TRANSCRIPTION OF BROADCAST NEWS - SYSTEM ROBUSTNESS ISSUES AND ADAPTATION TECHNIQUES

Raimo Bakis, Scott Chen, Ponani Gopalakrishnan, Ramesh Gopinath, Stéphane Maes, and Lazaros Polymenakos

Human Language Technologies, Computer Science Dept.,
IBM T. J. Watson Research Center, Yorktown Heights, NY.,
email:rameshg@watson.ibm.com, phone: (914)-945-2794

ABSTRACT

This paper describes some of the main problems and issues specific to the transcription of broadcast news and describes some of the methods for solving them that have been incorporated into the IBM Large Vocabulary Continuous Speech Recognition System.

1. INTRODUCTION

Significant advances in speech recognition technology have been achieved recently, as seen on tests conducted with read speech corpora such as the Wall Street Journal corpus [1]. The focus of research has shifted recently to transcription of "found" speech like radio/TV broadcast news. Transcription of broadcast news presents technical challenges arising from several sources of signal variability. A typical broadcast news segment contains speech and non-speech data from several sources, such as the signature tune of the show, interviews with people on location - possibly under very noisy conditions - and interviews over the telephone, commercials, etc. Broadly speaking, the data in such broadcasts can be characterized using three criteria: the quality of the microphone or channel, the characteristics of the speaker, and the condition of the background. The signal might be acquired using a high quality microphone, a low bandwidth microphone, or could be telephone quality. The speaker may be an experienced announcer or correspondent or an inexperienced speaker.The speech from the former appears similar to read speech, whereas the latter produces largely spontaneous speech. The background may contain music, noise, or other interfering speech. In some cases, there is no speech present - the signal might consist of a musical interlude or an extended period of noise such as street noises added to evoke an environment.

Decoding this data with a system trained on a clean training corpus such as the Wall Street Journal gives very high error rates It is necessary to develop new techniques to deal with such data. Preliminary ideas along these lines were explored in the IBM system used in the ARPA sponsored, November 1995 Hub4 radio broadcast news transcription task. Error rates dropped from 47% to 27% on the 1995 Hub4 evaluation test data [5, 7, 8]. This paper describes continuing work on the various problems encountered and the solutions attempted for transcription of broadcast news.

The basic philosophy is to first try and identify the segments of input data that belong to one of several classes and use separate modeling techniques appropriate for each class. For instance, segments detected as pure music are discarded and not decoded, segments identified as telephone quality speech are decoded by a system trained on telephone bandwidth speech, and so on. In the following sections, we describe techniques to handle issues in each class.

A brief description of our base recognition system follows (see [2, 4, 3] for details). The system uses acoustic models for sub-phonetic units with context-dependent tying. The instances of context dependent sub-phone classes are identified by growing a decision tree from the available training data [2] and specifying the terminal nodes of the tree as the relevant instances of these classes. The acoustic feature vectors that characterize the training data at the leaves are modeled by a mixture of Gaussian pdf's, with diagonal covariance matrices. The HMM used to model each leaf is a simple 1-state model, with a self-loop and a forward transition.

The training data used for the models in this paper comes from the following sources: WSJ-SI284 [5], MP-10 [5], BN-87 (the official 1996 Hub4 evaluation training data distribution consisting of 30 hours of broadcast shows from radio and TV) The test data is from one of the following sources: Dev95H4 (1995 Hub4 development test data), Eval95H4 (1995 Hub4 evaluation test data) and Dev96H4 (1996 Hub4 development test data). Unlike Dev95H4 and Eval95H4 test data, Dev96H4 data is distributed with class information (prepared clean - F0, spontaneous clean - F1, low fidelity - F2, music corrupted - F3, noisy - F4, non-native - F6, others - FX) allowing one to use it in a *partitioned mode* (manual segmentation followed by decoding) or *unpartitioned mode* (i.e., automatic segmentation followed by decoding). The language model used for all experiments in this paper (unless otherwise stated) is the one described in [5].

Section 2. describes the segmentation and classification scheme and Section 3. the models for the various conditions used in our experiments. The baseline models will be called M94 (trained on WSJ-SI284 [6]), M95c, M95m and M95t (trained by MAP adaptation [10] of M94 on clean, music-corrupted and telephone speech portions of MP-10 respectively [5]).

2. SEGMENTATION AND CLASSIFICATION

First, the distribution of feature vectors for each condition is modeled as a Gaussian mixture [5] trained from hand-labeled data from the MP-10 and BN-87 databases. For each feature vector x_t, and model M_j for condition j, $P(x_t/M_j)$ gives the likelihood of the frame coming from j. Since the condition is typically stable for a duration of a second or so, one imposes a minimum-length constraint on the segments. This is done by assuming a hidden Markov model for the generation of the input data as shown in Fig. 1. The j^{th} path in the model corresponds to the input data belonging to the j^{th} class, and the probability distribution of the arcs $c_{j,1} - c_{j,N}$ is given by M_j. The minimum length constraints on the segments are imposed by constraining the minimum length of the paths. The Viterbi algorithm is used to trace a path through the trellis corresponding to the model in Fig. 2., and to assign a class id to contiguous sets of the input feature vectors.

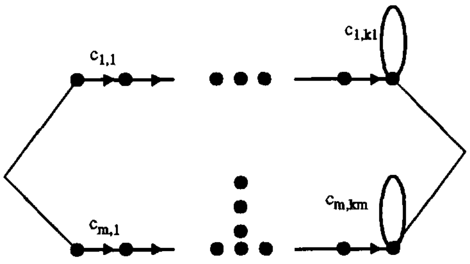

Fig. 1

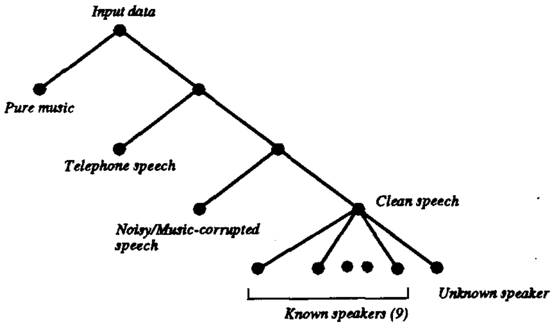

Fig. 2

For the Dev95H4 test data conditions were separated one at a time, i.e. first pure music segments, then telephone segments, and then music-corrupted segments were identified and separated. Finally one is left with clean speech. This organization enables use of different feature spaces for each binary classification problem. For instance, the feature space used to model the pure music segments was the 60-dimensional feature space that was also used for decoding. The feature space used to model telephone-speech was 24-dimensional cepstra augmented with their first and second differences to make up a 72-dimensional feature vector. Table 1 shows the performance of this segmentation algorithm on Dev95H4.

Table 1

Class	Corr	Miss%	Err%
Music	163.53	9.2	5.3
Telephone	766.62	0.13	4.2
Music & speech	308.66	2.8	39.6
Correct speaker	1185.96	17.3	13.6

On Dev96H4, a different strategy was used. Only linear *HMM* with 72 dimensional feature vectors were used. The separation between classes is no longer done in a binary fashion. All the gaussians are trained on the corresponding condition in BN-87 training data, except for the first level of extraction of clean data which was trained on MP-10. The training data in MP-10 was carefully hand-tagged so that no distorted speech was tagged as clean. Using BN-87 clean data resulted in often classification of clean speech as noisy speech or even speech plus music.

It was observed on Dev95H4 as well as Dev96H4 that this strategy sometimes tags telephone segments as pure music or music plus speech. Therefore, after segmentation, music, music plus speech and telephone segments are further classified into *BL* (bandlimited) - and *NBL* (non-bandlimited) using the system previously described. Music.*BL*, telephone.*BL* and music+speech.*BL* segments are tagged as telephone.*BL*, music.*NBL* segments are considered as pure music and speech+music.*NBL* is considered as speech+music. On Dev96H4 development data, this strategy takes care of such observed misclassifications of long telephone segments.

3. CLASS-SPECIFIC MODELS

3.1. Robust Model

By using all the data in BN-87 that excludes music-corrupted speech (F3) and low-fidelity speech (F2) we build a "conglomerate" model (M96ALL) from scratch (i.e., MP-10 and WSJ-SI-284 data were not used to build this model) that is quite robust to most conditions as the results below show.

3.2. Clean Speech

Of the Dev96H4 data about 18% is clean and prepared speech (Dev96H4-F0). For this data the system was built by MAP adaptation of the clean speech model used in [5] (which was obtained by MAP using MP-10 on the WSJ-SI-284 models) using the BN-87 F0 data. Unsupervised adaptation is performed on this data depending on duration d (in seconds) as follows: if $d < 1$ no adaptation is done, if $1 \leq d < 10$ 1 iteration of MLLR is done, if $10 \leq d < 30$ 3 iteration of MLLR is done followed by ABC adaption (described below), if $d \geq 30$ the CT adaption technique described in [13] is used. Some of performance results on Dev96H4-F0 are described in Table 2.

Table 2

Data	Model	WER
Dev96H4-F0	M95c	16.6%
Dev96H4-F0	M96F0	14.8%
Dev96H4-F0	M96F0+Adaptn.	14.0%

3.3. Spontaneous Speech

For spontaneous speech the models are obtained by MAP adaptation of M95c models using BN-87 F0 and F1 data. Further unsupervised adaptation is done using iterative MLLR. Nearly 25% of the Dev96H4 data is spontaneous speech (Dev96H4-F1). Performance on this data is given in Table 3.

Table 3

Data	Model	WER
Dev96H4-F1	M95c	42.0%
Dev96H4-F1	M96FALL	38.8%
Dev96H4-F1	M96F1	38.8%

3.4. Music Corrupted Speech

Table 4

Data	Model	WER
Dev96H4-F3	M96FALL	38.8%
Dev96H4-F3	M96F3	38.8%
Dev96H4-F3++	M96F3	37.5%

The clean WSJ SI-284 training data is transformed to be close to the test acoustic space by digitally adding pure music samples of various types from BN-87. This transformed training data is used to train music-corrupted models (M96F3) that are then MAP adapted using music-corrupted broadcast BN-87 training data. The silence models were augmented using gaussian mixtures modeling pure music data (M96F3++). On Dev96H4-F3 data (which is about 7% of the Dev96H4 data) the WER is given in Table 4.

3.5. Noise Corrupted Speech

Close to 13% of the data in Dev96H4 is noise-corrupted speech data (Dev96H4-F4). The models (M96F4) were built using map adaptation of the clean models used in the 1994 Hub4 evaluation [5] using data from BN-87 corresponding to the noise or F4 class. To increase the length of the segments used for unsupervised adaptation, segments with similar acoustic properties are aggregated with a *VQ* classification of the feature vectors. It essentially aggregates same speakers or same *SNR* when noise dominates. Adaptation is done using iterative MLLR on the base model. The final models are further improved by exploiting the correlation between HMM-states to better predict the gaussians with little or no adaptation data from those with sufficient adaptation data. This technique is called adaptation by correlation (ABC) and will be described in a later paper. The results on Dev96H4-F4 are shown in Table 5.

Table 5

Data	Model	WER
Dev96H4-F4	M96FALL	26.5%
Dev96H4-F4	M96F4	26.3%
Dev96H4-F4	M96F4+MLLR	24.1%
Dev96H4-F4	M96F4+MLLR+ABC	23.6%

3.6. Telephone Bandwidth Speech

About 18% of the data in Dev96H4 is low-fidelity speech data (Dev96H4-F2). This is typically composed of mainly telephone and some non-telephone data. M95t and M95m models give *WERs* of 59.95% and 57.7% respectively. MAP adaptation on the former degrades performance significantly while on the latter (using BN-87 data corresponding to the F4 and F2 classes) it reduces the *WER* to 50%, with large variability in performances across segments of the test data. Clearly telephone and non-telephone low fidelity data must be treated separately. Using gaussian mixture models for these two classes and the segmentation algorithm described earlier, BN-87 F2 training data is separated into *BL* (bandlimited) and *NBL* (non-band-limited) segments. The Gaussians where trained on MP-10 where the telephone tags always correspond to *BL*. On the Dev96H4-tele, the classification has no error. Both classes were roughly of equal size. Starting respectively from the M95t models and M96m models we performed MAP adaptation using respectively the F2.BL and F2.NBL+F4 portions of BN-87. The results are summarized in Table 6. The combined error rate dropped to 43%. Note that most of Dev96H4-F2.BL segments are spontaneous, explaining the high error rate.

Table 6		
Data	Model	WER
Dev96H4.F2.BL	M95t	61.0%
Dev96H4.F2.NBL	M95t	52.0%
Dev96H4.F2.BL	M95m	70.0%
Dev96H4.F2.NBL	M95m	52.0%
Dev96H4.F2.NBL	M96F2NBL	27.0%
Dev96H4.F2.BL	M96F2BL	59.8%

3.7. Speech From Non-Native Speakers

Nearly 9% of the data in Dev96H4 are from non-native speakers (Dev96H4-F5). When decoded with the 1995 Hub4 baseline system described in [5] the *WER* is 37.5%. We built a model (M96F5) by MAP adaptation using data from BN-87 corresponding to the non-native speakers - both prepared and spontaneous. Furthermore, unsupervised iterative MLLR adaptation is applied using scripts from an initial decoding. Despite the difference in vocabulary size (1700) for IBM96-NN, the improvements are comparable. The results on Dev96H4-non-native are summarized in Table 7.

Table 7		
Data	Model	WER
Dev96H4-F5	M95c	37.5%
Dev96H4-F5	M96ALL	28.7%
Dev96H4-F5	M96F5	26.2%
Dev96H4-F5	M96F5+MLLR	20.7%

4. MUSIC SUPPRESSION

We attempted to suppress steady notes due to music, while preserving the faster-varying speech frequency components, by calculating the rate-of-change of the dominant frequency in local areas of the short-term power spectrum. We trained this algorithm by means of a novel numerical optimization method, using data generated by mixing controlled levels of music with clean speech, and using the signal-to-noise ratio after suppression as the objective function. Although the music suppressor achieved a small improvement in the signal-to-noise ratio, the recognition deteriorated from an error rate of 45.1% to 46.2% on a subset of Dev96H4-F3.

5. CONCLUSIONS

Transcription of radio broadcasts poses several challenges. Many of these are problems whose solution will significantly advance the state-of-the-art in speech recognition. Recognition systems have to be developed that can cope with a variety of signal environments, speaking styles and accents, and multiple background noise sources. We have made an initial attempt at developing a system for transcription of broadcast news shows. The results obtained in the initial test are encouraging. Clearly much more work needs to be done in order to obtain an acceptable level of accuracy.

REFERENCES

[1] Proceedings of ARPA Speech and Natural Language Workshop, 1995, Morgan Kaufman Publishers.

[2] L. R. Bahl et al., "Robust Methods for using Context-Dependent features and models in a continuous speech recognizer", Proc. ICASSP, 1994.

[3] P. S. Gopalakrishnan, L. R. Bahl, R. Mercer, "A tree search strategy for large vocabulary continuous speech recognition", Proceedings of the ICASSP, pp , 1995.

[4] L. R. Bahl et al., "Performance of the IBM large vocabulary continuous speech recognition system on the ARPA Wall Street Journal task", Proceedings of the ICASSP, pp 41-44, 1995.

[5] P. S. Gopalakrishnan, R. Gopinath, S. Maes, M. Padmanabhan, L. Polymenakos, H. Printz, M. Franz, "Transcription of Radio Broadcast News with the IBM Large Vocabulary Speech Recognition System," Proc. of ARPA SLT Workshop, Feb 1996.

[6] L.R. Bahl, S. Balakrishnan-Aiyer, M. Franz, P.S. Gopalakrishnan, R. Gopinath, M. Novak, M. Padmanabhan, S. Roukos, "The IBM Large Vocabulary Continuous Speech Recognition System for the ARPA NAB News Task", Proc. of ARPA SLT Workshop, Jan 1995.

[7] P. S. Gopalakrishnan, R. Gopinath, S. Maes, M. Padmanabhan, L. Polymenakos, "Acoustic Models Used in the IBM System for the ARPA Hub 4 Task," Proc. of ARPA SLT Workshop, Feb 1996.

[8] L. Polymenakos, M. Padmanabhan, D. Nahamoo, P.S. Gopalakrishnan, "Suppressing background music from music corrupted data of the ARPA Hub 4 task," Proc. of ARPA SLT Workshop, Feb 1996.

[9] C. J. Legetter and P. C. Woodland, "Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous density HMM's", Computer Speech and Language, vol. 9, no. 2, pp 171-186.

[10] J. L. Gauvain and C. H. Lee, "Maximum-a-Posteriori estimation for multivariate Gaussian observations of Markov chains", IEEE Trans. Speech and Audio Processing, vol. 2, no. 2, pp 291-298, Apr 1994.

[11] S. Wegmann, D. McAllister, J. Orloff and B. Peskin, "Speaker Normalization on Conversational Speech", Proc. of ICASSP 96, pp. 339-343, 1996.

[12] G. Zavagliakos, J. McDonough, "Speaker Adapted Training", presentation at LVCSR Workshop, Baltimore, 1996.

[13] M. Padmanabhan, L. R. Bahl, D. Nahamoo, M. A. Picheny, "Speaker Clustering and Transformation for Speaker Adaptation in Large Vocabulary Speech Recognition Systems", ICASSP-96, vol. II, pp 701-704.