S11.3

ACOUSTIC MARKOV MODELS USED IN THE TANGORA SPEECH RECOGNITION SYSTEM

L. R. Bahl, P. F. Brown, P. V. de Souza, R. L. Mercer and M. A. Picheny

Speech Recognition Group, IBM Thomas J. Watson Research Center,
P.O. Box 218, Yorktown Heights, NY 10598, USA

ABSTRACT

The Speech Recognition Group at IBM Research has developed a real-time, isolated-word speech recognizer called Tangora, which accepts natural English sentences drawn from a vocabulary of 20,000 words. Despite its large vocabulary, the Tangora recognizer requires only about 20 minutes of speech from each new user for training purposes. The accuracy of the system and its ease of training are largely attributable to the use of hidden Markov models in its acoustic match component. This paper describes an automatic technique for constructing Markov word models, and includes results of experiments with speaker-dependent and speaker-independent models on several isolated-word recognition tasks.

1. INTRODUCTION

The speech recognition group at IBM Research has recently described an experimental large-vocabulary natural-language isolated-word speech recognition system [1, 2]. The PC-based recognizer, named for Albert Tangora who is listed in the Guinness Book of Records as the fastest typist, is capable of handling a 20,000-word vocabulary in real time. At the heart of the Tangora is an acoustic match component in which hidden Markov models are used to represent the pronunciation of words.

In this paper we present a new approach to constructing acoustic Markov models. We describe a method for deriving an acoustic representation of a word automatically from sample utterances of the word. This method results in a substantial decrease in recognition error rate when compared with methods based on phonetic representations of words.

First, let us consider word-based Markov models. An example of a recognition system which uses word-based Markov models is the system implemented by Bakis [7, 13]. Here, each word is represented by a Markov model which is derived from sample utterances of the word. The number of states in the model for a word is equal to the average duration of the word in frames. The frame size in Bakis' system is 10 milliseconds. Figure 1 shows an example of a Bakis-type model.

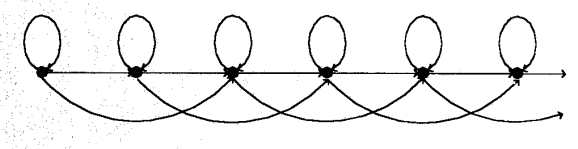

Figure 1. Word-based Markov models used by Bakis.

The Markov model in Figure 1 is most easily interpreted as a speech synthesis machine. Starting at the left-most state a transition is made during each time frame until the final state is reached. In the Bakis model, there are three possible transitions out of most states: one leading back to the same state, one leading to the next state on the right, and one leading to the state after next. Associated with each transition is an output probability distribution describing the various sounds which can be output during the transition, and a transition probability indicating the likelihood of taking the transition.

In a speech recognizer it is the job of the acoustic match component to determine the probability of some observed speech being generated by such a model. If the probability of generating the observed speech is relatively high, then the word represented by the Markov model is considered a likely candidate for the true word. An efficient lattice-based algorithm for computing the required probability is described by Jelinek [13].

In the Bakis model, the direct path through the model represents the average pronunciation of the word. The self-loops allow for elongation of the word, and the transitions which skip around states allow for shortening of the word. The transition probabilities and output distributions for each word are estimated from several sample utterances of the word, using the Forward-Backward parameter estimation algorithm [8, 9, 13]. This system worked very successfully on a 250-word continuous-speech speaker-dependent recognition task. However, there is one major drawback to such a recognizer: the user must provide several sample utterances of each word for parameter estimation, so it is not practical for large-vocabulary systems. The number of Markov parameters and the training data requirements grow linearly with the vocabulary size.

More recently, Rabiner and Levinson [16] described a system in which the number of states in each word model was reduced to about five. This results in a substantial reduction in the number of parameters, without much degradation in the accuracy of the model. This is because neighboring states in the Bakis model tend to be quite similar, and reducing several similar consecutive states into a single state does not degrade the model very much. However, training each word model requires several sample utterances, so the training data size still grows linearly with vocabulary size.

Another way to reduce the number of parameters is to build word models from a small inventory of sub-word models, such as phones, diphones, syllables, etc. Most attempts to define sub-word units are based on linguistic or phonetic concepts. An example of a system in which phones are used as the building blocks for constructing word models is described by Bahl, Jelinek and Mercer [3, 4, 14]. Each word is represented by a sequence of phones, called the phonetic baseform of the word. A Markov model is established for each phone, and the Markov model for a

CH2561-9/88/0000-0497 $1.00 © 1988 IEEE

word is obtained by replacing each phone in its baseform by the Markov model for the phone. This decomposition of words into phone sequences removes the dependence of the training data size on the vocabulary. In order to train the system adequately, one only needs several samples of each phone, and not several samples of each word. Phonetics-based models, however, have the drawback that the phonetic baseform is based on "expert" human knowledge, which unfortunately is subjective, difficult to extract, and subject to error.

When sufficient data are available to train them, the Bakis word-based models outperform the phonetics-based models [5]. The performance of phonetics-based models can be improved by using context-dependent phone models [6, 18], but it is doubtful if they can be made to perform as well as word-based models. The main reason for the superiority of the word-based models is that they model the words at a much finer level of detail. The effects of context-dependence and coarticulation are implicitly built into the models.

Our aim is to construct Markov models that retain the accuracy of the word-based models while requiring no more training data than the phonetics-based models.

We can consider phones to partition the acoustic space into regions. The partition is based on *a priori* human linguistic knowledge. A phonetic baseform of a word is then a specification of the sequence of acoustic regions traversed in pronouncing that word. Instead of using phones, we will show in the following sections how to use a partition of the acoustic space which is derived automatically through the use of a vector quantizer. Based on this partition, we will construct a new type of baseform called a fenonic baseform. These baseforms can be used to construct Markov word models in the same way that phonetic baseforms are used for constructing word models.

2. SPEAKER-DEPENDENT SINGLETON BASEFORMS

We will assume that readers are familiar with the basic concepts of vector quantization as used in speech recognition systems. Articles by Gray [11], and Makhoul *et al* [15] contain excellent surveys of vector quantization techniques. A vector quantizer has as its input an acoustic waveform, and produces as its output a sequence of discrete labels. The input acoustic waveform is digitized, and a vector of acoustic parameters is extracted from the signal at regular intervals. This vector is then compared to a set of reference vectors, and a label is produced which identifies the closest reference vector. In our system, we extract a vector of 20 ear-model parameters [1, 10] from the speech signal at regular intervals of 10 milliseconds, and compare this vector to 200 reference vectors using a Euclidean distance measure. So labels are produced at a fixed rate of 100 per second, and the label alphabet is of size 200.

Let $\boldsymbol{y}_{1\to m} = y_1, y_2, \ldots y_m$ be the label sequence produced by the acoustic processor in response to a speaker uttering the word w. We can treat the label sequence $\boldsymbol{y}_{1\to m}$ as if it were a baseform for the word w. Of course, other utterances of the same word will not result in label sequences that are identical to $\boldsymbol{y}_{1\to m}$, but they will generally produce sequences that are similar to $\boldsymbol{y}_{1\to m}$. We can model this variation by replacing each label in the baseform by a Markov model. What we are doing here is replacing the phonetic-phone alphabet used in making phonetic baseforms by an alphabet of label-based phones. We shall call these label-based phones fenones, and we shall call a baseform expressed in terms of fenones a fenonic baseform. When the baseform is constructed from a single utterance, as here, we shall call it a singleton-fenonic baseform, or singleton baseform for short.

Let $\boldsymbol{P} = \{p_1, p_2, \ldots p_V\}$ represent the alphabet of fenones. There is an obvious one-to-one correspondence between the elements of the fenone alphabet and the underlying label alphabet. And from the method of construction it can be seen that each fenone in a baseform will represent a single time frame on average.

Since each fenone represents a very short acoustic interval, the variation can be adequately modeled by very simple Markov models like those shown in Figure 2.

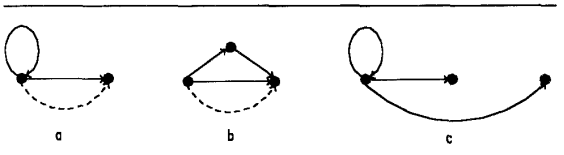

Figure 2. Examples of Markov models for fenones.

A 2 or 3 state model with 3 or 4 transitions suffices to account for the variation in a fenone, while a much larger model is needed for adequately modeling the variation in a phone. The transitions drawn as dotted lines result in no output.

The Markov model for a word is obtained by concatenating the Markov models of the fenones in its fenonic baseform. In Fig. 2c, the lowest transition for fenone n connects to the initial state of fenone $n + 2$, so concatenating these models results in a Bakis-type word model. The model in Fig. 2a is the simplest, and allows for deletion and unlimited elongation of a fenone. The model in Fig. 2b limits a fenone from producing more than 2 labels, while the model in Fig. 2c allows shortening by no more than a factor of two. Obviously, many other models are possible. All the experimental results in this paper were obtained using a fenone model of the type in Fig. 2a. Experiments were also carried out with models of the type shown in Fig. 2b and 2c, and there were no significant differences in recognition accuracy.

Figure 3 shows the topology of a fenonic baseform using the fenone model of Fig. 2a.

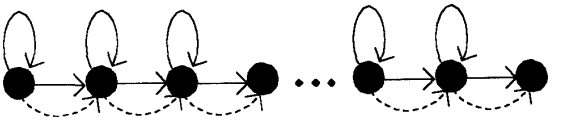

Figure 3. Topology of a typical fenonic Markov word model.

By taking one instance of each word in the vocabulary we can produce singleton baseforms for each word in the vocabulary. The Markov model for any given word may then be constructed by concatenating the elementary Markov models of the fenones in its baseform. Note that the total number of Markov model parameters depends only on the size of the fenone alphabet, and not on the size of the vocabulary.

Fenonic baseforms model the acoustic trajectory of a word at a much finer level of detail than phonetic baseforms. In this respect, fenonic baseforms are similar to word templates used in DP matching [12, 17]. The lattice [13] generated from the fenonic model of Fig. 3 is the same as the lattice used in conventional DP matching. But, whereas the transition and output probabilities associated with the Markov models of fenones are trained to the speaker, no such training process exists in DP matching. Also, since all the words are composed of the same fenones, there is an implicit "tying" amongst subparts of different words. It is this tying that makes it possible to train all the word models even if the training data does not contain samples of all the words in the vocabulary.

Since a singleton baseform is derived from a single sample of a word, it may not be typical and may not result in good recognition accuracy.

3. SPEAKER-DEPENDENT FENONIC BASEFORMS

Since singleton baseforms may not perform very well, we investigated the possibility of using fenonic baseforms derived from multiple utterances of words.

We start with singleton baseforms, and train the acoustic Markov models using some training data. This establishes transition and output probabilities for the fenones in $\boldsymbol{P}$. Let $\boldsymbol{y}_{1 \to m_1}^{(1)}, \boldsymbol{y}_{1 \to m_2}^{(2)}, \ldots, \boldsymbol{y}_{1 \to m_n}^{(n)}$ be the label sequences corresponding to n different utterances of the word w. The optimal fenonic baseform for w is defined to be the fenone sequence $\hat{\boldsymbol{p}}_{1 \to M}$ which maximizes the probability of the n observed label sequences for w, i.e.

$$\hat{\boldsymbol{p}} = \underset{\boldsymbol{p}}{\operatorname{argmax}} \ \Pr\{\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \ldots, \boldsymbol{y}^{(n)} | \boldsymbol{p}\}. \tag{1}$$

Note that, for the sake of simplicity we have dropped the subscripts on $\boldsymbol{y}$ and $\boldsymbol{p}$.

Assuming that, given $\boldsymbol{p}$, $\boldsymbol{y}^{(i)}$ is conditionally independent of $\boldsymbol{y}^{(j)}$ for $i \neq j$, then

$$\hat{\boldsymbol{p}} = \underset{\boldsymbol{p}}{\operatorname{argmax}} \ \prod_{i=1}^{n} \Pr\{\boldsymbol{y}^{(i)} | \boldsymbol{p}\}. \tag{2}$$

From the equations above we can see that the problem of finding the optimal fenonic baseform is essentially the same as the problem of maximum likelihood word sequence decoding as formulated in the papers by Bahl, Jelinek and Mercer [4, 14], with the following two differences: 1) we search for the most likely fenone sequence rather than most likely word sequence, and 2) we have multiple label sequences rather than a single sequence. With minor modifications, the decoding algorithm of [4, 14], which carries out a tree search using a stack can be used to search for the optimal fenonic baseform. The first modification is to replace the vocabulary of words by the vocabulary of fenones. The second modification is that the acoustic match is carried out for multiple label strings; the total acoustic-match probability being the product of the individual acoustic-match probabilities for the label strings. Obviously, many other search algorithms can also be adapted to look for the optimal fenonic baseform.

Recognition experiments on Keyboard and Office Correspondence tasks were then carried out using fenonic baseforms obtained from multiple utterances. All experiments were carried out with a single speaker, and all recordings consisted of isolated words.

The Keyboard task consisted of recognizing keyboard characters from a vocabulary of 62 words. The vocabulary included the letters of the alphabet A-Z, the numbers 0-9, punctuation characters like .,:; and other characters such as @#%&$. The test data consisted of 10 utterances of each word spoken in random order. Ten utterances of each word were used to construct fenonic baseforms, and a word error rate of 0.8% was obtained. The error rate with phonetic baseforms was 4.7%.

In the Office Correspondence task, the vocabulary consisted of the 2000 most frequent words in a database of IBM office correspondence. Two sets of test data were used. The first consisted of one utterance of each word in the vocabulary, spoken in random order, and the second consisted of 100 natural sentences obtained from memos, comprising a total of 1297 words. In the case of the natural sentences, recognition was carried out with a 3-gram language model [4]. Homophonous words such as "to", "too", and "two" were constrained to have identical baseforms, so that when a uniform language model was used, the scores of homophonous words were identical, thereby eliminating homophone errors. Four utterances of each word were used for baseform construction. On random words, the error rates were 2.2% for fenonic baseforms and 5.2% for phonetic baseforms. And on natural sentences with a 3-gram language model, the word error rates were 0.7% for fenonic baseforms and 2.5% for phonetic baseforms.

4. SPEAKER-INDEPENDENT FENONIC BASEFORMS

The baseforms used in Section 3 do not alleviate the problem of reducing the amount of training data, since the speaker must provide one or more utterances of each word. Our goal was the construction of systems with 5,000 or 20,000 words, and most users would be unwilling to provide even one sample of each word for vocabularies of this size.

In order to overcome this difficulty we investigated the possibility of using speaker-independent fenonic baseforms. Experiments were carried out on a 5000-word Office Correspondence task, which is similar to the former except that the vocabulary was increased to include the 5000 most frequent words. Details of this task can be found in [1].

We recorded 10 utterances of each word in the vocabulary, one each from 10 different speakers. The speakers were obtained from an agency that supplies temporary office help. They were all male and were natives of the New York-New Jersey area. Using a few minutes of speech from each of the speakers, a speaker-independent label alphabet was constructed by vector quantization. The entire speech database was then labeled with this speaker-independent alphabet. Next, fenonic baseforms were constructed for each word in the vocabulary from the 10 recorded utterances.

A user of the 5000-word recognition system was required to read a training script of 100 natural sentences, comprising a total of about 1200 words. A speaker-dependent label alphabet was established for each user by constructing a vector quantization dictionary from five minutes of speech. Note that the speaker-dependent label alphabet will not be the same as the fenone alphabet. This is no different from the phonetic case, where the label and phone alphabets are not the same. The fenone Markov models were trained using the above-mentioned training data. Recognition was performed on a test script of 50 sentences, comprising a total of 591 words spoken as isolated utterances. These sentences were selected at random from a collection of IBM internal office correspondence. Sentences

containing words outside the vocabulary were rejected. Recognition was performed for 8 speakers, 4 male and 4 female, none of whom contributed utterances from which the fenonic baseforms were obtained. In all cases a 3-gram language model was used. The use of fenonic models reduced the average word error rate to 2.5% from 3.5% for phonetic models – a reduction of 28% in the number of errors.

5. CONCLUSIONS

We have described a method for constructing a new type of acoustic baseform for a word. This baseform is a sequence of fenones. Fenones are very short acoustic events and the inventory of fenones is obtained automatically by vector quantization of the acoustic space. Fenones are completely independent of any pre-conceived linguistic or phonetic notions. Fenonic baseforms model words at a much greater level of detail than phonetic baseforms, and result in a substantial reduction in the word error rate. Training of the fenonic Markov models for a new speaker can be accomplished with a short training script.

Fenonic baseforms are somewhat similar to word templates used in DP matching, but have a crucial difference. The word models constructed from fenonic baseforms can be trained to a new speaker, whereas DP word templates cannot be trained.

ACKNOWLEDGEMENTS

We would like to thank the other members of the Speech Recognition group at the IBM Research Center for their help and insight.

REFERENCES

[1] A. Averbuch, L. Bahl, R. Bakis, P. Brown, A. Cole, G. Daggett, S. Das, K. Davies, S. De Gennaro, P. de Souza, E. Epstein, D. Fraleigh, F. Jelinek, S. Katz, B. Lewis, R. Mercer, A. Nadas, D. Nahamoo, M. Picheny, G. Shichman and P. Spinelli, "An IBM-PC based large-vocabulary isolated-utterance speech recognizer", *Proceedings of the 1986 IEEE International Conference on Acoustics, Speech, and Signal Processing, Tokyo, Japan,* pages 53-56, April 1986.

[2] A. Averbuch, L. Bahl, R. Bakis, P. Brown, G. Daggett, S. Das, K. Davies, S. De Gennaro, P. de Souza, E. Epstein, D. Fraleigh, F. Jelinek, B. Lewis, R. Mercer, J. Moorhead, A. Nadas, D. Nahamoo, M. Picheny, G. Shichman, P. Spinelli, D. Van Compernolle and H. Wilkens, "Experiments with the TANGORA 20,000 word speech recognizer", *Proceedings of the 1987 IEEE International Conference on Acoustics, Speech, and Signal Processing, Dallas, Texas,* pages 701-704, April 1987.

[3] L.R. Bahl and F. Jelinek, "Decoding for channels with insertions, deletions, and substitutions with applications to speech recognition", *IEEE Transactions on Information Theory,* IT-21(4):404-411, July 1975.

[4] L.R. Bahl, F. Jelinek, and R.L. Mercer, "A maximum likelihood approach to continuous speech recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* PAMI-5(2):179-190, March 1983.

[5] L. R. Bahl, R. Bakis, P. S. Cohen, A. G. Cole, F. Jelinek, B. L. Lewis, and R. L. Mercer, "Recognition results with several experimental acoustic processors", *Proceedings of the 1979 IEEE International Conference on Acoustics, Speech, and Signal Processing, Washington, DC,* pages 249-251, April 1979.

[6] L. R. Bahl, R. Bakis, P. S. Cohen, A. G. Cole, F. Jelinek, B. L. Lewis, and R. L. Mercer, "Further results on the recognition of a continuously read natural corpus", *Proceedings of the 1980 IEEE International Conference on Acoustics, Speech, and Signal Processing, Denver, Colorado,* pages 872-875, April 1980.

[7] R. Bakis, "Continuous speech recognition via centisecond acoustic states", *91st Meeting Acoustical Society of America, Washington, DC,* April 1976.

[8] L.E. Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes", *Inequalities,* 3:1-8, 1972.

[9] L.E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains", *The Annals of Mathematical Statistics,* 36(6):1554-1563, December 1966.

[10] J. Cohen, "Application of an adaptive auditory model to speech recognition", *110th Meeting Acoustical Society of America, Nashville, Tennessee,* November 1985.

[11] R. M. Gray, "Vector quantization", *IEEE ASSP Magazine,* 1(2):4-29, April 1984.

[12] F. Itakura, "Minimum prediction residual principle applied to speech recognition", *IEEE Transactions on Acoustics, Speech, and Signal Processing,* ASSP-23(1):67-72, February 1975.

[13] F. Jelinek, "Continuous speech recognition by statistical methods", *Proceedings of the IEEE,* 64(4):532-556, April 1976.

[14] F. Jelinek, L.R. Bahl, and R.L. Mercer, "Design of a linguistic decoder for the recognition of continuous speech", *IEEE Transactions on Information Theory,* IT-21(3):250-256, May 1975.

[15] J. Makhoul, S. Roucos, and H. Gish. "Vector quantization in speech coding". *Proceedings of the IEEE,* 73(11):1551-1588, November 1985.

[16] L. R. Rabiner and S. E. Levinson, "A speaker-independent, syntax-directed, connected word recognition system based on hidden Markov models and level-building", *IEEE Transactions on Acoustics, Speech, and Signal Processing,* ASSP-33(3):561-573, June 1985.

[17] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition", *IEEE Transactions on Acoustics, Speech, and Signal Processing,* ASSP-26(1):194-200, February 1978.

[18] R. Schwartz, Y. Chow, S. Roucos, M. Krasner, and J. Makhoul, "Improved hidden Markov modeling of phonemes for continuous speech", *Proceedings of the 1984 IEEE International Conference on Acoustics, Speech, and Signal Processing,* San Diego, California, pages 35.6.1-35.6.4, March 1984.