang, M. Siu, I. Bulyko, and X. Lei, "Web-data augmented language models for Mandarin conversational speech recognition", *in Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 589–593, Philadelphia, Mar. 2005.

[52] C. J. Chen, R. A. Gopinath, M. D. Monkowski, M. A. Picheny, and K. Shen, "New methods in continuous Mandarin speech recognition", in G. Kokkinakis, N. Fakotakis, and E. Dermatas, editors, *Proceedings of the 5th European Conference on Speech Communication and Technology*, vol. 3, pp. 1543–1546, Rhodes, Greece, Sep. 1997.

[53] M. K. Sönmez, L. Heck, M. Weintraub, and E. Shriberg, "A lognormal tied mixture model of pitch for prosody-based speaker recognition", in G. Kokkinakis, N. Fakotakis, and E. Dermatas, editors, *Proceedings of the 5th European Conference on Speech Communication and Technology*, pp. 1391–1394, Rhodes, Greece, Sep. 1997.

Andreas Stolcke (M'95–SM'05) received a Ph.D. in computer science from the University of California, Berkeley, in 1994.

He is currently a Senior Research Engineer at SRI International and ICSI. His research interests are in applying novel modeling and learning techniques to speech recognition, speaker identification, and natural language processing. He authored and coauthored over 120 research papers, as well as a widely used toolkit for statistical language modeling.

Barry Chen received a Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, in 2005.

He is currently a research engineer at the Lawrence Livermore National Laboratory. His interests include neural networks, graphical models, and general systems for pattern recognition.

Horacio Franco (M'92) received an Engineer Degree in electronics in 1978, and a Doctor in Engineering degree in 1996, both from the University of Buenos Aires, Buenos Aires, Argentina.

He is currently Chief Scientist at the Speech Technology and Research Laboratory, SRI International, Menlo Park, CA. Since joining SRI in 1990, he has made contributions in the areas of acoustic modeling, recognizer architectures, speech technology for language learning, noise robustness, and speech-to-speech translation systems. He has coauthored over 70 papers and several communications and book chapters.

Venkata Ramana Rao Gadde received the B.Tech. in electronics and electrical communications from IIT Kharagpur, Kharagpur, India, in 1982 and the M.Tech. and Ph.D. degrees in computer science from IIT Madras, Chennai, India, in 1986 and 1994, respectively.

He is a Senior Research Engineer at the Speech Technology and Research laboratory of SRI International, Menlo Park, CA. From 1988 to 1997, he was a member of the faculty at IIT Madras. His research interests include speech technology, image processing, statistical modeling and robust systems.

Martin Graciarena (M'06) received a M.S. degree from the State University of Campinas, Campinas, Brazil, and is currently pursuing the Ph.D. degree from the University of Buenos Aires, Buenos Aires, Argentina.

He is currently a Research Engineer at SRI International, Menlo Park, CA. His research interests are in noise robust speech recognition and speaker identification, feature development for large vocabulary speech recognition and deception detection. He has published more than 20 papers.

Mei-Yuh Hwang received the Ph.D. in computer science from Carnegie Mellon University, Pittsburgh, PA, in 1993.

She is currently a Senior Research Scientist at the Signal Speech, and Language Interpretation (SSLI) Laboratory, Department of Electrical Engineering, University of Washington, Seattle. Her research interests lie in speech and handwriting recognition, particularly in acoustic modeling and heuristic search. She has copublished more than 50 research papers and more than a dozen of patents.

Katrin Kirchhoff (M'99) received a Ph.D. degree in computer science from the University of Bielefeld, Bielefeld, Germany, in 1999.

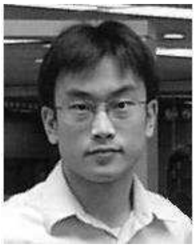
She is currently a Research Assistant Professor at the Signal Speech, and Language Interpretation (SSLI) Laboratory, Department of Electrical Engineering, University of Washington, Seattle. Her research interests are in speech and natural language processing, with an emphasis on multilingual applications. She has published over 50 research papers and is coeditor of a recent book on *Multilingual Speech Processing* (Academic, 1996).

Arindam Mandal received the B.E. degree from Birla Institute of Technology, Ranchi, India in 1997 and the M.S. degree from Boston University, Boston, MA, in 2000, both in electrical engineering. He is currently pursuing the Ph.D. degree at the Signal Speech, and Language Interpretation (SSLI) Laboratory, Department of Electrical Engineering, University of Washington, Seattle.

He was a Software Engineer at Nuance Communications until 2001. From October 2005 to June 2006, he was an International Fellow at SRI International. His research interests are in speech recognition and machine learning.

Nelson Morgan (S'76–M'80–SM'87–F'99) received the Ph.D. in electrical engineering from the Univesity of California, Berkeley, in 1980.

He is the Director of the International Computer Science Institute, Berkeley, CA, where he has worked on speech processing since 1988. He is Professor-in-Residence in the Electrical Engineering Department of the University of California, Berkeley, and has over 180 publications including three books.

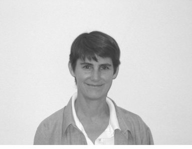


Xin Lei received the B.Eng. from Tsinghua University, Beijing, China, in 1999 and the M.S. degree from the University of Washington, Seattle, in 2003. He is currently pursuing the Ph.D. degree in electrical engineering at the University of Washington.

He was an intern at Microsoft Speech Component Group, Redmond, WA, in summer 2005. His research interests include Mandarin speech recognition, speaker adaptation and statistical pattern recognition.

Tim Ng received the M.Phil. degree in electrical engineering from Hong Kong University of Science and Technology, Hong Kong, China, in 2005.

He is currently in the Speech Group, BBN Technologies, Cambridge, MA. His research interests are in acoustic and language modeling for large-vocabulary speech recognition.

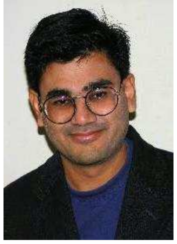


Mari Ostendorf (M'85–SM'97–F'05) received the Ph.D. in electrical engineering from Stanford University, Stanford, CA, in 1985.

She has worked at BBN Laboratories (1985-1986) and Boston University (1987-1999), and since 1999, she has been a Professor of Electrical Engineering at the University of Washington, Seattle. Her research interests are in dynamic and linguistically-motivated statistical models for speech and language processing, resulting in over 160 publications.

Kemal Sönmez (M'04) received the Ph.D. in electrical engineering from the University of Maryland, College Park, in 1998.

During his graduate studies, he was a student member of the TI Speech Group for three years. He is currently a Senior Research Engineer at SRI International, Menlo Park, CA. His research interests include computational prosody, stylistic modeling of speech, speaker recognition, acoustical modeling for STT, and biological sequence analysis for computational biology. He was a visiting research scholar at the Division of Applied Mathematics at Brown University, Providence, RI, in 2004, and a senior member at the 2004 JHU-CLSP summer workshop.



Anand Venkataraman received the Ph.D. degree in computer science from Massey University, Palmerston North, New Zealand, in 1997.

He is currently a Research Engineer at SRI International, Menlo Park, CA. His research interests are in the novel use of speech and language technologies to voice-enabled applications and especially in information retrieval using speech. He has authored numerous research papers in speech technology and public domain toolkits for hidden Markov modeling, artificial neural networks, and probabilistic finite state automata.

Dimitra Vergyri (M'00) received the M.S. and Ph.D. degrees in electrical and computer engineering from Johns Hopkins University, Baltimore, MD, in 1996 and 2000, respectively.

She is currently a Research Engineer at SRI International, Menlo Park, CA. Her research interests encompass all aspects of statistical modeling for speech recognition, audio-visual speech processing, and information theory. Her recent work has focussed on recognition of Arabic and low-resource languages.

Wen Wang (S'98–M'04) received the Ph.D. in computer engineering from Purdue University, West Lafayette, IN, in 2003.

She is currently a Research Engineer at SRI International, Menlo Park, CA. Her research interests are in statistical language modeling, speech recognition, natural language processing techniques and applications, and optimization. She authored and coauthored about 30 research papers and served as reviewer for over ten journals and conferences.

Dr. Wang is a member of the Association for Computational Linguistics.

Jing Zheng (M'06) received the Ph.D. in electrical engineering from the Tsinghua University, Beijing, China, in 1999.

He is currently a Research Engineer at SRI International, Menlo Park, CA. His research interests include automatic speech recognition, machine translation, and handwriting recognition. He authored and coauthored over 35 publications in these fields.

Qifeng Zhu (M'01) received a B.E. degree from Tsinghua University, Beijing, China, in 1994, the M.S. degree from the Chinese Academy of Sciences, Beijing, in 1997, and the Ph.D. degree from the University of California, Los Angeles, in 2001.

He was a Senior Researcher at ICSI, Berkeley, until 2005. Before that he was a Research Engineer at Nuance Communications, in Menlo Park, CA. Currently he is a Member of Technical Staff in the Speech Laboratory, Texas Instruments, Dallas, TX.