A SEGMENTAL SPEECH MODEL WITH APPLICATIONS TO WORD SPOTTING

Herbert Gish *Kenney Ng*

BBN Systems and Technologies
70 Fawcett Street 15/1c, Cambridge MA 02138 USA

ABSTRACT

In this paper we present a segmental speech model that explicitly models the dynamics in a variable duration speech segment by using a time varying trajectory model of the speech features in the segment. In this approach, each speech segment is represented by a set of statistics which includes a time varying trajectory, a residual error covariance around the trajectory, and the number of frames in the segment. These statistics replace the frames in the segment and become the data that is modeled by either HMMs or mixture models. This segment model is used to develop a secondary processing algorithm that rescores putative events hypothesized by a primary HMM word spotter to try to improve performance by discriminating true keywords from false alarms. This algorithm is evaluated on a keyword spotting task using the Road Rally Database and performance is shown to improve significantly over that of the primary word spotter. In addition, the segmental model is also used on a TIMIT vowel classification task to evaluate its modeling capability.

1. INTRODUCTION

It is well known, e.g. [6,7], that the dynamics and time correlation of speech frames contain useful information that can be exploited in the modeling of speech. In fact, cepstral derivatives are examples of features that exploit this dependence in a simple way. In addition, a variety of other methods have been employed to further exploit this dependence [1,5].

In terms of modeling variable duration segments of speech, empirical observations of the behavior of features in a speech segment show that in general the features are not stationary but vary with time across the duration of the segment. This suggests that a model that can take into account the dynamics of the features in a segment may provide useful information.

In the new segmental modeling approach described in this paper, we explicitly model the dynamics in a variable duration speech segment by using a time varying trajectory model of the speech features in the segment. A set of statistics based on this model is then derived to represent the segment. These statistics include a time varying trajectory, a residual error covariance around the trajectory, and the number of frames in the segment. These statistics replace the frames in the segment and become the actual data that is modeled by either HMMs or mixture models. In this paper, we examine segment models with constant, linear, and quadratic trajectories in the context of mixture models.

In the next section, the segmental model is described in detail. Section 3 then describes two applications of the segment model. One is a speaker and context independent vowel classification task using the TIMIT database. The other is the secondary processing of putative events generated by a primary HMM word spotter using the Road Rally Database. Finally, experimental results for these two applications are presented in Section 4.

2. MODELING SPEECH SEGMENTS

In our segmental modeling approach, we model each feature dimension of a speech segment as

$$c(n) = \mu(n) + e(n) \qquad \text{for } n = 1, \ldots, N \tag{1}$$

where $c(n)$ are the observed cepstral features in a segment of length N, $\mu(n)$ is the mean feature vector as a function of frame number and represents the dynamics of the features in the segment, and $e(n)$ is the residual error term which we assume to have a $N(0, \Sigma)$ distribution. In addition, the errors are assumed to be independent from frame to frame. The mean feature vector models that we consider in this paper will be at most a quadratic function of time, i.e.,

$$\begin{aligned} \mu(n) &= b_1 + b_2 n + b_3 n^2 \qquad \text{for } n = 1, \ldots, N \\ &= \underline{\mathbf{z}}' \cdot \underline{\mathbf{b}} \end{aligned} \tag{2}$$

where $\underline{\mathbf{z}}' = [1 \; n \; n^2]$ and $\underline{\mathbf{b}}' = [b_1 \; b_2 \; b_3]$.

In the following section, we formulate the segment model in matrix notation, provide solutions for the model parameters, and present an iterative EM formulation to allow the training of segment mixture models.

2.1. Segment Model

Given a speech segment with a duration of N frames, where each frame is represented by a D dimensional feature vector, the segment can be expressed in matrix notation as:

$$\mathbf{C} = \begin{bmatrix} c_{1,1} & \cdots & c_{1,D} \\ c_{2,1} & \cdots & c_{2,D} \\ \vdots & & \vdots \\ c_{N,1} & \cdots & c_{N,D} \end{bmatrix} = \begin{bmatrix} \underline{\mathbf{C}}_1 & \cdots & \underline{\mathbf{C}}_D \end{bmatrix} \tag{3}$$

and modeled, after Equation 1, as:

$$\mathbf{C} = \mathbf{ZB} + \mathbf{E} \tag{4}$$

where $\mathbf{Z}$ is a $N \times R$ design matrix that specifies the type of model to use, $\mathbf{B}$ is a $R \times D$ trajectory parameter matrix, and $\mathbf{E}$ is a residual error matrix. R is the number of parameters in the trajectory model: $R = 1$ for constant, $R = 2$ for linear, and $R = 3$ for quadratic trajectories.

This matrix equation can be rewritten, so that each feature dimension i is modeled explicitly, as:

$$\underline{\mathbf{C}}_i = \mathbf{Z}\underline{\mathbf{B}}_i + \underline{\mathbf{E}}_i \qquad \text{for } i = 1, \ldots, D \tag{5}$$

where $\underline{\mathbf{B}}_i$ is a $R \times 1$ trajectory vector for feature i, $\underline{\mathbf{E}}_i$ is a residual error vector for feature i, and $\mathbf{Z}\underline{\mathbf{B}}_i$ is the trajectory component for feature i analogous to $\mu(n)$, $n = 1, \ldots, N$ in Equation 2.

0-7803-0946-4/93 $3.00 © 1993 IEEE

Expanding out Equation 5 for a quadratic trajectory model and a segment with N frames, we get:

$$\begin{bmatrix} c_{1,i} \\ c_{2,i} \\ \vdots \\ c_{N,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{N-1} & (\frac{1}{N-1})^2 \\ \vdots & \vdots & \vdots \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_{1,i} \\ b_{2,i} \\ b_{3,i} \end{bmatrix} + \begin{bmatrix} e_{1,i} \\ e_{2,i} \\ \vdots \\ e_{N,i} \end{bmatrix}$$
$$\text{for } \; i = 1, \ldots, D \qquad (6)$$

or

$$c_{n,i} = b_{1,i} + b_{2,i}\left(\tfrac{n-1}{N-1}\right) + b_{3,i}\left(\tfrac{n-1}{N-1}\right)^2 + e_{n,i}$$
$$\text{for } \; n = 1, \ldots, N \; \text{ and } \; i = 1, \ldots, D. \qquad (7)$$

From this equation, we see that the number of trajectory parameters for a segment is dependent on the order of the model and is given by RD. The residual error covariance matrix Σ, however, has a fixed dimension of D independent of the model order. Since these segment parameters will be modeled and combined with those of other segments, the duration of each segment needs to be normalized to a nominal value. This normalization is handled in the design matrix $\mathbf{Z}$. As illustrated in Equation 6, each segment is normalized so that its frames are distributed uniformly between times 0 and 1 inclusively. The design matrix also allows great flexibility in modifying the segment model. For example, the choice of using a constant, linear, or quadratic trajectory requires only a straightforward change in the matrix entries.

2.2. Estimating Model Parameters: One Segment

Given the segment model in Equation 4, the next step is to solve for the model parameters. Assuming that the errors are independent and identically distributed (normal with covariance Σ), the Maximum Likelihood (ML) estimate of the trajectory parameter matrix, $\hat{\mathbf{B}}_k$, is given by the linear least squares estimate:

$$\hat{\mathbf{B}}_k = \left[\mathbf{Z}_k'\mathbf{Z}_k\right]^{-1}\mathbf{Z}_k'\mathbf{C}_k \qquad (8)$$

for a segment k with data matrix, $\mathbf{C}_k$, and design matrix, $\mathbf{Z}_k$.

With $\hat{\mathbf{B}}_k$ estimated, the residual error covariance matrix for the segment, $\hat{\Sigma}_k$, is given by:

$$\hat{\Sigma}_k = \frac{\hat{\mathbf{E}}_k'\hat{\mathbf{E}}_k}{N_k} = \frac{\left(\mathbf{C}_k - \mathbf{Z}_k\hat{\mathbf{B}}_k\right)'\left(\mathbf{C}_k - \mathbf{Z}_k\hat{\mathbf{B}}_k\right)}{N_k} \qquad (9)$$

where N_k is the number of frames in segment k.

As an example, let us examine a speech segment consisting of the vowel **/ay/** extracted from the TIMIT database. Shown in Figure 1 are the actual trajectories of cepstral coefficients C_1 and C_2 (solid lines) for this 33 frame long segment and the fit of three different trajectory models to the features. It is clear that the cepstral features are not stationary during the entire segment, but are time varying. Examining the different trajectory models, we observe that the linear (dot-dashed) and quadratic (dashed) models fit the data better than the constant model (dotted).

After parameter estimation, each segment is replaced by its set of statistics: $\{\hat{\mathbf{B}}_k, \hat{\Sigma}_k, N_k\}$. This collection of parameters is sufficient to answer questions about the likelihood of the frames comprising the original segment in the context of the Gaussian model. This property means that these parameters are sufficient statistics. This is important since in using the segments in an HMM or mixture model, we will be asking questions about the segment likelihood. In addition, since the reestimation of model parameters in a mixture model (or HMM) via the Estimate-Maximize (EM) algorithm is a linear function of all the segment statistics weighted by the segment likelihoods conditioned on the mixture component (or HMM state), using the segment sufficient statistics in the reestimation is mathematically equivalent to having used the original frames in the segment.

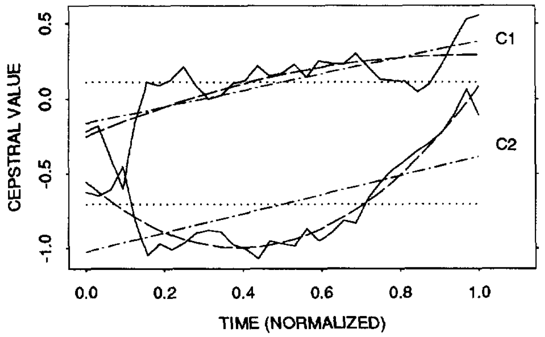

Figure 1. Temporal Variation of Cepstral Features C_1 and C_2 for an **/ay/** Vowel Segment, and the Fit of Constant, Linear, and Quadratic Trajectory Models to the Features.

2.3. Estimating Model Parameters: EM Formulation

Once we can estimate parameters for individual segments, the next step is to come up with an iterative EM formulation to allow the training of Gaussian mixture segment models from a set of K speech segments.

Given a segment k represented by parameters $\hat{\mathbf{B}}_k$, $\hat{\Sigma}_k$, and N_k, and a segment model m with parameters Σ_m and $\mathbf{B}_m$, the likelihood of segment k given model m can be expressed as:

$$L(\hat{\mathbf{B}}_k, \hat{\Sigma}_k | \mathbf{B}_m, \Sigma_m) \; = \; \mathrm{l}(k|m) \; = \qquad (10)$$
$$(2\pi)^{-\frac{DN_k}{2}} \, |\Sigma_m|^{-\frac{N_k}{2}} \cdot \exp\left(-\frac{N_k}{2}\mathrm{tr}\left[\Sigma_m^{-1}\hat{\Sigma}_k\right]\right) \cdot$$
$$\exp\left(-\frac{1}{2}\mathrm{tr}\left[\mathbf{Z}_k(\hat{\mathbf{B}}_k - \mathbf{B}_m)\Sigma_m^{-1}(\hat{\mathbf{B}}_k - \mathbf{B}_m)'\mathbf{Z}_k'\right]\right).$$

Once $\mathrm{l}(k|m)$ is computed for all K segments and all M components in the mixture model, the probability of the model given the segments can then be computed as:

$$\mathrm{p}(m|k) = \frac{\mathrm{l}(k|m)\mathrm{p}(m)}{\sum_{j=1}^{K}\mathrm{l}(j|m)\mathrm{p}(m)} \qquad (11)$$

and used to obtain new ML estimates for the model parameters $\mathrm{p}(m)$, $\mathbf{B}_m$, and Σ_m using the reestimation equations:

1. Prior probability for model m:

$$\mathrm{p}(m) = \frac{1}{M}\sum_{k=1}^{K}\mathrm{p}(m|k) \qquad (12)$$

2. Trajectory parameter for model m:

$$\mathbf{B}_m = \left[\sum_{k=1}^{K}\mathrm{p}(m|k)\mathbf{Z}_k'\mathbf{Z}_k\right]^{-1}\left[\sum_{k=1}^{K}\mathrm{p}(m|k)\mathbf{Z}_k'\mathbf{Z}_k\hat{\mathbf{B}}_k\right] \qquad (13)$$

3. Covariance matrix for model m:

$$\Sigma_m = \frac{\sum_{k=1}^{K}\mathrm{p}(m|k)(\mathbf{C}_k - \mathbf{Z}_k\mathbf{B}_m)'(\mathbf{C}_k - \mathbf{Z}_k\mathbf{B}_m)}{\sum_{k=1}^{K}\mathrm{p}(m|k)N_k} \qquad (14)$$

These values are then used to estimate new values of $\mathrm{l}(k|m)$ for the next step in the iteration. This iterative reestimation procedure is repeated until convergence is reached in order to train the mixture model parameters.

3. APPLICATIONS OF THE SEGMENT MODEL

We have used the segment model in two different applications. The first is the classification of vowels in American English. The main goal in this task is to evaluate the modeling capability of the segment model. The second, and main, application is in the development of a secondary processing algorithm for our primary HMM word spotter [4]. The motivation here is to try to improve on the performance of the word spotter by taking advantage of the segment model's ability to model the cepstral dynamics. The two applications are described in this section. Experimental results are presented in Section 4.

3.1. Vowel Classification

To evaluate the segment model in a direct way, we performed experiments on a speaker independent vowel classification task. The corpus for this task consists of 16 vowels: 13 monothongs /**iy, ih, ey, eh, ae, aa, ah, ao, ow, uw, uh, ux, er**/ and 3 diphthongs /**ay, oy, aw**/. The vowels are excised, using the given phonetic segmentations, from the acoustically phonetically compact portion of the TIMIT corpus without any restrictions on the phonetic contexts of the vowels. From the 420 available speakers, 370 are used for training and the remaining 50 are used for testing. The test speakers are the same as those used in [3]. There is a total of 15,116 training tokens and 1,871 test tokens.

After the tokens are extracted, segment statistics are computed for each token and one Gaussian model is trained for each of the 16 vowels using the equations in Section 2. Since the segment boundaries are known, the maximum *a posteriori* probability rule is used for classification of an unknown test segment k:

$$\max_{m} \left[\mathrm{l}(k|m, N)\mathrm{p}(N|m)\mathrm{p}(m) \right] \qquad (15)$$

where $\mathrm{p}(N|m)$ is the probability that phoneme m has length N, and is computed as a histogram of the training segment durations. In order to match the dynamic ranges of $\mathrm{l}(k|m, N)$ and $\mathrm{p}(N|m)$, an exponential weighting factor is placed on the duration term and selected to optimize performance on the training set.

3.2. Secondary Processing

The main application of the segment model has been in the development of a secondary processing algorithm that rescores putative events hypothesized by a primary HMM word spotter to try to improve performance by discriminating true keywords from false alarms. Since the boundaries of the segments in the putative events are not known in this task, we currently generate a deterministic segmentation based on the spectral properties of the speech. The segmentation algorithm is data driven and segments speech at points of discontinuity in the observed spectrum. Details of the segmentation algorithm are described in [2].

After segmentation, sufficient statistics are generated for the segments which are then modeled by a Gaussian mixture model. Currently, two mixture models are created for each keyword using labeled putative events generated by the primary word spotter. One mixture is for segments from true keywords, and the other is for segments from false alarms. The mixture model parameters are trained using a two step procedure: the first is initialization using the results of an unsupervised clustering of the training segments, and the second is iteration over the segments using the EM algorithm. The clustering is used to determine the number of different types of segments in the keyword and, as a result, the number of components to use in the mixture model. In addition to modeling segment characteristics, first-order segment transition probabilities and word duration probabilities are also estimated.

Once these models for a particular keyword are trained, they are used to rescore new putative events of that keyword. The first step is the computation of a "secondary" score for each putative event, which is computed as the log likelihood ratio between the probability that it came from the truth model, and the probability that it came from the false alarm model. Both of these probability scores are computed using a dynamic programming (DP) search that finds the most probable (highest scoring) path through the segments of the putative event given the truth and false alarm models. The DP search takes into account the likelihood of each putative event segment belonging to each class in the mixture model, the segment class transition probabilities, and the keyword duration probabilities. To obtain the final scores, which are used to reorder the set of putative events, the new secondary scores and the original HMM scores from the primary word spotter are first normalized and then summed. Details of the secondary processing algorithm can be found in [2].

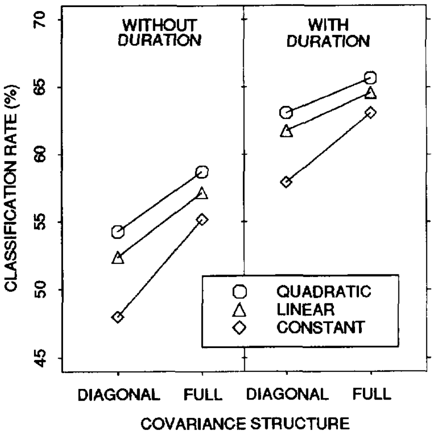

Figure 2. Performance on the TIMIT Vowel Classification Task.

4. EXPERIMENTS

Performance results of the vowel classification and secondary processing experiments are presented in this section. The input features used in both sets of experiments are normalized mel-warped LPC cepstra ($nc_1 - nc_{10}$), and cepstral derivatives ($dc_0 - dc_{10}$). A simple cepstral subtraction normalization is performed to compensate for channel and speaker variability. The TIMIT classification experiments use 8 kHz wide-band speech data analyzed at a 5 ms frame rate. The word spotting experiments use 300-3300 Hz band-limited speech data analyzed at a 10 ms frame rate.

4.1. Vowel Classification

Performance of the segment model on the 16 vowel classification task is shown in Figure 2 for various model complexities. Experiments are performed with and without using the duration information $\mathrm{p}(N|m)$. We see that using the duration information significantly improves performance. We also see that using a full residual error covariance is much better than using a diagonal covariance. And finally, we see that increasing the complexity of the trajectory from constant (◇) to linear (△) and finally to quadratic (○) results in a steady, although moderate, improvement in performance. Using quadratic trajectory models with a full residual error covariance matrix and duration information, a classification rate of 65.6% correct is achieved.

We observe that only some vowels, mainly the diphthongs, show significant improvement in performance when more complex trajectory models are used. Many of the other vowels show only small gains. This implies that more complex models are needed only for some segments, while simpler models may be sufficient for others. Another observation is that the context independent nature of the task degrades the accuracy of the segment models by introducing large variability near the boundaries

Data Set Label	Duration	# KWs	# Speakers	Speaker IDs	Purpose
Waterloo males	55 min.	2796	28	wm29–wm56	HMM Training
Stonehenge Training males	25 min.	454	10	sm49c–sm57c, sm59c	Secondary Training
Stonehenge Testing males	26 min.	412	10	sm33c–sm41c, sm43c	Testing Set

Table 1. Description of the Road Rally Data Sets used in the Secondary Processing Experiments.

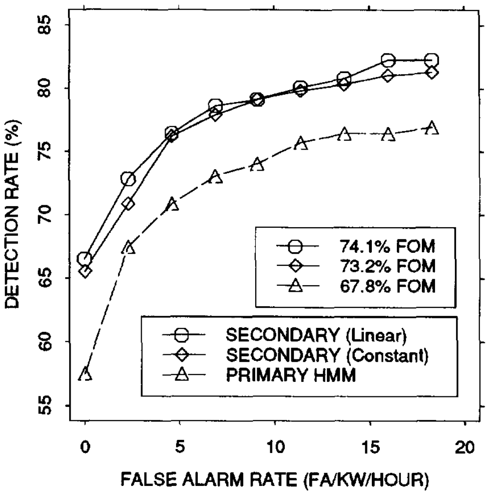

Figure 3. ROC Performance Curves for Secondary Processing.

of the segment. The least squares parameter estimation used in the segment model is sensitive to this variability especially for the linear and quadratic models. A possible solution is to use a mixture model for each vowel where each component models segments that have similar contexts in order to minimize the variability along the segment boundaries.

4.2. Secondary Processing

The secondary processing algorithm is evaluated on a twenty keyword spotting task using the Road Rally Database. The three data sets used in this experiment are described in Table 1. The marked keyword occurrences from the read paragraph speech of the 28 male speakers in the Waterloo portion of the database are used to train the primary HMM word spotter. Twenty male conversational speech files from the Stonehenge portion of the database are then processed with the primary word spotter in order to generate putative events for training and testing the secondary processor. Putative events from 10 Stonehenge speakers are used to train the secondary processor. The remaining 10 speakers make up the test set for the secondary processor. Putative events for the eleven keywords listed in Table 2 are rescored with the secondary processor. The other nine keywords are not rescored because the primary word spotter did not generate enough false alarms to train the secondary processor models for those words. Due to the small amount of training data, only diagonal residual error covariance matrices are used in these experiments.

chester	conway	interstate	look
middleton	minus	mountain	road
thicket	track	want	

Table 2. List of the Eleven Rescored Keywords.

Figure 3 shows the ROC (Receiver Operating Characteristic) performance curves of the primary HMM word spotter with and without secondary processing for the 10-speaker Stonehenge test set. Composite detection rate in percent for all twenty keywords (P_d) is plotted against false alarm rate in fa/kw/hr. We see that performance after rescoring the putative events with the secondary processor using either the constant trajectory model (◇) or the linear trajectory model (○) is significantly better than the performance obtained with just the primary HMM scores alone (△). At 0 fa/kw/hr, P_d improves from 57.5% to 65.5% with the constant model, and to 66.5% with the linear model. In terms of Figure of Merit (FOM), which is the average P_d from 0 to 10 fa/kw/hr, performance improves from 67.8% to 73.2% FOM with the constant model, and to 74.1% FOM with the linear model.

Performance improves only slightly in going from the constant to the linear trajectory model. A possible reason for this is that currently, all segments are modeled with the same type of model, i.e., either constant or linear trajectory. We believe that significant additional performance gains can be obtained by assigning different models to different segments based on segment duration and spectral variability within a segment. This approach should enable more efficient use of the small amount of training data by not over parameterizing the segments.

5. SUMMARY

In this paper we present a segmental speech model that explicitly models the dynamics in a variable duration speech segment by using a time varying trajectory model of the speech features in the segment. This segment model is evaluated on a TIMIT vowel classification task and gives performance comparable to those reported in other studies [3]. In addition, the segment model is used to develop a secondary processing algorithm that significantly improves word spotting performance by rescoring putative events to try to discriminate true keywords from false alarms.

ACKNOWLEDGMENT

This work was sponsored by the U. S. Department of Defense.

REFERENCES

1. V. Digalakis, J.R. Rohlicek, and M. Ostendorf, "A Dynamical System Approach to Continuous Speech Recognition," in *Proc. ICASSP 1991*, pp. 289-292.
2. H. Gish, K. Ng, and J.R. Rohlicek, "Secondary Processing using Speech Segments for an HMM Word Spotting System" in *Proc. ICSLP 1992*, pp. 17-20.
3. H. M. Meng, V. W. Zue, and H. C. Leung, "Signal Representation, Attribute Extraction, and the Use of Distinctive Features for Phonetic Classification," in *Proc. DARPA Workshop on Speech and Natural Language*, Pacific Grove, CA, February, 1991, pp. 176-181.
4. J.R. Rohlicek, W. Russell, S. Roucos, and H. Gish, "Continuous Hidden Markov Modeling for Speaker-Independent Word Spotting," in *Proc. ICASSP 1989*, pp. 627-630.
5. S. Roucos, M. Ostendorf, H. Gish, and A. Derr, "Stochastic Segment Modeling Using the Estimate-Maximize Algorithm," in *Proc. ICASSP 1988*, pp. 127-130.
6. F. K. Soong, and A. E. Rosenberg, "On the Use of Instantaneous and Transitional Spectral Information in Speaker Recognition," in *Proc. ICASSP 1986*, pp. 877-880.
7. C. J. Wellekens, "Explicit Time Correlation in Hidden Markov Models for Speech Recognition," in *Proc. ICASSP 1987*, pp. 384-387.