SPEECH RECOGNITION PERFORMANCE ON A NEW VOICEMAIL TRANSCRIPTION TASK

M. Padmanabhan, B. Ramabhadran, S. Basu

IBM T.J. Watson Research Center
PO Box 218, Yorktown Heights, NY, 10598, USA
mukund@us.ibm.com

ABSTRACT

In this paper we describe a new testbed for developing speech recognition algorithms - a VoiceMail transcription task, analogous to other tasks such as the Switchboard, CallHome, and the Hub 4 tasks, which are currently used by speech recognition researchers. We describe the collection and use of a new VoiceMail database (that is available to the research community through the LDC), and also describe some algorithmic techniques that were devised based on this data, and the initial results of transcription performance on this task.

1. INTRODUCTION

In this paper we describe a new testbed for developing speech recognition algorithms - a VoiceMail transcription task, analogous to other tasks such as the Switchboard, CallHome [1] and the Hub 4 tasks [2] which are currently used by speech recognition researchers. Voicemail represents a very large volume of real-world speech data that is not well represented in any of the currently existing databases. Consequently there is a need for a Voicemail database in order to improve transcription performance on a voicemail transcription task, and also to establish a new test bed for speech recognition algorithms.

2. DATA COLLECTION

Next, we will briefly describe the characteristics of this data (for details of the data collection scheme see [3]), and experimental results that establish a baseline for speech recognition performance on this database **this data is available to the research community through the Linguistic Data Consortium (www.ldc.upenn.edu)**.

- The data represents extremely spontaneous speech.
- The data contains both long-distance and local calls.
- The average length of a voicemail message is 31 seconds, however, the peak of the histogram of voicemail durations occurs at 18 seconds.
- The average rate of the speech is approximately 190 words per minute.
- The topics covered in the collected data ranged from personal messages to messages with technical or business-related content.
- The database was not quite gender balanced, with the percentage of male speakers being 38 %.

3. SYSTEM OVERVIEW

We will first briefly describe the IBM large-vocabulary speech recognition system. Essential aspects of the system used in the experiments here have been described earlier [4]; however, we will summarize the main features here :

The acoustic features used are 13-dimensional cepstra and their first and second differences, and a feature vector is extracted every 10 msec from the 8KHz sampled voicemail data. Words are represented as sequences of phones. Each phone is further divided into 3 sub-phonetic units which correspond roughly to the beginning, middle, and end of each phone. The system uses context-dependent HMM acoustic models for these sub-phonetic units. For each sub-phonetic unit a decision tree is constructed from the training data [4]. Each leaf of the tree corresponds to a different set of contexts. The acoustic observations that characterize the training data at each leaf are modeled as a mixture of gaussian pdf's, with diagonal covariance matrices. The systems used in this paper had approximately 2700 leaves, and anywhere from 17000 to 170000 gaussians. The system also uses an envelope-search algorithm [4] to hypothesize a sequence of words corresponding to the utterance. A simple word N-gram (bigram or trigram) model is used to compute the language model probabilities.

4. ACOUSTIC MODELS

4.1. Clean-up of transcriptions

The initial transcriptions that we started off with for the 20 hours of voicemail data were not very clean, and had a fair number of transcription errors. As it would have been impractical to verify all these transcriptions manually, we devised an automatic scheme to identify possible transcription errors. This tagged around 1 % of the data, and we then corrected these transcriptions manually. Very briefly, the main idea used in the tagging scheme was to viterbi align the speech data against the (possibly incorrect) transcription, and then identify regions where the log-likelihood assigned to a phone by the alignment process was particularly low. For more details see [3]. This process identified script errors as well as baseform errors, which were then corrected manually.

4.2. Compound words

An additional observation arising from the tagged segments of the acoustic data was that crossword co-articulation was very common in this data because of the casual nature of the speech and the fast speaking rate. For instance, the phrase 'going to take' would often be pronounced as 'gontake = G OW N T EY KD'. For our initial experiments, we chose to model such effects by constructing compound words [8, 9]. We selected these compound words based on the tagged segments of the acoustic training data. Some examples of the compound words and their pronunciations is given in Table I

Table I

CAN − WE	*K AX W IY*
FOR − YOU	*F AX Y UW*
GIVE − ME	*G IH M IY*
GOOD − MORNING	*G UH M AA N IX N*
IT − WAS	*IX W AX Z*
SO − IF	*S OW F*
TO − YOU	*CH Y UW*
TRYING − TO	*T R AY N AX*
WANT − TO	*W AA N AX*
YOU − CAN	*Y UW N*

The use of these compound words serves a dual purpose. Firstly, they enable the modelling of crossword co-articulation effects. Secondly, it is generally the case that decoding errors are more common in shorter words, hence, as the compound words have relatively long baseforms, there are fewer errors in the compound words. We decided to extend the second piece of reasoning above and apply it to model commonly occurring phrases in the voicemail data. Hence, we constructed compound words of the form 'give-me-a-call', 'thank-you', 'thanks-a-lot', 'when-you-get-a-chance' etc.

4.3. Phonological rules

In order to model co-articulation effects in words other than compound words, we used some of the phonological rules described in [5]. Examples of such co-articulation effects are plosive deletion (deletion of word final TD in the word sequence 'excellent point'), palatization (did-you being pronounced as 'D IH JH UW'), etc. Such effects can be accounted for using linguistic rules [5], that specify the conditions under which the boundary phones in a word may be deleted or replaced by other phones.

Some of the rules that we implemented are listed below (P_{n-1} and P_n denote the last two phones of the first word, and N_1 denotes the first phone of the next word).

1. Geminate Deletion: If P_n = Consonant and N_1 = Same consonant then delete P_n Example: this-street DH IH S T R IY TD
2. Palatization: If P_n =D and N_1 = Y then replace P_n with JH and delete N_1 Example: did-you D IH JH UW and what-you W AH CH UW
3. Plosive Deletion: If P_{n-1} = N, P_n = plosive and N_1 = plosive the delete P_n Example: went-down W EH N D AW N

4.4. Model Complexity Adaptation

As mentioned earlier, we model leaves in our system with mixtures of gaussians. In general, ad-hoc rules are used to determine the number of mixture components that will be used to model a particular leaf - for example, the number of components is made proportional to the amount of data, subject to a maximum number. This choice of the number of components may not necessarily provide the best classification performance - consequently, we introduced a discriminant measure to choose the number of mixture components in a more optimal manner. The details of this algorithm are given elsewhere [10], so we will only summarize it briefly here.

The essence of the algorithm is to start with a small baseline system, and evaluate how well the gaussian mixture model for a leaf models the data for that leaf. This is done by computing the posterior probability of correct classification of the data for that leaf. If this probability is low, this implies that the model for the leaf does not match the data for the leaf very well; hence, the resolution of the model for the leaf is increased by adding more components to its model.

In our implementation, we start with two systems (say S1 and S2), where S2 models each leaf with more gaussians than S1. Subsequently, we find those leaves that are not adequately modelled by S1 according to our discriminant criterion, and replace the model for that leaf in S1 with the corresponding model from S2.

4.5. Tree growing experiments

As mentioned earlier, the voicemail database comprises of messages from a variety of acoustic environments. Consequently, constructing the decision trees (to identify phonetic context dependence of the HMM states) from this data could result in the tree trying to isolate the environment rather than acoustically dissimilar phonetic pronunciations. Further the amount of available acoustic data is only 20 hours. Consequently, we experimented with constructing the decision tree from (i) bandlimited WSJ data (60 hours)(ii) bandlimited Hub 4 data (from the F0 and F1 conditions) (40 hours) and (iii) from the Voicemail data (20 hours). Subsequently, the gaussians modelling the leaves of the tree were trained using the Voicemail acoustic data. Results indicated that the use of the bandlimited WSJ data for constructing the trees gave the best performance.

4.6. Feature extraction experiments

Our initial experiments used 13-dimensional Mel cepstra and their first and second derivatives, but we also experimented with using alternative features such as PLP cepstra [6] and linear discriminant features. We are also currentky experimenting with the use of smoothed estimates for the Mel cepstra [7], the rationale being that the smoothing would lead to a reduction in the variance of the estimated feature vectors, thus leading to "tighter" models.

5. EXPERIMENTAL RESULTS

Our first set of experiments were conducted when we had only 10 hours of training data available, and several of these experiments were repeated on 20 hours of training data. We will present experimental results for both

these training sets (we will refer to them as Vmail10 and Vmail20), as the difference in performance gives an indication of the effect of increasing the amount of training data on different components of the recognizer (acoustic model, language models, etc.).

5.1. Test data

The test data was 43 voicemail messages (picked at random from the collected data, and not included in the training set). The size of the Vmail10 vocabulary was 6K words, and the out-of-vocabulary (o.o.v.) rate of the test data with respect to this vocabulary was 4.6 %. The size of the Vmail20 vocabulary was 10K words, and the oov rate of the test data with respect to this vocabulary was 3.5 %. The perplexity of the test set was around 120.

5.2. Results

We conducted a number of incremental experiments to observe the effect of adding additional training data to different components of the recognizer. The word error rates are given in Table VI (any reference to row numbers in the remainder of this section should be interpreted as row of Table VI).

(1) The initial system we started with was trained on the Vmail10 database and used a bigram LM. This gave an error rate of 49.75 %. Re-estimated the parameters of this acoustic model using the Vmail20 datbase (LM was still a bigram estimated from the Vmail10 data) dropped the error rate to 46.22 % (row 1).
(2) Subsequently, we re-estimated the bigram LM using the Vmail20 database, and decoded the test data using the same acoustic model as in row 1. This dropped the error rate to 45.12 % (row 2).
(3) Subsequently, we estimated a trigram LM using the Vmail20 database, and used this with the same acoustic model of row 1. This dropped the error rate to 42.7 % (row 3).
(4) Next we used a weighted mixture of the Vmail trigram LM of row 3, and a trigram built off the Switchboard data (in the proportion 0.3 Swb LM probability + 0.7 Vmail20 LM probability). The error rate corresponding to this condition was 42.95 % (row 4).
(5) Next, we estimated a MCA model putting together a system (S1) with 83.5K gaussians, and a system (S2) with 175K gaussians. The resulting MCA model had 78K gaussians. Using the mixture trigram LM of row 4, and the MCA model dropped the error rate to 41.94 % (further details are given in the next section).
(6) Next, we used VTL [11] to normalize the spectra of the training speakers on a per-message basis and re-estimated the acoustic model of (5) with the normalized training data. Further, the normalization was also applied to the test speakers on a per-message basis. This dropped the error rate to 40.52 % (row 6).
(7) Next, we used the MLLR speaker adaptation technique [12] to do unsupervised adaptation of the acoustic models of (6) on a per-message basis. This further dropped the error rate to 38.92 % (row 7).
(8) Finally, we applied the phonological rules of Section. 4.3 in the decoding process, and used them with the models of (7). This brought the error rate down to 38.18 % (row 8).

Table II (word error rate)

(1) Bigram LM - Vmail10	46.22
(2) Bigram LM - Vmail20	45.12
(3) Trigram LM - Vmail20	42.70
(4) Trigram LM - Vmail20 + Swb	42.95
(5) MCA model	41.94
(6) Adaptation (VTL)	40.52
(7) Adaptation (VTL+MLLR)	38.92
(8) Phonological rules	38.18

Model Complexity Adaptation We now present some experimental results on model complexity adapation (MCA) (see Section. 4.4) that indicate that the new method of determining the complexity of the model yields consistent gains over standard methods. We constructed five models using the standard ad-hoc method of allocating a fixed number of gaussians for the each leaf. These models respectively had a maximum of 7, 12, 35, 60, and 150 gaussians per mixture (gpm). Subsequently, we used MCA to construct models that replace the gaussian mixtures for some leaves in the 7 gpm model with gaussian mixtures from the 35 gpm model. This model will be referred to as 7x35 in the following table (Table III). Table III tabulates the error rates and the size of several models, constructed by conventional means, and using MCA.

Table III

	# gaussians	Word error rate
Conventional models		
7 gpm	16.5K	47.53
12 gpm	25.8K	45.07
35 gpm	69.5K	43.25
60 gpm	89.5K	43.86
150 gpm	128.5K	44.16
MCA models		
7x35	25K	45.72
12x35	33K	43.2
35x150	78.2K	42.2
60x150	96.5K	42.9

The error rate as a function of the number of gaussians in the model is shown plotted in Fig. 1, and it can be seen that the MCA models consistently outperform the conventional models by around 5% (relative). Also, note that due to the limited amount of training data, the error rate starts increasing as the number of parameters increases beyond a certain point. Additional details of these experiments are given in [13].

Results on tree-growing and feature-space related experiments For these experiments, the phonology used was a little different from that used in the experiments described in Tables II and III - We also added three additional phones to explicitly model fillers, UH, UM, and ER. In the first set of experiments, we used 13-dimensional Mel cepstra and their first two derivatives as the feature vectors, and constructed the decision trees using different data sets (as tabulated in Table IV). Subsequently, once the trees had been grown, the parameters of the gaussians modelling the leaves of the tree were estimated from the Voicemail acoustic training

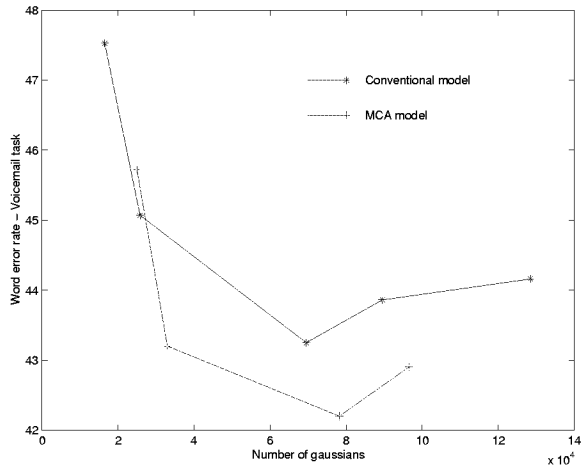

Figure 1:

data. The results are summarized in Table IV.

Table IV

Data type	Amount of data	Word error rate
Bandlted WSJ	60 hours	41.49
Bandlted Hub4	40 hours	42.75
Voicemail	20 hours	44.66

For the feature space related experiments, we used the decision tree grown from bandlimited WSJ data and simply re-estimated the gaussian mixtures at its leaves in different feature spaces. The feature spaces we experimented with are (i) 13-dimentional Mel cepstra and their first 2 derivatives (ii) 13-dimensional PLP cepstra and their first 2 derivatives (iii) we computed the leading 39 linear discriminants of the 39-dimensional feature space in (i), with the LDA being designed to separate out the leaves of the decision tree as classes (iv) finally, we experimented with using a smoothed estimate of the Mel cepstra [7]. In (i) the Mel cepstra are computed every 10 ms (using a 32ms window of speech); in the smoothed cepstra, we compute the Mel cepstra every 2 ms (using a 25ms window of speech), and average five adjacent cepstral vectors to extract one every 10 ms. The word error rate obtained with these different feature spaces is tabulated in Table V.

Table V

Feature space	Word error rate (%)
Mel Cepstra	41.49
PLP Cepstra	41.44
Mel+LDA	43.76
Smoothed Mel	40.68

6. ACKNOWLEDGEMENT

We would like to acknowledge the support of DARPA under Grant MDA972-97-C-0012 for funding this work.

7. REFERENCES

1. Proceedings of LVCSR Workshop, Oct 1996, Maritime Institure of Technology.
2. Proceedings of ARPA Speech and Natural Language Workshop, 1995, Morgan Kaufman Publishers.
3. M. Padmanabhan, et al., "Issues involved in voicemail data collection", Proceedings ARPA Hub4 Workshop, Lansdowne VA, Feb 1998.
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. E. P. Giachin, et al., "Word juncture modeling using phonological rules for HMM-based continuous speech recognition", Computer, Speech and Language, pp 155-168, Academic Press, 1991.
6. H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech", Journal of the Acoustic Society of America, pp 1738-1752, April 1990.
7. S. Dharanipragada et al., "Techniques for capturing temporal variations in speech signals with fixed-rate processing", elsewhere in these proceedings.
8. M. Finke and A. Waibel, "Speaking mode dependent pronunciation modeling in large vocabulary conversational speech recognition", Proceedings of EUROSPEECH 1997, vol. 5, pp 2379-2382.
9. P. Jeanrenaud, et al., "Reducing word error rate on conversational speech from the Switchboard corpus", Proceedings of ICASSP, 1995, pp 53-56.
10. L. R. Bahl, M. Padmanabhan, "A discriminant measure for model complexity adaptation", submitted to ICASSP 98.
11. S. Wegman, D. McAllaster, J. Orloff, and B. Peskin, "Speaker normalization on conversational telephone speech".
12. 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.
13. M. Padmanabhan et al., "Transcription of new speaking styles - Voicemail", Proceedings ARPA Hub4 Workshop, Lansdowne VA, Feb 1998. *Also available at http://www.nist.gov/speech.*