

The LIMSI Broadcast News Transcription System

Jean-Luc Gauvain, Lori Lamel, Gilles Adda

Spoken Language Processing Group
LIMSI-CNRS, BP 133
91403 Orsay cedex, FRANCE
{gauvain, lamel, gadda}@limsi.fr

June 3, 2002

Keywords: speech recognition, broadcast news transcription, audio partitioning, acoustic modeling, language modeling, lexical modeling

Abstract

This paper reports on activities at LIMSI over the last few years directed at the transcription of broadcast news data. We describe our development work in moving from laboratory read speech data to real-world or ‘found’ speech data in preparation for the ARPA Nov96, Nov97 and Nov98 evaluations. Two main problems needed to be addressed to deal with the continuous flow of inhomogeneous data. These concern the varied acoustic nature of the signal (signal quality, environmental and transmission noise, music) and different linguistic styles (prepared and spontaneous speech on a wide range of topics, spoken by a large variety of speakers).

The problem of partitioning the continuous stream of data is addressed using an iterative segmentation and clustering algorithm with Gaussian mixtures. The speech recognizer makes use of continuous density HMMs with Gaussian mixture for acoustic modeling and 4-gram statistics estimated on large text corpora. Word recognition is performed in multiple passes, where initial hypotheses are used for cluster-based acoustic model adaptation to improve word graph generation. The overall word transcription error of the LIMSI evaluation systems were 27.1% (Nov96, partitioned test data), 18.3% (Nov97, unpartitioned data), 13.6% (Nov98, unpartitioned data) and 17.1% (Fall99, unpartitioned data with computation time under 10x real-time).

Cet article présente les travaux effectués au LIMSI pour le développement d’un système de traitement automatique d’informations radio et télédiffusées. Partant d’un système de transcription de textes lus, nous décrivons les adaptations qui ont été nécessaires pour le traitement d’un flux audio continu et de données dites “trouvées”. Ces développements ont été validés dans le cadre des évaluations ARPA BN (Nov96, Nov97, Nov98 et Dec99). Les principales difficultés posées par ce type de données sont liées à leur nature hétérogène, qu’il s’agisse de changements de nature acoustique (environnement, communication, musique) ou de nature linguistique (styles d’élocution, diversités des sujets et des locuteurs),.

La partition du flux continu est effectuée de manière itérative, par un algorithme de segmentation-agglomération reposant sur des mélanges de Gaussiennes. Le système de reconnaissance utilise des modèles de Markov cachés à densités continues pour la modélisation acoustique, et des statistiques 4-grammes de mots estimées sur un grand corpus de textes et de parole transcrite pour modèle de langage. La transcription en mots est obtenue en plusieurs passes de décodage, où les hypothèses intermédiaires sont utilisées pour adapter les modèles acoustiques. Les taux d’erreur obtenues avec différentes versions de ce système lors des évaluations ARPA sont 27,1% (Nov96 avec partition manuelle), 18,3% (Nov97), 13,6% (Nov98) et 17,1% (Dec99, moins de 10 fois le temps réel).

Dieser Artikel berichtet über die Tätigkeiten am LIMSI während der letzten Jahren mit dem Ziel der Spracherkennung von Nachrichtensendungen. Wir beschreiben unsere Forschungsarbeiten der Portierung von unter Laborbedingungen gelesener Sprache zu natürlicher freier Sprache während der Vorbereitung der ARPA Nov96, Nov97 und Nov98 Evaluierungen. Zur Bearbeitung des kontinuierlichen Stroms von inhomogenen Daten sind zwei grundlegende Probleme zu lösen.

Diese betreffen einerseits die unregelmässige akustische Natur des Signals (Signalqualität, Hintergrund- und Übertragungsrauschen, Musik, ...) und andererseits die unterschiedlichen linguistischen Stile (vorbereitete oder spontane Sprache, eine große Themenvielfalt und viele unterschiedliche Sprecher).

Der kontinuierliche Audiostrom wird mit Hilfe eines iterativen Segmentierungs- und Klusterungsalgorithmus auf der Basis von Gauss Mischverteilungen partitioniert. Das Spracherkennungssystem verwendet HMMs mit kontinuierlichen Gauss Mischverteilungen zur akustischen Modellierung und 4-gram Statistiken, welche mit Hilfe grosser Textsammlungen geschätzt wurden. Die Worterkennung erfolgt in mehreren Phasen, wobei die Wortgraphen nach und nach mit Hilfe akustischer Modellanpassung verbessert werden. Die Wordfehlerrate von LIMSI's Spracherkennungssystemen beträgt 27,1% (Nov96, segmentierte Testdaten), 18,3% (Nov98, unsegmentierte Testdaten) und 17,1% (Herbst 99, unsegmentierte Testdaten und Ausführungszeiten von weniger als 10-facher Echtzeit).

Stichwörter: Spracherkennung, Nachrichtenübersetzung, Audio-Segmentierung, akustische Modellierung, Spachmodellierung

1 Introduction

Over the last 5 years significant advances have been made in large vocabulary, continuous speech recognition, which has been a focal area of research, serving as a test bed to evaluate models and algorithms [5, 6, 45]. However, these tasks remain relatively artificial as they mainly make use of laboratory read speech data. In this paper we report on moving toward real-world speech data in order to build a system for transcribing radio and television broadcast news [6, 7, 8, 9]. While this paper focuses on our work in developing a broadcast news transcription system for American English, in the context of the LE-4 OLIVE project we have also developed systems for the French and German languages.

Radio and television broadcast shows are challenging to transcribe as they contain signal segments of various acoustic and linguistic nature. The signal may be of studio quality or have been transmitted over a telephone or other noisy channel (ie., corrupted by additive noise and nonlinear distortions), or can contain speech over music or pure music segments. Gradual transitions between segments occur when there is background music or noise with changing volume, and abrupt changes are common when there is switching between speakers in different locations. The speech is produced by a wide variety of speakers: news anchors and talk show hosts, reporters in remote locations, interviews with politicians and common people, unknown speakers, new dialects, non-native speakers, etc. Speech from the same speaker may occur in different parts of the broadcast, and with different channel conditions. The linguistic style ranges from prepared speech to spontaneous speech. Acoustic models trained on clean, read speech, such as the Wall Street Journal (WSJ) corpus [35], are clearly inadequate to process such inhomogeneous data.

Our research has been aimed at addressing two principle types of problems encountered in transcribing broadcast news data: those related to the varied acoustic properties of the signal, and those related to the linguistic properties of the speech. The first problem is resolved by partitioning the data into homogenous segments, where each segment can then be classified as to the segment type. Specific acoustic models can then be trained for the different acoustic conditions. The work on data partitioning is described in Section 3. Issues in acoustic modeling are discussed in Section 4.

In order to address variability observed in the linguistic properties, we analyzed differences in

read and spontaneous speech, with regard to lexical items, word and word sequence pronunciations, and the frequencies and distribution of hesitations, filler words, and respiration noises. As a result of this analysis, these phenomena were explicitly modeled in both the acoustic and language models as described in [14]. The phone set was enlarged to explicitly model filler words and breath noise, resulting in specific context-dependent acoustic models. Compound words were introduced as a means of modeling reduced pronunciations for common word sequences. These aspects are discussed in Section 5. In Section 7 the word decoder is described, with considerations for processing time. We conclude with a discussion of issues in broadcast news transcription and highlight some of the lessons we have learned in working on this problem.

2 Background

The broadcast news task has been used to assess and improve speech recognition technology since Nov95, when a DARPA dry run evaluation was held using 10 hours of MarketPlace data. Prior to the next three evaluations, substantially more transcribed broadcast news acoustic training data and textual data for language modeling have been made available via the Linguistic Data Consortium (for more detail see the LDC contribution to this issue). In Nov96, about 50 hours of transcribed data were available. These data came from 10 different sources: ABC (Nightline, World News Now, World News Tonight), CNN (Early Prime, Headline News, Prime News, The World Today), CSPAN Washington Journal, and NPR (All Things Considered, Marketplace). For the Nov97 evaluation, an additional 50 hours of transcribed data from the same sources were made available. In 1998 the amount of transcribed acoustic training data was doubled, resulting in a total of 200 hours of data from (in addition to the above sources): CNN (Early Edition, Prime Time Live), and CSPAN Public Policy. (For more details see the LDC paper in this issue.)

As mentioned above, broadcast data is comprised of acoustic segments of varied acoustic and linguistic natures. The acoustic differences primarily concern the different recording channels (wide-band/telephone) and recording environment (studio/on-site location, background music or noise). Given the variety of acoustic and linguistic data types, a set of *focus conditions* [41] were identified by NIST so as to evaluate system performance under certain specified conditions.¹

The test data for each year were chosen from multiple sources, including some not present in the training material. The Nov96 test contained 106 minutes of data taken from four shows. The Nov97 and Nov98 test consisted for about 3 hours of audio data, where portions were extracted from broadcasts so as to focus on the F0 and F1 data types. (For more information see the NIST paper in this issue.)

For the Nov96 evaluation, we trained different acoustic model sets so as to address the different focus conditions [14]. Wideband acoustic models were trained on about 100 hours (46k sentences) from 355 speakers in the WSJ0/1 corpus and 50 hours of broadcast news data distributed by NIST. The WSJCAM0 corpus was also used to train models for British English speakers, since some non-native speakers of American English may more closely resemble British speakers. For telephone speech models, reduced bandwidth models were first trained on a bandlimited version the WSJ corpus. The resulting models were then adapted using MAP estimation with 7k sentences of WSJ telephone speech data taken primarily from the Macrophone corpus, and then adapted with the telephone portion of the broadcast data. Type-specific acoustic models were trained for the different categories of data defined for the Nov96 partitioned evaluation: high quality prepared speech, high quality spontaneous speech,

¹In the Nov96 evaluation there were two components, the “partitioned evaluation” (PE) and the “unpartitioned evaluation” (UE). The PE condition was used to compare systems. In later evaluations the focus conditions have been used to assess performance on the different data types. For more details, see the papers from NIST and LDC in this issue.

telephone speech, speech over music, speech in noise, non-native speakers, and miscellaneous. For the Nov96 partitioned evaluation the focus condition of each test segment was provided by NIST. In the LIMSIS Nov96 system the telephone decision was based on the output of the Gaussian segment classifier, and all other attributes were taken from the provided segment annotation. Five acoustic model sets were trained for broadcast quality speech (conditions F0 and F1), telephone quality speech (condition F2), speech in the presence of music (condition F3), speech in the presence background noise (condition F4), and non-native speech (condition F5). In total there were 20 model sets: 5 conditions \times 2 genders \times 2 decoding passes. Dealing with so many different model sets was relatively difficult to manage, both for training and decoding. The performance differences were also quite small: On the 1996 development data (2 hours taken from 6 shows), the word error rate resulting from a second decoding pass with a trigram language model was 26.8% using the type-specific model sets as compared to 27.1% with the two model sets [15]. When more transcribed data was made available, more accurate acoustic models could be trained and it no longer seemed as “interesting” to use focus condition-specific models. Additionally, in the transcriptions of the second set of 100 hours BN acoustic data the background conditions were not annotated, so supervised training was not an option for this part of the data. However, it is probably worth looking into using a set of background acoustic conditions (speech in music, noisy speech) if accurate labels can be automatically obtained. When transcriptions of additional acoustic training data were released by LDC, we once again investigated various approaches to build acoustic models from the available read-speech and Hub4 training data. Acoustic model development aimed to minimize the word error rate on the eval96 test data. Since these experiments showed no clear gain from using the WSJ data to initialize the acoustic models, most of the development work was carried out using only the Hub4 data [17].

In addition to the acoustic training data, in subsequent years more textual data sources were distributed via the LDC. In our Nov96 system the language model was trained on 161 million words of newspaper texts (the 1995 Hub3 and Hub4 LM material), 132 million words of broadcast news transcriptions (years 92 to 96), as well as 430 K words from the transcriptions of the 1995 and 1996 acoustic training data. In 1997, the same training texts sources were available, with a total of 866 K words in acoustic data transcriptions. In 1998, substantially more LM training texts were used: a total of 203 M words of broadcast news transcripts (from LDC and PSMedia), 343 M words of NAB newspaper texts and AP Wordstream texts, and 1.6 M words of transcriptions of the acoustic training data.

While the LIMSIS Nov’98 systems serves as basis for the remainder of this paper, reference is made to earlier systems and recent progress when appropriate.

3 Data Partitioning

3.1 Need for partitioning

While it is evidently possible to transcribe the continuous stream of audio data without any prior segmentation, partitioning offers several advantages over this straight-forward solution. First, in addition to the transcription of what was said, other interesting information can be extracted such as the division into speaker turns and the speaker identities. Prior segmentation can avoid problems caused by linguistic discontinuity at speaker changes. By using acoustic models trained on particular acoustic conditions, overall performance can be significantly improved, particularly when cluster-based adaptation is performed. Finally eliminating non-speech segments and dividing the data into shorter segments (which can still be several minutes long), reduces the computation time and simplifies decoding.

Various approaches have been proposed to partition the continuous stream of audio data. Most

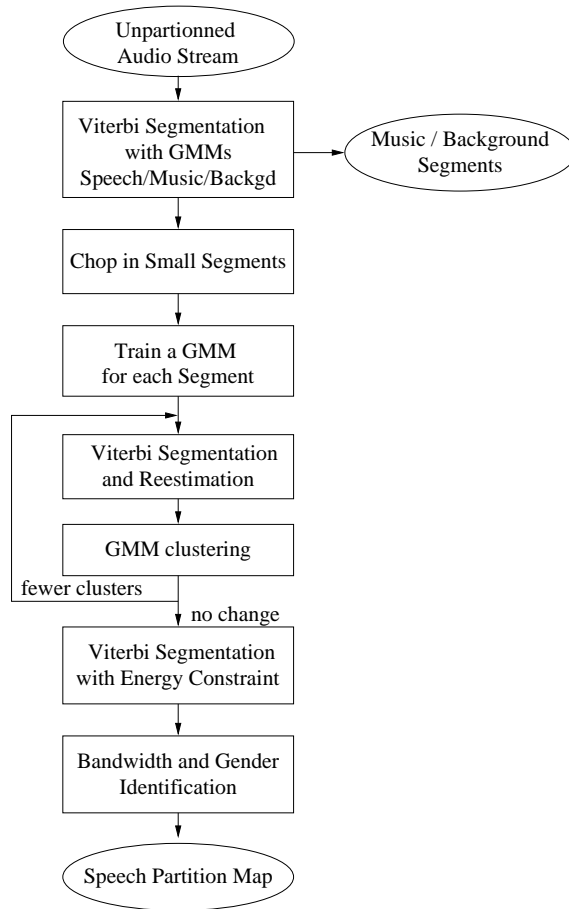


Figure 1: Partitioning algorithm.

of these approaches rely on a two step procedure, where the audio stream is first segmented in an attempt to locate acoustic changes (associated with changes in speaker, background or environmental condition, and channel condition) and then the resulting segments are clustered (usually using Gaussian models). Each cluster is assumed to identify a speaker or more precisely, a speaker in a given acoustic condition. The segmentation procedures can be classified into three approaches: those based on phone decoding [25, 31, 42], distance-based segmentations [29, 40], and methods based on hypothesis testing [12, 43]. Our partitioning approach, which is not based on such a two step procedure, relies on an audio stream mixture model. Each component audio source, representing a speaker in a particular background and channel condition, is in turn modeled by a mixture of Gaussians. The segment boundaries and labels are jointly identified using the iterative procedure described below.

3.2 Audio Stream Mixture Model

The segmentation and labeling procedure introduced in [17, 18] is shown in Figure 1. First, the non-speech segments are detected (and rejected) using Gaussian mixture models (GMMs). These GMMs, each with 64 Gaussians, serve to detect speech, pure-music and other (background). The acoustic feature vector used for segmentation contains 38 parameters. It is the same as the recognition feature vector except that it does not include the energy, although the delta energy parameters are included. The GMMs were each trained on about 1h of acoustic data, extracted from the training data after segmentation with the transcriptions. The speech model was trained on data of all types,

with the exception of pure music segments and silence portions of segments transcribed as speech over music. In order to detect speech in noisy conditions a second speech GMM was trained on the F4 segments in the 1996 data set. These models are expected to match all speech segments. The music model was trained only on portions of the data that were labeled as pure music, so as to avoid mistakenly detecting speech over music segments. The silence model was trained on the segments labeled as silence during forced Viterbi alignment, after excluding silences in segments labeled as containing speech in the presence of background music. All test segments labeled as music or silence are removed prior to further processing.

A maximum likelihood segmentation/clustering iterative procedure is then applied to the speech segments using GMMs and an agglomerative clustering algorithm. Given the sequence of cepstral vectors corresponding to a show (x_1, \dots, x_T) , the goal is to find the number of sources of homogeneous data and the places of source changes. The result of the procedure is a sequence of non-overlapping segments (s_1, \dots, s_N) with their associated segment cluster labels (c_1, \dots, c_N) , where $c_i \in [1, K]$ and $K \leq N$ is the number of segment clusters. Each segment cluster is assumed to represent one speaker in a particular acoustic environment. In absence of any prior knowledge about the stochastic process governing (K, N) and the segment lengths, we use as objective function a penalized log-likelihood of the form

$$\sum_{i=1}^N \log f(s_i | \lambda_{c_i}) - \alpha N - \beta K$$

where $f(\cdot | \lambda_k)$ is the p.d.f. (with a fixed number of parameters) corresponding to the cluster k , and where $\alpha > 0$ and $\beta > 0$. The terms αN and βK , which can be seen as segment and cluster penalties, correspond to the parameters of exponential prior distributions for N and K . It is easy to prove that starting with overestimates of N and K , alternate Viterbi reestimation and agglomerative clustering gives a sequence of estimates of (K, N, λ_k) with non decreasing values of the objective function. In the Viterbi step we reestimate (N, λ_k) so as to increase $\sum_i \log f(s_i | \lambda_{c_i}) - \alpha N$ (i.e. adding a segment penalty α in the Viterbi search) whereas in the clustering step two or more clusters can be merged as long as the resulting log-likelihood loss per merge is less than β .² Since merging two models can reduce the number of segments, the change in segment penalty is taken into account during clustering. This algorithm stops when no merge is possible. A constraint on the cluster size is used to ensure that each cluster corresponds to at least 10s of speech. (Recall that the previously rejected non-speech segments are not considered here.)

For single Gaussian models the merging criterion is easy to implement since the log-likelihood loss can be directly computed from the sufficient statistics of the corresponding segments [24, 28]. In the more general case of Gaussian mixtures, there are no sufficient statistics and there is no direct solution to compute the resulting mixture and/or the log-likelihood loss. We can envision estimating the new mixture from the data but this is a costly procedure. Another solution that we adopted for this work is to modify the objective function, replacing the likelihood function by the complete data likelihood of the Gaussian mixtures and extending the maximum likelihood clustering method to the Gaussian level. To estimate the log-likelihood loss for two Gaussian mixtures, we simply have to compute the sum of the log-likelihood loss while clustering the Gaussians of the 2 mixtures (until we get the desired number of Gaussians). We have used 8 mixture components per cluster, so to compute the log-likelihood loss induced by merging two clusters agglomerative clustering is performed starting with 16 Gaussians until 8 Gaussians are left.

The process is initialized using a simple segmentation algorithm based on the detection of spectral change (similar to the first step used in the CMU'96 system [40]). The threshold is set so as to over-

²This clustering criterion is closely related to the MDL or BIC criterion.

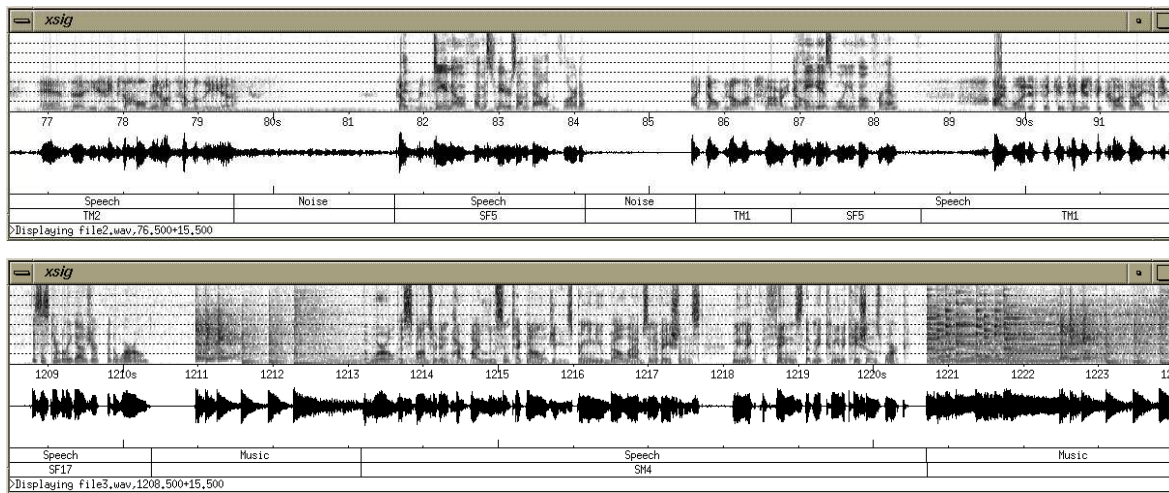


Figure 2: Spectrograms illustrating results of data partitioning on sequences extracted from broadcasts. The transcript gives automatically generated segment type: Speech, Music, or Noise. For the speech segments the cluster labels specify the identified bandwidth (T=telephone-band/S=wideband) and gender (M=male/F=female), as well as the number of the cluster.

generate segments, roughly 5 times as many segments as true speaker turns. Initially, the cluster set consists of a cluster per segment. This is followed by Viterbi training of the set of GMMs (one 8-component GMM per cluster). This procedure is controlled by 3 parameters: the minimum cluster size (10s), the maximum log-likelihood loss for a merge (α), and the segment boundary penalty (β). When no more merges are possible, the segment boundaries are refined using the last set of GMMs and an additional relative energy-based boundary penalty, within a 1s interval. This is done to locate the segment boundaries at silence portions, attempting to avoid cutting words (but sometimes this still occurs).

Speaker-independent GMMs corresponding to wideband speech and telephone speech (each with 64 Gaussians) are then used to label telephone segments. This is followed by segment-based gender identification, using 2 sets of GMMs with 64 Gaussians (one for each bandwidth). The result of the partitioning process is a set of speech segments with cluster, gender and telephone/wideband labels as illustrated in Figure 2.

3.3 Partitioning Results

In developing the partitioner we used the dev96 data set, and we evaluated the frame level segmentation error (similar to [25]) on the 4 half-hour shows in the eval96 test data using the manual segmentation found in the reference transcriptions. The NIST transcriptions of the test data contain segments that were not scored, since they contain overlapping or foreign speech, and occasionally there are small gaps between consecutive transcribed segments. Since we considered that the partitioner should also work correctly on these portions, we relabeled all excluded segments as speech, music or other background.

Table 1(top) shows the segmentation frame error rate and speech/non-speech errors for the 4 shows. The average frame error is 3.7%, but is much higher for show 1 than for the others. This is due to a long and very noisy segment that was deleted. Averaged across shows the gender labeling has a 1% frame error. In addition to these errors, there are 6.2% female speech frames deleted (largely due to the same segment) and 1.7% of the male frames deleted. The bottom of Table 1 shows measures of

<i>Show</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>Avg</i>
<i>Frame Error</i>	7.9	2.3	3.3	2.3	3.7
<i>M/F Error</i>	0.4	0.6	0.6	2.2	1.0
<i>#spkrs/#clusters</i>	7/10	13/17	15/21	20/21	-
<i>ClusterPurity</i>	99.5	93.2	96.9	94.9	95.9
<i>Coverage</i>	87.6	71.0	78.0	81.1	78.7

Table 1: Top: Speech/non-speech frame segmentation error (%), using NIST labels, where missing and excluded segments were manually labeled as speech or non-speech. Bottom: Cluster purity and best cluster coverage (%).

<i>System Step</i>	<i>Test set (Word Error)</i>		
	<i>Eval96</i>	<i>Eval97</i>	<i>Eval98</i>
<i>Step1 3gram manual</i>	24.7	18.2	18.0
<i>automatic</i>	25.3	18.4	18.3
<i>Step2 3gram manual</i>	20.2	14.2	13.5
<i>automatic</i>	21.0	14.6	14.2

Table 2: Word error with manual/automatic segmentations using the Nov98 system for 3 data sets.

the cluster homogeneity. The first entry gives the total number of speakers and identified clusters per file. In general there are more clusters than speakers, as a cluster can represent a speaker in a given acoustic environment. The second measure is the cluster purity, defined as the percentage of frames in the given cluster associated with the most represented speaker in the cluster. (A similar measure was proposed in [12], but at the segment level.) The table shows the weighted average cluster purities for the 4 shows. On average 96% of the data in a cluster comes from a single speaker. When clusters are impure, they tend to include speakers with similar acoustic conditions. The “best cluster” coverage is a measure of the dispersion of a given speaker’s data across clusters. We averaged the percentage of data for each speaker in the cluster which has most of his/her data. On average 80% of the speaker data is going to the same cluster. In fact, the average value is a bit misleading as there is a large variance in the best cluster coverage across speakers. For most speakers the cluster coverage is close to 100%, i.e., a single cluster covers essentially all frames of their data. However, for a few speakers (for whom there is a lot of data), the speaker is covered by two or more clusters, each containing comparable amounts of data.

We also investigate the effect of automatic vs manual partitioning on the recognizer performances. Table 2 compares the word error rates with automatic and manual (NIST) partitions on three evaluation data sets. The performance loss is about 1.5% relative after the first decoding step (ie. no adaptation). It is higher (2.4%) on the eval96 data due to a long deleted segment in show 1. After adaptation (step 2) the relative performance loss is about 4%, indicating that the clustering process is inappropriately merging or splitting some of the speakers’ data. It appears that clustering errors are more detrimental to performance than segmentation ones.

4 Acoustic Modeling

The acoustic models were trained on all the available transcribed task-specific training data, amounting to about 150 hours of audio data. We used the August 1997 and February 1998 releases of the LDC transcriptions. Overlapping speech portions were detected in the transcriptions and removed

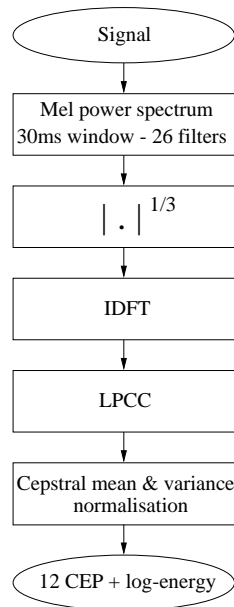


Figure 3: PLP-like frontend

from the training data. The phone set contains 48 units, including specific phone symbols used to explicitly model silence, filler words and breath noises.³ The decision to model these with specific phones was based on a desire to capture any possible acoustic differences from similar phones in the phone set and at the same time to avoid possible contamination of these other phone models.

The following PLP-like [26] acoustic parameterization has been used in the LIMSI systems since 1996. The speech features consist of 39 cepstral parameters derived from a Mel frequency spectrum estimated on the 0-8kHz band (or 0-3.5kHz for telephone data) every 10ms. For each 30ms frame the Mel scale power spectrum is computed, and the cubic root taken followed by an inverse Fourier transform. LPC-based cepstrum coefficients are then computed. These cepstral coefficients are normalized on a segment cluster basis using cepstral mean removal and variance normalization (cf. figure 3). Each resulting cepstral coefficient for each cluster has a zero mean and unity variance. The 39-component acoustic feature vector consists of 12 cepstrum coefficients and the log energy, along with the first and second order derivatives. This feature vector has fewer parameters than the 48-component feature vector used previously [22], but has better performance on the Hub4 data (3% relative gain).

The acoustic models are sets of tied-state word-position dependent triphones. Each phone model is a tied-state left-to-right, 3-state CDHMM with Gaussian mixture observation densities (typically 32 components). The triphones are word-position dependent in the sense that different models are used for word internal phones and word boundary phones. The word boundary phones are subsequently distinguished as word-initial, word-final, or both word-initial and final (monophone words). The triphone contexts to be modeled are selected based on their frequencies in the training data. We do not try to predict unseen triphones, but rather backoff by merging contexts for infrequent triphone contexts. First we try to merge phones with a common right context, then a common left context, and finally the remaining data are merged into a context-independent model. With the Hub4 training data over 28000 triphone contexts are modeled, resulting in a triphone coverage of over 99%.

³The silence (or background noise) word model is special, as it can be inserted between any two words and does not appear in the language model. In contrast, the filler word and breath noise models are explicitly represented in the language model.

In our Nov96 system, position-dependent acoustic models were used in the first decoding pass in order to reduce the search space and the decoding time, even though slightly better performance was obtained with position-independent models [14]. However, in 1997 with twice as much acoustic training data available we were able to model a larger number of contexts, and a slight gain was observed with position-dependent models on the Hub4 data [17].

HMM training requires an alignment between the audio signal and the phone models, which usually relies on a perfect orthographic transcription of the speech data and a good phonetic lexicon. Each speech segment is first Viterbi aligned to the orthographic transcription so as to produce a time-aligned phone transcription. Since the reference transcriptions and the phonetic lexicon are not really perfect, this alignment procedure may not succeed. In this case the error can be manually corrected, or the segment can simply be discarded. (In practice, errors are corrected when the training data is limited, and segments are discarded when a lot of training data are available. As more data was made available, we spent less time correcting errors.) Discarded segments are those for which there is no complete Viterbi alignment due to beam-pruning or when some duration criteria are not respected such as a maximum allowable phone duration. For example, a phone duration longer than 500ms is likely to be indicative of an error, for phones other than silence or breath noise.

After alignment, HMM parameter estimation is done using the EM estimation procedure starting with a single Gaussian per tied-state and splitting each Gaussian until the maximum number of Gaussians per state (usually 32) is reached. To avoid problems due to data sparseness (which is unlikely with state-tying) a Bayesian estimation procedure is used with a common prior for all Gaussians of a given state and a minimum frame count (accumulated Gaussian probabilities for all frames) is also required to keep a Gaussian. This alignment/reestimation procedure is iterated several times to refine the acoustic models, usually increasing the number of parameters progressively.

Separate male and female models obtained with MAP estimation of SI seed models [23] are used to more accurately model the speech data. Both wideband and telephone band models were estimated, where the telephone band models are trained using a low pass filtered version of the data set. Each model set contains about 11500 tied-states and a total of 330k Gaussians.

We have compared divisive decision tree clustering with agglomerative clustering for state-tying. Both approaches can obtain comparable model sets, but we have found that divisive decision tree clustering is particularly interesting when there are a very large number of states to cluster since it is at the same time both faster and is more robust than a bottom-up greedy algorithm, and therefore much easier to tune. The set of 184 questions used in our Nov'98 system concern the phone position, the distinctive features (and identities) of the phone and the neighboring phones. The questions are given in Table 3, and the most frequently used questions for the largest model set are given in Table 4. One tree is constructed for each state of each phone. The tree is built so as to maximize the likelihood of the training data using single Gaussian state models, penalized by the number of tied-states.

Unsupervised acoustic model adaptation (both means and variances) is performed for each cluster using the MLLR technique [32] after each decoding pass. The mean vectors are adapted using a single block-diagonal regression matrix (where a block is used for each parameter stream, i.e. cepstrum, delta-cepstrum and delta-delta cepstrum), and a diagonal matrix is used to adapt the variances. When less than 12 seconds of adaptation data are available, only diagonal matrices are used for both the means and the variances. A single regression matrix is used since we have never observed a gain using multiple regression matrices for unsupervised adaptation.

Position: state-position, word-begin, word-end, monophone
General classes: vowel, consonant, continuant, sonorant, voiced-consonant, voiceless, fricative, strident, stop, nasal, semivowel, aspirated, anterior, high, coronal, slack, rounded, tense, retroflex, syllabic, fillers
Vowel classes: high-vowel, low-vowel, rounded-vowel, tense-vowel, reduced, diphthong, front-vowel, back-vowel, long-vowel, short-vowel, retroflex-vowel, diphthong-F2up, diphthong-F2down
Consonant classes: labial, dental, alveolar, palatal, velar, affricate
Individual Phones: b, d, g, p, t, k, dʒ, tʃ, s, ʃ, z, ʒ, f, v, θ, ð, m, n, ŋ, ɱ, ɳ, l, ɭ, r, w, y, h, i, ɪ, e, ɛ, æ, ʌ, ʌʲ, ʌʷ, o, ɔ, ɔʲ, u, ʊ, ə, X, ʔ, Ɂ, [filler], [breath], [silence]

Table 3: Questions used for decision tree clustering concern the phone position and class, the distinctive features and the phone identity.

<i>question</i>	<i>% log likelihood gain</i>	<i>question</i>	<i>% log likelihood gain</i>
vowel[+1]	6.3%	phone-r[+1]	2.2%
sonorant[+1]	5.5%	phone-H[+1]	2.1%
sonorant[-1]	3.8%	strident[+1]	1.9%
front-vowel[+1]	3.6%	phone-l	1.8%
semivowel[+1]	3.6%	nasal[-1]	1.7%
voiced-consonant[+1]	3.1%	vowel[-1]	1.6%
wordbody-pos[0]	2.5%	high-vowel[+1]	1.5%
nasal[+1]	2.3%	voiceless[-1]	1.5%
voiceless[+1]	2.2%	phone-n[+1]	1.5%
wordbegin-pos[0]	2.2%	phone-s[+1]1]	1.4%

Table 4: The most frequently used decision tree questions. The [+1] and [-1] indicate that the question has been applied to the right or left context respectively, and [0] to the phone itself.

5 Language modeling

Different approaches for language model training were explored and tested in the context of a complete transcription system. Language model efficiency was investigated for the following aspects: mixing of different training material (sources and epoch); approach for mixing (interpolation vs count merging); and using class-based language models. The experimental results indicate that judicious selection of the training source and epoch is important, and that given sufficient broadcast news transcriptions, newspaper and newswire texts are not necessary. The combined improvements in text selection, interpolation, 4-gram and class-based LMs led to a 20% reduction in the perplexity of the LM of the final pass (3-gram class interpolated with a word 4-gram) compared with the 3-gram LM used in the LIMSI Nov'97 BN system.

5.1 Text normalization and wordlist selection

For transcription of American English Broadcast News shows, very large text corpora are available for constructing language models. Three different sources of data were used:

- NEWS: Over 700M words of news texts from various sources (newspapers and newswires from 1994 to 1998). These data, available through the LDC, consist of texts from the Los Angeles Times, New York Times, Wall Street Journal, Washington Post, Reuters News Service, and Associated Press WordStream.
- BNA: 1.5M words of accurate broadcast news transcripts of the acoustic training data. Non lexical items such as breath noise, hesitations, word fragments are transcribed.
- BNC: 200M words of commercial transcripts of various broadcast shows (from 1992 to 1998). These transcripts do not include extra-lexical events.

It should be noted that only a very small proportion of the LM data (about 2%) is truly representative of the real data to be transcribed.

The training texts were processed to clean errors inherent in the texts or arising from the preprocessing tools, and transformed to be closer to the observed American speaking style. The cleaning consisted primarily of correcting obvious misspellings (such as MILLION, OFFICALS, LITTLEKNOWN), systematic bugs introduced by the text processing tools, and expanding abbreviations and acronyms in a consistent manner. The texts were also transformed to be closer to the observed American reading style using a set of rules and the corresponding probabilities derived from the alignment of the WSJ0/WSJ1 prompt texts with the transcriptions of the acoustic data. Some example rules and their probabilities are shown in Table 5. The cleaning of the training texts reduced perplexity on development data in a better coverage for the 65k lexicon [22].

Filler words such as “uh” and “uhm” were mapped to a unique form. The training texts were processed in order to add a proportion of breath markers (4%), and of filler words (0.5%) [14]. While it would seem more elegant to incorporate these in the LM by interpolating LMs estimated on the clean text (without noises) and on the transcripts (with noises), adding them to the clean texts via a generation model resulted in a lower word error rate ($\sim 2\%$ relative). This result can be explained by the observation that breath noise and filler words do not occur at random, but at specific places. Adding them at such places in the clean texts is equivalent to adding a priori information about the distribution of these phenomena in the transcripts.

The training texts were also processed to treat the most common 1000 acronyms as distinct lexical entries [19] (as opposed to a sequence of individual letters) and to represent some frequent word sequences subject to reduction as compound words [14].

HUNDRED <nb>	⇒	HUNDRED AND <nb> (0.50)
ONE EIGHTH	⇒	AN EIGHTH (0.50)
CORPORATION	⇒	CORP. (0.29)
INCORPORATED	⇒	INC. (0.22)
ONE HUNDRED	⇒	A HUNDRED (0.19)
MILLION DOLLARS	⇒	MILLION (0.15)
BILLION DOLLARS	⇒	BILLION (0.15)

Table 5: Some example transformation rules with probabilities.

<i>4gram LM</i>	<i>Word Error rate</i>			<i>Perplexity</i>		
	<i>Eval96</i>	<i>Eval97</i>	<i>Eval98</i>	<i>Eval96</i>	<i>Eval97</i>	<i>Eval98</i>
NEWS	22.7	15.8	15.3	291.8	246.3	257.4
BNC+BNA	20.3	14.3	13.8	175.7	175.6	181.6
BNC+BNA+NEWS	20.0	14.0	13.6	167.4	163.3	168.8

Table 6: Word error rate and perplexity for LMs constructed on different sources (NEWS: newspaper & newswire, 340M words; BNA: accurate broadcast news transcripts, 1.5M words; BNC: commercial broadcast news transcripts, 200M words) on 3 evaluation data sets.

The recognition vocabulary (or word list) contains 65,122 words, and includes all words occurring a minimum of 15 times in the BNC (63,954 words) or at least twice in the BNA data (23,234 words). The lexical coverage was 99.14%, 99.53% and 99.73% on the eval96, eval97 and eval98 test sets respectively.

5.2 Combining data sources

One easy way to combine training material from different sources is to train an n -gram backoff LM per source and to interpolate them. The interpolation weights can be directly estimated on some development data with the EM algorithm. The resulting LM is a mixture of n -gram backoff LMs. An alternative is to simply merge the n -gram counts and train a single n -gram backoff language model on these counts. If some data sources are more representative than others for the task, the n -gram counts can be empirically weighted to minimize the perplexity on a set of development data. While this can be effective, it has to be done by trial and error and cannot easily be optimized. In addition, weighting the n -gram counts can pose problems in properly estimating the backoff coefficients. Using the three available data sources, we compared the two approaches on one hand by generating interpolated 4-gram backoff LMs and on the other hand by merging the n -gram counts with the manually optimized weights. The results obtained with word graph rescoring show that on 3 eval sets the approach which merged the n -gram counts had a slightly higher word error rate (0.2% absolute) 15.73% compared to 15.46%.

Two strategies were explored to add cross sentence trigram counts in the trigram model [39]: add the whole texts with and without sentences boundaries, and renormalize the counts; or add only the cross sentence trigrams. Both strategies led to similar results in terms of perplexity and recognition error. For the Nov'98 evaluation, the language models were constructed using the second approach.

Selecting the appropriate LM training material evidently affects the resulting LM accuracies. There is the sometimes conflicting need for sufficient amounts of text data to estimate LM parameters and assuring that the data is representative of the task. For instance, in [17] it was reported that, for the broadcast news transcription task, while the use of all the available newspaper data led to a small decrease in perplexity, it also led to a small increase in the recognition error rate. Therefore, all NEWS texts that did not lower the perplexity were eliminated.

To optimize the selection of texts for the LIMSI Nov'98 system, the newspaper and commercial transcription sources were split into 5 non-overlapping time periods, based on proximity to the test epoch (15oct96-14nov96). For each of these periods (jan94-sep95, oct95-jun96, jul96-feb97, mar97-aug97, sep97-dec97) separate LMs were constructed for each source. The interpolation coefficient for each component LM was optimized on the development data (containing shows recorded in oct96). LMs with very low interpolation coefficients were eliminated. Subsets with comparable interpolation coefficients (different sources or epochs) were merged in order to decrease the size of the resulting LM. Only very small variations in perplexity were observed during this process, and the final optimization resulted in interpolation of four 4-gram LMs, constructed on the following texts: BNC (200M words, interpolation coefficient 0.56); BNA (1.5M words, interpolation coefficient 0.22); NEWS period jan94-sep95 (200M words, interpolation coefficient 0.10); and NEWS period jul96-aug97⁴ (141 Mwords, interpolation coefficient 0.12). It can be noted that the weight of the BNA LM is equal to the weight of the NEWS LMs (0.22) even though the text is much smaller.

Some experiments were conducted in order to evaluate the influence of each source on the recognition word error rate. 4-gram LMs were constructed using the following data sets: NEWS only, BNC (0.75) + BNA (0.25), BNC (0.56) + BNA (0.22) + NEWS (0.22). The latter corresponds to the 4-gram used in the ARPA'98 evaluation. Recognition results obtained via word graph rescoring using these three LMs are summarized in Table 6 for the three eval data sets. The true differences between models may be slightly larger since all results used the same word graph generated with the BNC+NEWS+BNA LM. There is a large reduction both in perplexity and in word error rate when transcripts are used to train the LM, as opposed to NEWS texts. Interpolating the NEWS LM with the transcription based LM yields a small but consistent reduction in perplexity and word error. The combination of LMs estimated on commercially produced transcripts BNC and on accurate transcripts is quite performant. However, if commercial transcripts are not available, newspaper sources are a reasonable source of language model training data: although the LM constructed only on NEWS data has a perplexity 43% higher than BNC+BNA+NEWS, the recognition word error rate is only 11% higher.

6 Lexical Modeling

Lexical design entails selecting the vocabulary items and determining their pronunciation. The word list selection was discussed in the previous section, in this section we address pronunciation modeling. Our experience is that systematic lexical design can improve the overall system performance. The pronunciations are based on a 48 phone set (3 of them are used for silence, filler words, and breath noises) and include standard pronunciations but do not explicitly represent allophones. In order to better model the observed speaking styles in the Hub4 data, two phones were added to the LIMSI WSJ phone set [30] so as to explicitly model filler words and breath noises [14] without contaminating the other phones. A phonemic representation is used as most allophonic variants can be predicted by rules, and their use is optional. More importantly, there often is a continuum between different allophones of a given phoneme and the decision as to which occurred in any given utterance is subjective. By using a phonemic representation, no hard decision is imposed, and it is left to

⁴All data from the same period as the eval98 test set (15/10/96-14/11/96) was excluded.

WHAT_DID_YOU	wa{t}dIdyu wa{t}dIdyə wa{t}dIdʒə w[əə]dʒə
I_DON'T_KNOW	ɑ ^j don{t}no ɑ ^j d^no α ^j dno
DON'T_KNOW	don{t}no d^no
LET_ME	lɛtmi lɛmi
LET_HIM	lɛthIm lɛtM lɛm
I_AM	ɑ ^j æm ɑ ^j ə m α ^j m
GOING_TO	gə ŋt[uə] g[^c]nə

Figure 4: Some example compound words and their pronunciations. Original concatenated pronunciation (1st line) and reduced forms (2nd line). Phones in { } are optional, phones in [] are alternates.

the acoustic models to represent the observed variants in the training data. A pronunciation graph is associated with each word so as to allow for alternate pronunciations which may depend upon the following word context. Frequently occurring inflected forms were verified to provide more systematic pronunciations.

There are a variety of words for which frequent alternative pronunciation variants are observed, and these variants are not due to allophonic differences. One common example is the suffix “IZATION” which can be pronounced with a diphthong (/ɑ^j/) or a schwa (/ə/). Out of 7 occurrences of the word “INDUSTRIALIZATION” in the training data, 3 are pronounced with /ɑ^j/ and 4 with /ə/. Another pronunciation variant is the palatalization of the /k/ in a /u/ context, such as in the word “coupon” (/kupaⁿ/ vs. /kyupaⁿ/). Alternate pronunciations may also reflect different parts of speech (verb or noun) as in words like “excuse, record”.

It is well known that in fluent speech, certain common word sequences can be subject to severe reduction. One easy way to model such effects are to use compound words for frequent word sequences, which is a way of incorporating phonological rules on a very limited basis. The example spectrograms of sentences including the word sequence “what did you” shown in Figure 5 illustrate the need for pronunciation variants for spontaneous speech. In the first spectrogram, the speaker said all three words clearly and palatalized the /dy/ into a /dʒ/. In the second, the speaker produced a flap for the combined final /t/ in “what” and the initial /d/ in “did”. In the third example, the sequence was reduced to /w^dʒə/. The recognition lexicon contains entries for the most common 1000 acronyms found in the training texts and compound words for about 300 frequent word sequences. Some example compound words and their pronunciations are given in Table 4. The first line corresponds to the original pronunciation formed by concatenation of the component words. The second line contains reduced forms added for the compound word.

The pronunciations in our American English lexicon were created semi-automatically using a pronunciation generation tool [30]. When an unknown word is encountered, affix rules are applied to the entries in one or more lexicons in an attempt to derive a pronunciation. When multiple pronunciations can be derived they are presented for selection, along with their source. Although the LIMSI “Master” lexicon contains over 100k entries, when processing a new set of acoustic training data, we

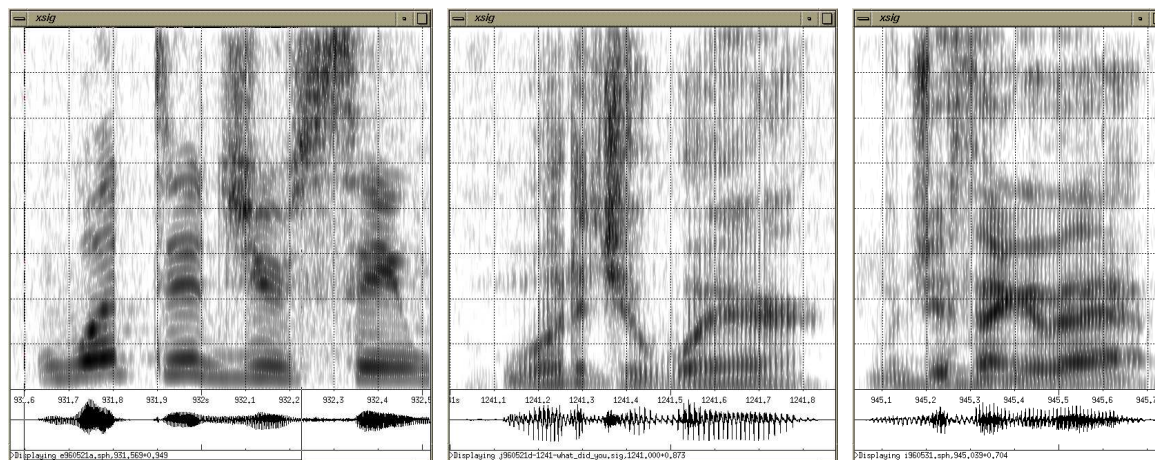


Figure 5: Spectrograms of the word sequences containing “what did”. “what did you see” (file e960521a), “what did you wear” (file j960521d), “what did you think of that” (file i960531).

generally need to add new words. These are often times proper names (which are difficult to generate automatically) and word fragments, which need to be included in a training lexicon even though they are not usually present in a recognition lexicon. When proper names appear in the training data, their pronunciations are manually verified.

7 Word Decoding

One of the most important problems in implementing the decoder is the design of an efficient search algorithm to deal with the huge search space, especially when using language models with a longer span than two successive words, such as 3-grams and 4-grams. Many potential applications making use of broadcast news transcriptions do not require on-line processing. Batch processing offers a substantial advantage as all of the data for a given show can be used for unsupervised model adaptation, resulting in significant improvement in recognition accuracy. Multiple pass decoders are well adapted to broadcast news transcription, where a first decoding pass can be used to generate a word hypothesis which is then used for model adaptation. While this approach has been very successful for acoustic model adaptation, to date attempts to adapt the language models have been less rewarding.

7.1 Baseline decoder

The two-step approach used in the LIMSI Nov’98 system transmits information between levels via word graphs [21]. Due to memory constraints, each step may consist of one or more passes, each using successively more refined models. All decoding passes use cross-word CD triphone models. In order to generate accurate word graphs, cluster-based model adaptation is carried out using an initial hypothesis. It is clear that this type of adaptation cannot be used in a real-time system, but is applicable to batch processing of data, which could occur immediately after the data is broadcast.

The word decoding procedure is shown in Figure 6. Prior to decoding, segments longer than 30s are chopped into smaller pieces so as to limit the memory required for the 3-gram and 4-gram decoding passes [14]. To do so a bimodal distribution is estimated by fitting a mixture of 2 Gaussians to the log-RMS power for all frames of the segment. This distribution is used to determine locations which are likely to correspond to pauses, thus being reasonable places to cut the segment. Cuts are made at the most probable pause 15s to 30s from the previous cut. Word recognition is performed in

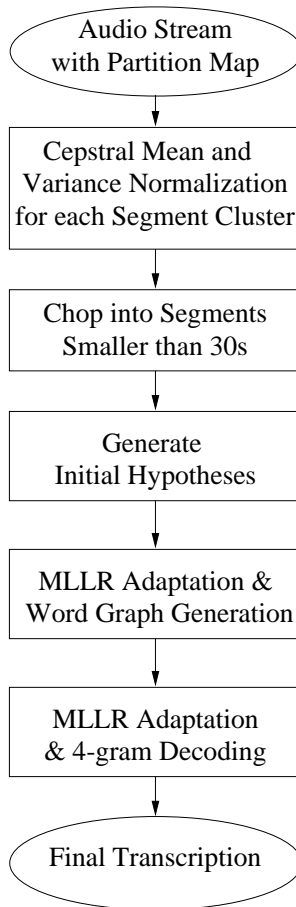


Figure 6: Word decoding.

three steps: 1) initial hypothesis generation, 2) word graph generation, 3) final hypothesis generation, each with two passes.

Step 1: Initial Hypothesis Generation This step, carried out in two passes, generates initial hypotheses which are used for cluster-based acoustic model adaptation. The first pass of this step generates a word graph using a small bigram backoff language model and gender-specific sets of 5416 position-dependent triphones with about 11500 tied states. This is followed by a second decoding pass with a larger set of acoustic models (27506 triphones with 11500 tied states) and a trigram language model (about 8M trigrams and 15M bigrams) to generate the hypotheses. Band-limited acoustic models are used for the telephone speech segments.

Step 2: Word Graph Generation Unsupervised acoustic model adaptation (both means and variances) is performed for each segment cluster using the MLLR technique [32]. The mean vectors are adapted using a single block-diagonal regression matrix, and a diagonal matrix is used to adapt the variances. Each segment is decoded first with a bigram language model and an adapted version of small set of acoustic models, and then with a trigram language model (8M bigrams and 17M trigrams) and adapted versions of the larger acoustic model set.

Step 3: Final Hypothesis Generation The final hypothesis is generated using a 4-gram interpolated with a category trigram model with 270 automatically generated word classes [27]. The first pass of this step uses the large set of acoustic models adapted with the hypotheses from Step 2, and a 4-gram language model. This hypothesis is used to adapt the acoustic models prior to the final

decoding step with the interpolated category trigram model.

<i>System</i>	<i>Test set (Word Error)</i>		
	<i>Eval96</i>	<i>Eval97</i>	<i>Eval98</i>
<i>Nov96 system</i>	27.1*		
<i>Nov97 system</i>	25.3	18.3	
<i>Nov98 system</i>	19.8	13.9	13.6

Table 7: Summary of BN transcription word error rates. *Nov96 system used a manual partition.

Table 7 reports the word recognition results on the eval test sets from the last three years. All of our system development was carried out using the eval96 data. The results shown in bold are the official NIST scores obtained by the different systems. Only the Nov96 system used a manual partition. In Nov97 our main development effort was devoted to moving from a partitioned evaluation to the unpartitioned one. The Nov97 system did not use focus-condition specific acoustic models as had been used in the Nov96 system. This system nevertheless achieved a performance improvement of 6% on the eval96 test data. The Nov98 system has more accurate acoustic and language models, and achieves a relative word error reduction of over 20% compared to the Nov97 system.

Table 8 gives the word error rates for the Nov98 system after each decoding step on the same three eval sets. The first decoding step that is used to generate the initial hypothesis runs in about 35xRT and has a word error of 25% on the eval96 data, and 18% on the eval97 and eval98 sets. A word error reduction of about 20% is obtained in the second decoding step which uses the adapted acoustic models and runs in about 130xRT. Relatively small gains are obtained in the 4-gram decoding pass (30xRT), even though these also include an extra acoustic model adaptation. The runs were done on Silicon Graphics Origin200, R10K processor running at 180MHz and with 1Gb memory. These processing times are only indicative as no effort was made to optimize the computation means, other than to fit within what was available.

7.2 10xRT decoder

In 1999 our goal was to achieve comparable performance with a decoding time of under 10x real-time. To reach this goal, a 4-gram single pass dynamic network decoder was developed [16]. It is a time-synchronous Viterbi decoder with dynamic expansion of LM state conditioned lexical trees [11, 34, 33] with acoustic and language model lookaheads. The decoder can handle position-dependent, cross-word triphones and lexicons with contextual pronunciations. It makes use of various pruning techniques to reduce the search space and computation time, including three HMM-state pruning

<i>System Step</i>	<i>Test set (Word Error)</i>		
	<i>Eval96</i>	<i>Eval97</i>	<i>Eval98</i>
<i>Step1 3-gram</i>	25.30	18.44	18.31
<i>Step2 3-gram</i>	20.95	14.56	14.24
<i>Step3 4-gram</i>	20.23	14.26	13.66
<i>4-gram class</i>	19.79	13.92	13.56

Table 8: Word error rates after each decoding step with the Nov98 system.

	<i>Pass</i>	<i>AM</i>	<i>LM</i>	<i>Time</i>	<i>Total time</i>	<i>Werr</i>
A	1	92k	3g	6.8xRT	6.8xRT	16.8%
B	1	350k	4g	10.8xRT	10.8xRT	15.9%
	1	92k	3g	0.8xRT		24.7%
C	2	350k+mllr	4g	9.9xRT	10.7xRT	14.6%
	1	92k	3g	0.8xRT		24.7%
D	2	350k+mllr	3g	6.1xRT	6.9xRT	15.4%
E	3	350k+mllr	4g	1.5xRT	8.4xRT	14.2%

Table 9: Comparison of decoding strategies on the NIST Hub4 eval98 set (partitioning and coding times are not included).

beams and fast Gaussian likelihood computations. It can also generate word graphs and rescore them with different acoustic and language models. Faster than real-time decoding can be obtained using this decoder with a word error under 30%, running in less than 100 Mb of memory on widely available platforms such as Pentium III or Alpha machines.

The decoder by itself does not solve the problem of reducing the recognition time as proper models have to be used in order to optimize the recognizer accuracy at a given decoding speed. In general, better models have more parameters, and therefore require more computation. However, since the models are more accurate, it is often possible to use a tighter pruning level (thus reducing the computational load) without any loss in accuracy. Thus, limitations on the available computational resources can significantly affect the design of the acoustic and language models. For each operating point, the right balance between model complexity and pruning level had to be found.

Table 9 gives the computation time and word error rates for various decoding strategies, using the Hub4 eval98. The pruning thresholds have been set so as to match the computing time of the most interesting setups. Each entry specifies the acoustic and language models used in the pass and the computation time. All passes perform a full decode, except the last decoding pass (labelled E) which is a word graph rescoring using a graph generated in the second 3-gram pass. These results clearly demonstrate the advantage of using a multiple pass decoding approach. Comparing the setups A (1 pass, 6.8xRT, 16.8%) and D (2 passes, 6.9xRT, 15.4%), the extra computation time needed for the first decode and the MLLR adaptation in D is largely compensated by the reduction in word error rate. Using adapted acoustic models allows us to use a tighter pruning threshold and have the same overall computing time but with a significantly lower word error rate. Also by comparing the setups C (2 passes, 10.7xRT, 14.6%) and E (3 passes, 8.4xRT, 14.2%) the advantage of using an extra decoding pass with the 4-gram LM and the 2nd pass hypotheses for the MLLR adaptation can be seen.

For reference, the official result on the eval98 test set using our Nov98 system was 13.6%, with a decoding time around 200xRT [20]. Using only the first decoding pass, unrestricted BN data can be decoded in less than 1.4xRT (including partitioning) with a word error rate around 30%. The same decoding strategy has been successively applied to the BN transcription in other languages (French, German and Mandarin) with comparable word error rates.

8 Perspectives and Conclusions

In this paper we have summarized our recent activities aimed at transcribing radio and television broadcasts. Most of this work has been carried out for the American English language in the context of developing systems for the annual DARPA benchmark tests. This framework has provided the

training materials (transcribed audio and textual corpora for training acoustic and language models), test data and a common evaluation framework. In the context of the LE-4 OLIVE project the LIMSI transcription system has been ported to the French and German languages, which has required a large investment in data collection.

Partitioning and transcribing television and radio broadcasts are necessary steps to enable automated processing of the vast amounts of audio and video data produced on a daily basis. The data partitioning algorithm makes use of Gaussian mixture models and an iterative segmentation and clustering procedure. The resulting segments are labeled according to gender and bandwidth. Many of the errors occur at the boundary between segments, and can involve silence segments which can be considered as with speech or non-speech without influencing transcription performance. Based on our experience, it appears that current word recognition performance is not critically dependent upon the partitioning accuracy.

Acoustic training on broadcast data is significantly more complicated than on read speech corpora like the Wall Street Journal corpus. Even when divided into speaker turns, segments can be quite long - several minutes in duration. Aligning even a perfect transcription with the signal can be difficult, and any minor problem may cause the alignment to fail [36]. Splitting long segments at silences is a possible solution, but requires manual intervention.

Explicitly modeling the NIST focus conditions is probably not worth the additional effort and complexity in training and decoding. However, the focus conditions are quite interesting as a factor for error analysis. In addition, some of the distinctions are clearly unrealistic to automatically detect, such as the distinction between read and spontaneous broadcast quality speech, or reliable detection of non-native speech. The wideband / telephone-band distinction can be made with reasonable accuracy, and using narrow-band models improves the relative performance on telephone data by about 10%.

Given the large amount of acoustic training data available for American English, it is possible to properly model many different triphone contexts with a very high coverage of over 99%. Tied-state acoustic models are efficient for reducing the number of parameters to be estimated. Different approaches for state-tying were investigated. Although comparable model sets were obtained using bottom-up agglomerative clustering and top-down decision tree clustering, the latter approach is much faster and thus shortens the development cycle.

Cepstral mean normalization and acoustic model adaptation are important techniques given the non-homogeneous nature of broadcast data. Both of these are cluster-based for the test data, allowing a better estimate of the speaker characteristics and acoustic environment.

The generation of word graphs with adapted acoustic models using an initial hypothesis obtained in a rapid decoding pass is essential for obtaining word graphs with low word error rates. Unsupervised HMM adaptation is performed prior to each decoding pass using the hypothesized transcription of the previous pass. This strategy leads to a significant reduction in word error rate.

Concerning language model development, the contributions of the various text sources were evaluated. It was determined that the transcriptions of broadcast data (both detailed acoustic and commercial transcripts) are by far the most important sources, and that newspaper and newswire texts are not very helpful should other closer sources such as commercial transcripts be available. Another potential source of related texts are closed captions, which have been explored in the context of the OLIVE project. However our initial experience is that the closed captions used a stylized language which is relatively limited compared to the true transcripts, and thus are less appropriate than commercial transcripts. We have also experimented with different approaches to combining data from different sources, based on count merging and LM interpolation. Interpolation is a very powerful approach allowing optimal combination of component LMs estimated on different text sources.

The overall word transcription error of the Nov98 unpartitioned evaluation test data (3 hours) was 13.6%. Although substantial performance improvements have been obtained, there is still plenty of

room for improvement of the underlying speech recognition technology. On unrestricted broadcast news shows, such as the 1996 dev and eval data, the word error rate is still about 20% (even though the NIST scoring program has removed overlapping speech).

With the rapid expansion of different media sources for information dissemination, there is a pressing need for automatic processing of the audio data stream. A variety of near-term applications are possible such as audio data mining, selective dissemination of information, media monitoring services [1], disclosure of the information content [4] and content-based indexation for digital libraries [3]. Although substantial performance improvements have been obtained over the last 4 years, there is still a need to improve the underlying speech recognition technology so as to increase the recognition accuracy and reduce the required processing time [2].

9 Acknowledgements

The authors acknowledge the participation of Martine Adda-Decker to the Nov'97 system and of Michèle Jardino in estimating the word classes.

References

- [1] <http://www.fb9-ti.uni-duisburg.de/alert/>
- [2] <http://coretex.itc.it/>
- [3] <http://pc-erato2.iei.pi.cnr.it/echo/>
- [4] <http://twentyone.tpd.tno.nl/olive/>
- [5] *Proc. ARPA Spoken Language Systems Technology Workshop*, Austin, TX, San Francisco: Morgan Kaufmann, January 1995.
- [6] *Proc. DARPA Speech Recognition Workshop*, Arden House, NY, San Francisco: Morgan Kaufmann, February 1996.
- [7] *Proc. DARPA Speech Recognition Workshop*, Chantilly, VA, San Francisco: Morgan Kaufmann, February 1997. <http://www.nist.gov/speech/publications/darpa97/index.htm>
- [8] *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, Landsowne, VA, San Francisco: Morgan Kaufmann, February 1998. <http://www.nist.gov/speech/publications/darpa98/index.htm>
- [9] *Proc. DARPA Broadcast News Workshop*, Herndon, VA, San Francisco: Morgan Kaufmann, February 1999. <http://www.nist.gov/speech/publications/darpa99/index.htm>
- [10] *Proc. 2000 Speech Transcription Workshop*, College Park, MD, May 2000. <http://www.nist.gov/speech/publications/tw00/index.htm>
- [11] X. Aubert, "One Pass Cross Word Decoding for Large Vocabularies Based on a Lexical Tree Search Organization," *Proc. ESCA Eurospeech'99*, 4, pp. 1559-1562, Budapest, Hungary, September 1999.
- [12] S.S. Chen, P.S. Gopalakrishnan, "Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion", *Proc. DARPA Broadcast News Transcription & Understanding Workshop*, pp. 127-132, February 1998.

- [13] P. Clarkson, R. Rosenfeld, "Statistical Language modelling using CMU-Cambridge Toolkit," *Proc. ESCA EuroSpeech'97*, Rhodes, Greece, pp. 2707-2710, September 1997.
- [14] J.L. Gauvain, G. Adda, L. Lamel, M. Adda-Decker, "Transcribing Broadcast News: The LIMSI Nov96 Hub4 System," *Proc. ARPA Speech Recognition Workshop*, Chantilly, VA, pp. 56-63, February 1997.
- [15] J.L. Gauvain, G. Adda, L. Lamel, M. Adda-Decker, "Transcription of Broadcast News," *Proc. ESCA EuroSpeech'97*, **2**, pp. 907-910, Rhodes, Greece, September 1997.
- [16] J.L. Gauvain, L. Lamel, "Fast Decoding for Indexation of Broadcast Data," *Proc. ICSLP'2000*, **3**, pp. 794-798, Beijing, China, October 2000.
- [17] J.L. Gauvain, L. Lamel, G. Adda, "The LIMSI 1997 Hub-4E Transcription System", *Proc. DARPA Broadcast News Transcription & Understanding Workshop*, pp. 75-79, Landsdowne, VA February 1998.
- [18] J.L. Gauvain, L. Lamel, G. Adda, "Partitioning and Transcription of Broadcast News Data," *Proc. ICSLP'98*, **5**, pp. 1335-1338, Sydney, Australia, December 1998.
- [19] J.L. Gauvain, L. Lamel, G. Adda, D. Matrouf, "The LIMSI 1995 Hub3 System," *Proc. DARPA Speech Recognition Workshop*, Arden House, NY, pp. 105-111, February 1996.
- [20] J.L. Gauvain, L. Lamel, G. Adda, M. Jardino, "The LIMSI 1998 Hub-4E Transcription System", *Proc. DARPA Broadcast News Workshop*, pp. 99-104, Hernon, VA, February 1999.
- [21] J.L. Gauvain, L. Lamel, M. Adda-Decker, "The LIMSI Nov93 WSJ System *Proc. ARPA Spoken Language Technology Workshop*, Princeton, NJ, March, 1994.
- [22] J.L. Gauvain, L. Lamel, M. Adda-Decker, "Developments in Continuous Speech Dictation using the ARPA WSJ Task," *Proc. IEEE ICASSP-95*, Detroit, pp. 65-68, May 1995.
- [23] J.L. Gauvain, C.H. Lee, "Maximum *a Posteriori* Estimation for Multivariate Gaussain Mixture Observation of Markov Chains," *IEEE Trans. on SAP*, **2**(2), pp. 291-298, April 1994.
- [24] H. Gish, M. Siu, R. Rohlicek, "Segregation of Speakers for Speech Recognition and Speaker Identification," *Proc. IEEE ICASSP-91*, pp. 873-876, May 1991.
- [25] T. Hain, S.E. Johnson, A. Tuerk, P.C. Woodland, S.J. Young, "Segment Generation and Clustering in the HTK Broadcast News Transcription System," *Proc. DARPA Broadcast News Transcription & Understanding Workshop*, pp. 133-137, Landsdowne, VA, February 1998.
- [26] H. Hermansky, "Perceptual Linear Prediction (PLP) Analysis for Speech," *J. Acouts. Soc. Amer.*, Vol. 87, pp. 1738-1752.
- [27] M. Jardino "Multilingual stochastic n-gram class language models," *Proc. IEEE ICASSP-96*, Atlanta, GA, **I**, pp. 161-164, May 1996.
- [28] A. Kannan, M. Ostendorf, J.R. Rohlicek, "Maximum Likelihood Clustering of Gaussians for Speech Recognition," *IEEE Trans. Speech & Audio*, **2**(3), July 1994.
- [29] F. Kubala, T. Anastasakos, H. Jin, J. Makhoul, L. Nguyen, R. Schwartz, N. Yuan, "Toward Automatic Recognition of Broadcast News," *Proc. DARPA Speech Recognition Workshop*, Arden House, NY, pp. 55-60, February 1996.

- [30] L. Lamel and G. Adda, "On Designing Pronunciation Lexicons for Large Vocabulary, Continuous Speech Recognition," *Proc. ICSLP'96*, **1**, pp. 6-9, Philadelphia, PA, October 1996.
- [31] D. Liu and F. Kubala, "Fast Speaker Change Detection for Broadcast News Transcription and Indexing," *Proc. ESCA EuroSpeech'99*, Budapest, **3**, pp. 1031-1034, Hungary, September 1999.
- [32] C.J. Leggetter, P.C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech & Language*, **9**(2), pp. 171-185, 1995.
- [33] H. Ney, R. Haeb-Umbach, B.H. Tran and M. Oerder, "Improvements in Beam Search for 10000-Word Continuous Speech Recognition," *Proc. IEEE ICASSP-92*, **1**, pp. 9-12, San Francisco, CA, March 1992.
- [34] J.J. Odell, V. Valtchev, P. Woodland, S. Young, "A One Pass Decoder Design for Large Vocabulary Recognition," *Proceedings ARPA Workshop on Human Language Technology*, pp. 405-410, Plainsboro, NJ, March 1994.
- [35] D.B. Paul and J.M. Baker (1992), "The Design for the Wall Street Journal-based CSR Corpus," *Proc. ICSLP-92*, Banff, **2**, pp. 899-902, October 1992.
- [36] M. Pitz, S. Molau, R. Schlüter, H. Ney, "Automatic Transcription Verification of Broadcast News and Similar Speech Corpora," *Proc. DARPA Broadcast News Workshop*, pp. 157-159, Herndon, VA, February 1999.
- [37] T. Robinson, J. Fransen, D. Pye, J. Foote, S. Renals, "WSJCAM0: A British English Speech CORpus for Large Vocabulary Continuous Speech Recognition," *Proc. IEEE ICASSP-95*, Detroit, MI, **1**, pp. 81-84, May 1995.
- [38] R. Schwartz, H. Jin, F. Kubala, S. Matsoukas, "Modeling Those F-Conditions – Or Not," *Proc. DARPA Speech Recognition Workshop*, Chantilly, VA, pp. 115-118, February 1997.
- [39] K. Seymore, S. Chen, M. Eskenazi, R. Rosenfeld, "Language and Pronunciation Modeling in the CMU 1996 Hub4 Evaluation," *Proc. DARPA Speech Recognition Workshop*, Chantilly, VA, pp. 141-146, February 1997.
- [40] M. Siegler, U. Jain, B. Raj, R. Stern, "Automatic Segmentation, Classification and Clustering of Broadcast News Audio," *Proc. DARPA Speech Recognition Workshop*, Chantilly, VA pp. 97-99, February 1997.
- [41] R. Stern et al., "Specification for the ARPA November 1996 Hub 4 Evaluation," November 1996.
- [42] S. Wegmann, F. Scattoni, I. Carp, L. Gillick, R. Roth, J. Yamron, "Dragon Systems' 1997 Broadcast News Transcription System," *Proc. DARPA Broadcast News Transcription & Understanding Workshop*, pp. 60-65, Landsdowne, VA, February 1998.
- [43] S. Wegmann, P. Zhan, L. Gillick, "Progress in Broadcast News Transcription at Dragon Systems," *Proc. IEEE ICASSP'99*, pp. 33-36, Phoenix, AZ, March 1999.
- [44] P.C. Woodland, T. Neider, E. Whittaker, "Language Modeling in the HTK Hub5 LVCSR", presented at the 1998 Hub5E Workshop, September 1998.

- [45] S.J. Young, M. Adda-Decker, X. Aubert, C. Dugast, J.L. Gauvain, D.J. Kershaw, L. Lamel, D.A. Leeuwen, D. Pye, A.J. Robinson, H.J.M. Steeneken, P.C. Woodland, "Multilingual large vocabulary speech recognition: the European SQALE," *Proc. Computer, Speech and Language*, **11**(1), pp. 73-89, Jan. 1997.