RECENT IMPROVEMENTS TO IBM'S SPEECH RECOGNITION SYSTEM FOR AUTOMATIC TRANSCRIPTION OF BROADCAST NEWS

S. S. Chen, E. M. Eide, M. J. F. Gales, R. A. Gopinath, D. Kanevsky, P. Olsen

IBM Thomas J. Watson Research Center
Yorktown Heights, NY 10598

ABSTRACT

We describe recent extensions and improvements to IBM's system for automatic transcription of broadcast news. The speech recognizer uses a total of 160 hours of acoustic traininng data, 80 hours more than for the system described in [6]. In addition to improvements obtained in 1997 we made a number of changes and algorithmic enhancements. Among these were changing the acoustic vocabulary, reducing the number of phonemes, insertion of short pauses, mixture models consisting of non-Gaussian components, pronunciation networks, factor analysis (FACILT) and Bayesian Information Criteria (BIC) applied to choosing the number of components in a Gaussian mixture model. The models were combined in a single system using NIST's script voting machine known as rover [8].

1. INTRODUCTION

Recently interest in large vocabulary continuous speech recognition recognition (LVCSR) research has shifted from read speech data to speech data found in the real world - like broadcast news (BN) over radio and TV and conversational speech over the telephone. A considerable amount of both acoustic (approximately 200 hours of which about 80% is usable) and linguistic (approximately 400 million words) training data for BN has been made by the Linguistic Data Consortium (LDC) in the context of DARPA sponsored Hub4 evaluations of large vocabulary continuous speech recognition (LVCSR) systems on BN [11]. BN transcription poses several challenges to LVCSR systems. The speech data exhibits a wide variety of speaking styles, environmental and background noise conditions and channel conditions. The general approach has been to classify the BN data into a set of homogeneous conditions and to build acoustic models for each condition. Test data is then segmented and classified along conditions and an appropriate acoustic model used for each condition. One particular classification scheme for BN news data that has been used in the DARPA sponsored Hub4 BN evaluation in 1996 splits the speech data along the so-called F-conditions [11]: prepared speech (F0), spontaneous speech (F1), low fidelity speech, including telephone channel speech (F2), speech in the presence of background music (F3), speech in the presence of background noise (F4), speech from non-native speakers (F5) and FX - all other speech. For rapid development we chose to extricate a subset of the testset of [6]. The amount of data from each of the F0–FX conditions was made equal in our test set.

In this paper we present algorithmic improvements to the baseline model used in the Hub4 evaluation in 1997, cf. [6]. Some of the improvements are: mixture models consisting of non-gaussian components, pronunciation networks, factor analysis (FACILT) and Bayesian Information Criteria (BIC) applied to choosing the number of components in a Gaussian mixture model. The focus of the research effort has been to improve all conditions (F0–FX) by improving the algorithmic foundation of last years recognizer. All the above mentioned methods were of this nature. To gain something from all of these methods we used NIST's script voting program, rover, that produces a single output from a number of scripts by voting. The roverized output is a considerable improvement over the individual systems.

2. OVERVIEW OF THE LVCSR SYSTEM

The IBM LVCSR system uses acoustic models for sub-phonetic units with context-dependent tying (see [2, 3] for details). 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 or Gaussian–like 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 recognizer used in the 1997 evaluation had 3.5K HMM states (or leaves) and 170K Gaussians. The decision trees for the HMM states were built using the relatively clean data from the F0 and F1 conditions, whereas the Gaussian mixtures were trained on the complete set of training data. As the data received from the additional training data was not segmented along conditions we decided to use the full set of data to build decision trees containing a total of 3.5K HMM states. The Gaussian mixtures were built from the full training data and the best single system we arrived at contained 289K Gaussian. The technique for finding optimal feature spaces developed last year was used in all models used in our current system [6]. For reasons pertaining to computational cost we used a language model without 4–grams for development as well as smaller Gaussian mixture models.

3. ACOUSTIC MODELING

3.1. Pronunciation Dictionary

As our phonetic spellings, also known as baseforms have been added to and composed in many different ways, the current list of baseforms comes from a variety of sources and contains many inconsistencies. To remove these inconsistencies we inspected spellings of words with common prefixes and suffixes. In addition we allowed words like "Human" with baseform HH Y UW M AX N to delete the HH as is done in some dialects of American–English. In baseforms where Y UW was preceded by a dental (T, D, TH or D) (e.g. as in duty D Y UW T IY or D UW T IY) we allowed the Y to be deleted for a similar reason. Lastly we went through words ending in "ING" and compared the baseforms to the baseform of it's root. The list of baseforms produced in this fashion was dubbed "clean". The resulting vocabulary gave little improvements, but made new types of errors as seen in section 6. A comparison is shown in table 1.

	All	F0	F1	F2	F3	F4	F5	FX
II	25.2	11.4	22.5	30.8	27.6	28.2	21.0	40.6
I	25.1	11.2	23.2	30.6	27.7	26.5	21.4	40.8

Table 1: Comparison of clean acoustic vocabulary (I) with old acoustic vocabulary (II). All numbers are percentages representing the word error rate.

3.2. Bayesian Information Criterion

The Bayesian Information Criterion (BIC) is a well known model selection criterion from the statistics literature. BIC was successfully used for segmentation and clustering for unsupervised adaptation in the 1997 evaluation, cf. [7].

A difficult problem one encounters when making a Gaussian mixture model is how to choose the number of Gaussians in the model. Too few Gaussians does not give sufficient model complexity and too many leads to overtraining. Using the BIC selection criteria we can automatically choose the number of mixture components in a data driven fashion. The higher the complexity of the data, the more clusters will be needed. Let n be the number of mixture components, C_n the clustering corresponding to n mixtures, N_{C_n} the number of parameters used in the mixture and N the number of data points. We define the BIC function BIC(n) as follows

$$\mathrm{BIC}(n) = \log(\mathrm{Likelihood}(C_n)) - \frac{\lambda}{2} * N_{C_n} * log(N)\ . \quad (1)$$

For an individual leaf we choose n to be such that it maximizes BIC(n) for a previously chosen value of λ.

The parameter λ in equation (1) allows us to choose the overall number of Gaussians in our system whereas the cardinality of Gaussians within individual leaves is left to be decided by the BIC function.

Experiments involving BIC consistently shows improved recognition for equally large Gaussian mixture models. This can be seen in Table 2. Systems of varying sizes was built by varying the value of λ. The accuracy was shown to consistently improve as the number of Gaussians increased to 289K, cf Table 3.

	All	F0	F1	F2	F3	F4	F5	FX
II	26.0	11.9	23.5	31.7	28.4	28.5	22.3	42.3
I	25.2	11.6	23.1	30.5	27.7	26.2	20.5	41.8

Table 2: Comparison of two systems: (I) Gaussian mixture models with 90K Gaussians for with and (II) without the BIC selection criterion.

	All	F0	F1	F2	F3	F4	F5	FX
135	24.7	11.2	21.2	29.5	29.0	26.8	21.6	41.2
178	24.2	10.7	21.5	29.3	26.5	25.9	21.4	40.3
237	23.8	10.7	21.6	29.3	26.5	24.2	19.7	39.6
289	23.5	10.5	21.5	28.9	24.4	24.6	20.7	39.0

Table 3: Gaussian mixture models built using the BIC selection criteria for different values of λ. The numbers of Gaussians are shown in terms of thousands in the leftmost column.

3.3. Short Pause

Previously our silence phone consisted of a 3-state Hidden Markov Model. This we felt was insufficient for modeling short pauses. To address this problem a new deleteable short pause phone SX was introduced at the end of each word. SX is modelled by a single deletable one-state Hidden Markov Model. This phone was introduced into our system and models retrained with the new phone. The idea being that short silences would not be "eaten up" by other phones at the endings and beginnings of words. The short pause appears to improve the conditions F0, F1 and FX as can be seen in Table 4

	All	F0	F1	F2	F3	F4	F5	FX
II	26.0	12.8	23.5	31.2	28.4	26.5	22.7	43.0
I	26.0	12.3	23.2	33.1	28.3	27.2	21.6	41.1

Table 4: Comparison of two systems: (I) with and (II) without the short pause phone SX.

3.4. Homogeneous Alpha Mixtures

To model data at the leaf level traditionally one assumes the distribution to be of the form

$$f(x) = \sum_{i=1}^{n} \omega^i \exp\left\{ -\left(\sum_{j=1}^{d} \frac{(x_j - \mu_j^i)^2}{2(\sigma_i^j)^2} \right) \right\}\ , \quad (2)$$

where d is the dimension of the vector $x = (x_1, \ldots, x_d)$ and the parameters to be decided are the number of mixture

	All	F0	F1	F2	F3	F4	F5	FX
II	24.6	11.1	21.1	29.1	29.1	26.8	21.3	41.1
I	24.1	10.6	21.3	29.8	25.9	26.6	21.8	39.9

Table 5: Comparison of two systems: (II) Gaussian mixture models and (II) homogeneous alpha mixture models.

components, m, the means $\{\mu^i\}_{i=1}^m = \{(\mu_1^i, \ldots, \mu_d^i)\}_{i=1}^m$, the standard deviations $\{\sigma^i\}_{i=1}^m = \{(\sigma_1^i, \ldots, \sigma_d^i)\}_{i=1}^m$ and the mixture weights $\{\omega^i\}_{i=1}^m$. Many of this years improvements deals with changes in this model. BIC is used to decide the value of m, FACILT is used to capture covariance structures and Homogeneous Alpha Mixtures (HAM) to capture the peakiness or impulsiveness of the data. When viewing graphical representation of densities of 1–dimensional projections of the data one is struck by the sharpness and asymmetries of the peaks of the pdf's. These are features that are difficult to capture using Gaussian mixtures. We decided to model the peakiness or impulsiveness using multidimensional generalizations of the power exponential distribution (also known as the alpha stable distribution)

$$f(x) = \sum_{i=1}^{n} \omega^i \rho_\alpha \exp\left\{ -\left(\gamma_\alpha \sum_{j=1}^{d} \frac{(x_j - \mu_j^i)^2}{2(\sigma_i^j)^2} \right)^{\frac{\alpha}{2}} \right\} , \quad (3)$$

where

$$\rho_\alpha = \frac{\alpha}{2} \frac{, (\frac{d}{2})}{(d\pi)^{\frac{d}{2}}} \frac{, (\frac{d+2}{\alpha})^{\frac{d}{2}}}{, (\frac{d}{\alpha})^{\frac{d}{2}+1}} \quad \text{and} \quad \gamma_\alpha = \frac{, (\frac{d+2}{\alpha})}{d, (\frac{d}{\alpha})} .$$

We refer to the case above where all the components have the same value of α as HAM (homogeneous alpha mixtures). The case of variable α–values is expounded in [5]. The re–estimation formulas for an EM-type re-estimation that we chose to use were previously published in [4]. They are as follows

$$\omega^\ell = \frac{1}{N} A_\ell,$$

$$\mu_i^\ell = \frac{\sum_{k=1}^N \left(\sum_{j=1}^d \frac{(x_j^k - \hat{\mu}_j^\ell)^2}{\hat{\sigma}_j^\ell} \right)^{\frac{\alpha-2}{2}} A_{\ell k} x_i^k}{\sum_{k=1}^N \left(\sum_{j=1}^d \frac{(x_j^k - \hat{\mu}_j^\ell)^2}{\hat{\sigma}_j^\ell} \right)^{\frac{\alpha-2}{2}} A_{\ell k}}$$

and

$$\sigma_i^\ell = \frac{\alpha \gamma_\alpha^{\frac{\alpha-2}{2}} \sum_{k=1}^N \left(\sum_{j=1}^d \frac{(x_j^k - \hat{\mu}_j^\ell)^2}{\hat{\sigma}_j^\ell} \right)^{\frac{\alpha-2}{2}} A_{\ell k} (x_i^k - \hat{\mu}_i^\ell)^2}{A_\ell}$$

where

$$A_{\ell k} = \frac{\hat{\omega}^\ell \rho_\alpha \exp\left\{ -\left(\gamma_\alpha \sum_{j=1}^d \frac{(x_j^k - \hat{\mu}_j^\ell)^2}{2(\hat{\sigma}_j^\ell)^2} \right)^{\frac{\alpha}{2}} \right\}}{\sum_{i=1}^m \hat{\omega}^i \rho_\alpha \exp\left\{ -\left(\gamma_\alpha \sum_{j=1}^d \frac{(x_j^k - \hat{\mu}_j^i)^2}{2(\hat{\sigma}_j^i)^2} \right)^{\frac{\alpha}{2}} \right\}}$$

and

$$A_\ell = \sum_{k=1}^N A_{\ell k} ,$$

for $\ell = 1, \ldots, m$, $k = 1, \ldots, N$ ($\{x^k\}_{k=1}^N$ is the training data) and $j = 1, \ldots, d$. Hatted quantities represent the previous values of the means, standard deviations and priors. Means, standard deviations and priors with no hats represent the new values. The value $\alpha = 1$ corresponding to Laplacian densities used by Phillips [10] was found to work best and yielded improvements over the standard systems as is seen in Table 5.

3.5. Factor Analyzed Covariances

Let j be an index referring to a specific mixture component. To better model covariances without modeling the full covariance matrices Σ_j whose dimensions are 60×60 we constrain the covariances to be of the form $\Sigma_j = A(\Lambda_j \Lambda_j^T + \Psi_j)A^T$ where A is a shared matrix capturing an optimal feature space, Λ_j is a "factor loading matrix" whose columns are less abundant than those of Σ_j, typically numbering 2 or 3 columns, and Ψ_j is a diagonal specific matrix. Methods for parameter estimation of Gaussian mixtures with covariances of this form are described in [9] and the method is named factor analyzed covariances invariant to linear transformations or FACILT for short. Some initial experiments with 2 column factor loading matrices are shown in Table 6. The only condition that improved significantly was FX. Experiments with different number of factors and tying structures of the covariances are still ongoing.

	All	F0	F1	F2	F3	F4	F5	FX
II	22.6	9.6	20.3	27.2	25.9	23.9	19.7	38.0
I	22.7	9.9	20.3	27.3	26.1	24.8	19.8	37.1

Table 6: Comparison of two systems: (I) FACILT (II) a comparable diagonal Gaussian model with an equivalent number of prototypes.

4. THE PHONE SET

We deleted 10 phones that we felt were treated erroneously and/or inconsistently in our set of baseform. These phones were AXR, AH, BD, DD, GD, IH, KD, PD, TD and TS. BD, DD, GD, KD, PD and TD are phones that were intended to model "double stops", i.e. stops that were followed by new stops and TS and AXR to model "T S" and "AX R" that was felt were such short sounds that individual phones had to be introduced. AH and IH are sounds that are very close to already existing sounds that are not distinguished well in our baseform set. After replacing all these phones in the acoustic dictionary we trained new Gaussian models and compared with the existing phone set. The results were significantly worse, cf. Table 7, but as seen in section 6 it helped yield an improved system when mixed with other pre-existing systems using rover.

5. PRONUNCIATION NETWORKS

Words in our speech recognizer are mapped to strings of phones, which are converted into subphonetic units corresponding to HMM states, and further converted into context dependent HMM states known as leaves. A mapping

	All	F0	F1	F2	F3	F4	F5	FX
II	25.2	11.4	22.5	30.8	27.6	28.2	21.0	40.6
I	27.8	13.9	25.0	33.1	31.3	30.2	26.0	43.1

Table 7: Comparison of two systems: (I) New phone set, 90K Gaussians (II) 130K Gaussians, old phoneset.

of the word "CAR" may look like "K AA R" in terms of phones, "$\mathtt{K}_1\ \mathtt{K}_2\ \mathtt{K}_3\ \mathtt{AA}_1\ \mathtt{AA}_2\ \mathtt{AA}_3\ \mathtt{R}_1\ \mathtt{R}_2\ \mathtt{R}_3$" in the feneme space and as leaves like: (l_{1970}, l_{1983}, l_{1998}, l_{75}, l_{83}, l_{92}, l_{3021}, l_{3103}, l_{3151}). Real speech is not as clean as these ideal labels. It would be desirable to find situations where individual sounds are closely related and allow these to be confused with each other. The intention of pronunciation networks is to remediate the phone confusion problem. Each phone is replaced by a small network of 3–14 HMM states corresponding to individual leaves chosen among the collection of all leaves from all phones. To build the networks a "ballistic" decoding that decodes as if the leaves were words, is performed on the training data. The string of decoded leaves are then aligned to the "correct" labels prescribed by a training transcription so that each "correct" leaf is assigned a string of ballistic leaf labels. Pairs of leafs and ballistic leaf strings with high co–occurrence counts are selected to build a network. This technique is an extension of work done on Fenonic modeling at IBM during the late eighties and early nineties. The pronunciation network models appear to improve F1 (spontaneous speech) as would be expected, cf. Table 8.

	All	F0	F1	F2	F3	F4	F5	FX
II	22.6	9.1	20.8	28.0	25.1	24.4	19.6	37.1
I	22.4	8.9	20.1	27.8	25.0	24.4	19.5	37.4

Table 8: Comparison of two systems: (I) Pronunciation networks and (II) traditional tristate HMM models.

6. ROVER

J. Fiscus introduced a voting scheme for combining word scripts produced by different speech recognizers, [8]. This program was named rover. We gleefully applied this program too many variations of our systems, arriving at an improved system. The philosophical technique was to locate systems that differed in as many ways as possible while still performing reasonable recognition. The best performing mixture of speech recognizers consisted of 4 systems with error rates shown in Table 9. The systems were: (I) a 289K Gaussian system built using BIC and retrained with the EM algorithm. This system uses the short pause phone. (II) A 135K homogeneous alpha mixture system with short stop phone and pronunciation networks. (III) a 120K Gaussian system built off of "clean" baseforms. (IV) An 80K Gaussian mixture built from our reduced set of phones.

	All	F0	F1	F2	F3	F4	F5	FX
I	21.5	8.9	19.7	26.7	23.0	23.0	16.9	36.1
II	22.4	8.9	20.1	27.8	25.0	24.4	19.5	37.4
III	23.1	10.3	21.5	27.8	25.7	24.5	18.2	37.8
IV	27.8	13.9	25.0	33.1	31.3	30.2	26.0	43.1
all	20.2	8.4	18.8	25.9	22.7	22.9	16.2	30.5

Table 9: Fully roverized system showing the 4 individual systems.

7. REFERENCES

[1] T. Anastasakos, et al., "A Compact Model for Speaker-Adaptive Training", Proc. ICSLP-96, (1996).

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

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

[4] S. Basu and C.A. Micchelli, "Parametric density estimation for the classification of acoustic feature vectors in speech recognition," Nonlinear Modeling: Advanced Black-Box Techniques (Eds. J. A. K. Suykens and J. Vandewalle), pp. 87-118, Kluwer Academic Publishers, Boston (1998).

[5] S. Basu, C. A. Micchelli, P. A. Olsen, "Maximum Likelihood Estimates for Exponential Type Density Families," submitted to ICASSP, (1999).

[6] S. S. Chen et al., "IBM's LVCSR System for Transcription of Broadcast News Used in the 1997 Hub4 English Evaluation," Proc. of DARPA Speech Recognition Workshop, Feb 8–11, Lansdowne VA, (1998).

[7] S. Chen et al, "Clustering via the Bayesian Information Criterion with Applications in Speech Recognition", Proc. ICASSP, (1998).

[8] J. G. Fiscus, "A post–processing system to yield reduced word error rates: recognizer output voting error reduction (rover)," technical report National Institute of Standards and Technology, (1997).

[9] R. A. Gopinath, "Constrained Maximum Likelihood Modeling with Gaussian Distributions," Proc. of DARPA Speech Recognition Workshop, Feb 8–11, Lansdowne VA, (1998).

[10] R. Haeb–Umbach, et al., "Acoustic modeling in the Phillips Hub4 continuous speech recognition system," Proc. of DARPA Speech Recognition Workshop, Feb 8–11, Lansdowne VA, (1998).

[11] D. Pallet, "Overview of the 1997 DARPA Speech Recognition Workshop," Proc. of DARPA Speech Recognition Workshop, Feb 2-5, Chantilly VA, (1997).