USING EXPLICIT SEGMENTATION TO IMPROVE HMM PHONE RECOGNITION

Carl D. Mitchell, Mary P. Harper, and Leah H. Jamieson

School of Electrical Engineering, Purdue University
West Lafayette, IN 47907-1285
{cdm,harper,lhj}@ecn.purdue.edu

ABSTRACT

We show that many of the errors in a context-dependent phone recognition system are due to poor segmentation. We then suggest a method to incorporate explicit segmentation information directly into the HMM paradigm. The utility of explicit segmentation information is illustrated with experiments involving five types of segmentation information and three methods of smoothing.

1. INTRODUCTION

One of the most attractive features of HMMs for speech recognition is that segmentation and classification are solved simultaneously. However, the maximum likelihood training criterion may not lead to a model that best utilizes the acoustic information for segmentation.

In this study, we investigate the possibility of improving HMM performance by providing explicit segmentation information. We define a *change function* as a function that directly measures the spectral variation of the acoustic signal. The change function is integrated into the HMM as the cost of making a transition from one phone to another phone during Viterbi alignment.

We consider a variety of change functions. The hand-labeled phone boundaries that are provided with the TIMIT corpus provide the ideal change function. We use the TIMIT phone transcriptions to provide an upper bound on the benefits of explicit segmentation. We also consider two change functions that can be automatically extracted from the acoustic signal: one based on delta cepstral coefficients, and the other based on the spectral variation function (SVF) used in [1]. Since smoothing the output parameters has an effect on segmentation, we consider three different methods of smoothing in conjunction with explicit segmentation information.

2. RELATED WORK

Other approaches [1, 2] have used spectral variation to explicitly segment the signal and before using HMM to determine the most likely sequence of phones or words. For the context-independent isolated word recognition task addressed in [1], a fixed number of frames are selected from each segment to represent the acoustic information contained in the signal. The context-independent phone recognition system described in [2] represents a phone by three spectral vectors, taken at the beginning, middle, and end of a segment. Our approach differs in the following ways:

1. Our HMM operates at the frame level, and all frames are processed by the HMM. Our goal is to improve segmentation rather than overcome the modeling errors due to the HMM assumption that multiple observations emitted from the same state are assumed to be independent. Hence, our system does not use a variable frame rate.
2. We tightly integrate the segmentation information in the HMM.
3. We can include segmentation information incrementally by weighting the change function. With a weight of zero, the model reduces to a standard HMM. This allows us to evaluate the utility of segmentation information by comparing the HMM results with and without segmentation information.
4. We use segmentation information in a context-*dependent* recognition system.

3. SYSTEM DESCRIPTION

Our HMM models 47 phones in context, which are mapped to a set of 39 phone classes for evaluation [3]. For these experiments, we haved used a discrete HMM with the following four codebooks: 1) 12 cepstral coefficients warped with a bilinear transform, 2) 12 delta cepstral coefficients, 3) power, and 4) delta power [4, 5]. A frame is 20 ms long with 10 ms overlap. We use 3360 sentences from 420 speakers for training, 1184 sentences from 148 speakers for smoothing output parameters, and 160 sentences from 20 speakers for testing. There is no overlap between the three speaker sets. All sentences are taken from the TIMIT corpus. The context-dependent output distributions are smoothed by mixing the following: a uniform distribution, a context-independent distribution, and the distributions associated with the other 46 contexts. For example, the beginning distribution of an 's' that follows 'k' is mixed with the beginning distribution of an 's' that follows 'ae', the beginning

This work is supported in part by the Purdue Research Foundation and NSF grant number IRI-9011179.

0-7803-2431-5/95 $4.00 © 1995 IEEE

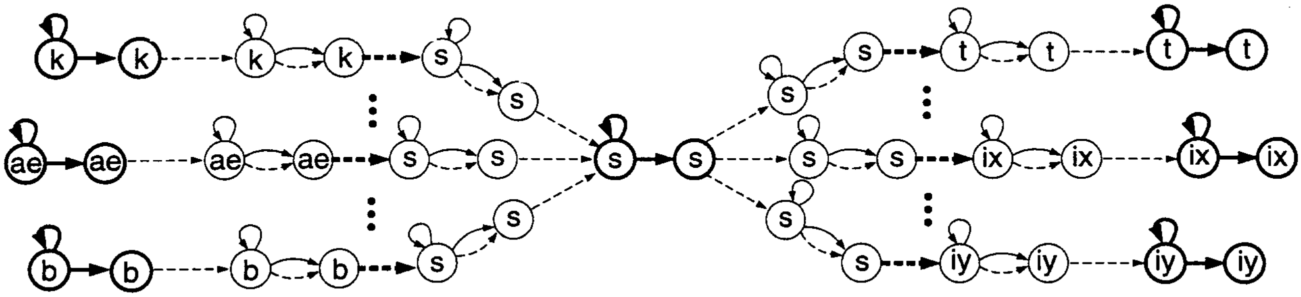

Figure 1: Topology of the phone recognizer. The boldface circles represent the middle of a phone, which is assumed to be context-independent. Dashed lines are null transitions, and boldface dashed lines represent phone boundaries.

distribution of an 's' that follows 'b', etc. We have found this method of mixing to be superior to mixing with only a context-independent distribution and a uniform distribution. Six iterations of the forward-backward algorithm are used to train the model and five iteration of smoothing are performed using held-out interpolation [6].

4. UTILIZING THE CHANGE FUNCTION AS A TRANSITION PENALTY

The topology of our phone recognizer is shown in figure 1. Each phone is comprised of three distributions. The beginning distribution is left context-dependent, the ending distribution is right context-dependent, and the middle distribution is context-independent. The dashed lines in figure 1 represent null transitions. The observations are emitted during the non-null transitions between states, represented by solid lines. The null transitions in figure 1 that are shown in boldface dashed lines serve as phone boundaries. A transition between phones incurs a cost that depends on how much the signal is changing at the time of the transition. Unlike variable frame rate processing, a spurious peak in the change function is not necessarily problematic for the HMM recognizer since a phone transition is not mandatory at points of large spectral change, and the HMM topology need not be modified to handle segmentation errors.

Let $c(t)$ denote the change function. Since the change function is used during Viterbi alignment, we will describe its values in the log domain. A large positive value for $c(t)$ encourages the HMM to make a phone transition at time t, hopefully to correct a deletion. A large negative value discourages a transition to a new phone, hopefully to eliminate an insertion. It is important to note that only designated transitions between phones (the boldface dashed lines in figure 1) utilize the change function; other state transitions are not affected.

5. CHANGE FUNCTIONS

5.1. No Explicit Segmentation

If the change function $c(t)$ is set to zero for all t, then there is no transition penalty or incentive, and the model reduces to a standard HMM. This provides a baseline performance.

5.2. Hand-Labeled Segmentation

We include segmentation information during recognition because we hypothesize that many classification errors are due to poor segmentation. We have tested this hypothesis by providing hand-labeled segmentation during both training and recognition. Ignoring human errors in the TIMIT transcriptions, the hand-labeled segmentations provide the ideal change function:

$$c(t) = \begin{cases} +\infty & \text{if a phone ends at } t \\ -\infty & \text{otherwise} \end{cases} \tag{1}$$

The vertical transitions of $c(t)$ occur at the given phone boundaries, which are shown in figure 2a.

In order to analyze classification errors that are due to segmentation, we also consider a change function that allows TIMIT phone boundaries to be skipped, but disallows spurious segment boundaries. This change function, which we will refer to as "no-insert", also uses the TIMIT transcriptions:

$$c(t) = \begin{cases} 0 & \text{if a phone ends at } t \\ -\infty & \text{otherwise} \end{cases} \tag{2}$$

A transition still must occur at a hand-labeled phone boundary, but the transition is not mandatory. This essentially eliminates insertions, but allows deletions. A small number of insertions still result, however, because the scoring algorithm finds the best alignment between the reference and test phones.

5.3. Automatic Segmentation

Using hand-labeled segmentation as the change function provides an upper bound to the advantages of explicit segmentation. For explicit segmentation to be useful, however, it must be automatically extracted from the signal. We have considered two alternatives for an automatically generated change function: the sum magnitudes of normalized delta cepstral coefficients and a variation of the spectral variation function (SVF) used in [1]. These change functions are derived directly from the observation vectors, so little signal processing overhead is required.

The delta cepstral change function estimates spectral change by summing the normalized time derivative of each

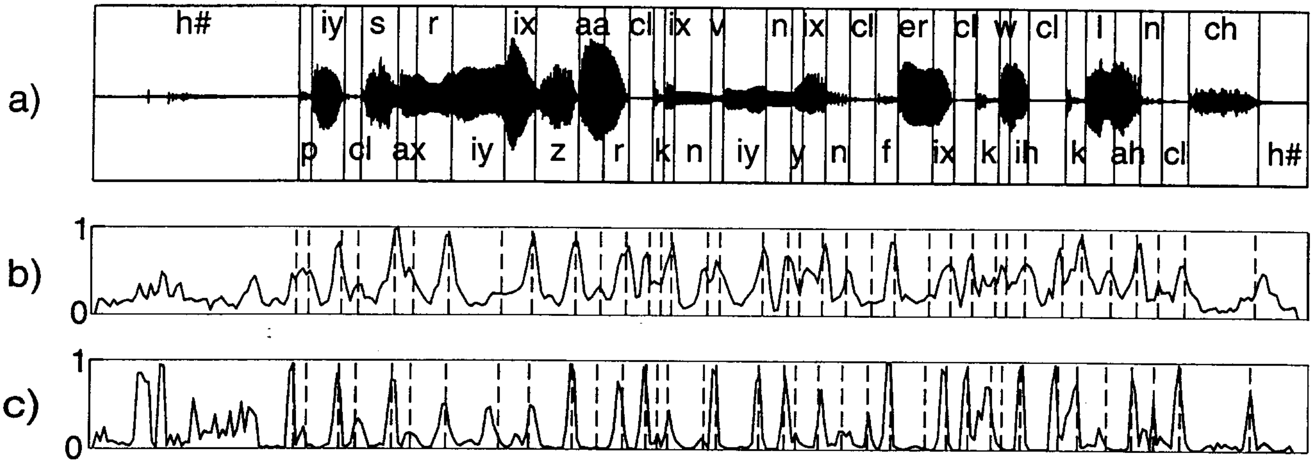

Figure 2: a) The TIMIT phone boundaries for the sentence: Pizzerias are convenient for a quick lunch. b) The change function formed by summing the delta cepstral coefficients. c) The change function found using the normalized scalar products.

cepstral dimension. We define the delta cepstral change function as follows:

$$d_k(t) = C_k(t+1) - C_k(t-1), \; k = 1, ..., K$$

$$d_{k,max} = \max_t |d_k(t)|$$

$$\hat{c_k}(t) = d_k(t)/d_{k,max}$$

$$\hat{c}(t) = \sum_{k=1}^{K} \hat{c_k}(t)$$

$$\hat{c}_{max} = \max_t \; \hat{c}(t)$$

$$c(t) = \hat{c}(t)/\hat{c}_{max} \qquad (3)$$

where $C_k(t)$ is the k^{th} cepstral coefficient for frame t and K is the number of cepstral coefficients. An example the the delta cepstral change function in shown in figure 2b.

The SVF change function estimates spectral change as the angle between two normalized spectral vectors (cepstral vectors in this study) that are separated in time by a fixed number of frames. The SVF defined in [1] has been modified slightly for this study. We calculate the SVF as follows:

$$\hat{c}(t) = \frac{\hat{C}(t-1) \cdot \hat{C}(t+1)}{\| \hat{C}(t-1) \| \; \| \hat{C}(t+1) \|}$$

$$\hat{c}_{max} = \max_t |\hat{c}(t)|$$

$$c(t) = 0.5 \times [1 - \hat{c}(t)/\hat{c}_{max}] \qquad (4)$$

where $\hat{C}(t)$ is the difference between the t^{th} cepstral vector, $C(t)$, and the time average of cepstral vectors that lie within a window centered at t, and "$\cdot$" indicates the scalar dot product operation. An example the the SVF change function in shown in figure 2c.

6. RESULTS

In table 1, we compare the recognition performance of a phone HMM with explicit segmentation to a standard HMM. The %correct is defined as the percentage of reference phone labels that were correctly recognized, and %accurate equals %correct minus the insertion rate.

In our experiments, we evaluate the utility of the various forms of segmentation information for three different methods of smoothing. For *SMOOTH*-2, the context-dependent output distributions are smoothed using only a uniform distribution. For *SMOOTH*-3, each context-dependent output distribution is mixed with the corresponding context-independent distribution as well as a uniform distribution. Lastly, model *SMOOTH*-49 mixes each context-dependent output distribution with the set of 47 related context-dependent output distributions (see section 3), the appropriate context-independent output distribution, and a uniform output distribution.

For the simple smoothing model, *SMOOTH*-2, the delta cepstral change function increases accuracy with only a slight degradation in %correct compared to using no segmentation information. For better smoothed models, however, the change function increases %correct slightly, but at the cost of a slightly lower %accuracy.

For all three methods of smoothing, the ideal change function reduced both insertions and deletions each by about 5% of the total number of reference phones, which added approximately 10% to the accuracy. When the TIMIT labels were used only to eliminate insertions (i.e., change function "no-insert"), both the %correct and %accurate increased by about 5%, which is consistent with the fact that the ideal function reduces 5% of the deletions.

For the simplest method of smoothing, *SMOOTH*-2, the delta-cepstral change function lowered the insertion rate by 1% compared to the standard HMM while having negligible affect on %correct.

For the intermediate method of smoothing, *SMOOTH*-

Smoothing	Type of Change Function				
Model	No Segmentation	Ideal	No-Insert	SVF	Delta-Cepstral
SMOOTH-2	67.1/58.3	70.9/69.3	-	-	67.0/59.2
SMOOTH-3	67.6/61.0	72.7/71.2	66.6/65.5	68.8/61.3	69.2/60.6
SMOOTH-49	68.4/63.0	73.8/72.5	68.3/67.2	69.0/62.4	69.3/61.7

Table 1: Results using various forms of segmentation information. Shown are %correct/%accurate.

3, both automatic methods reduced the number of deletions relative to the standard HMM, but at the cost of more insertions. The resulting overall accuracy (100 - %substitutions - %deletions - %insertions) improved slightly for SVF and dropped slightly for the delta-cepstral change function.

For the last method of smoothing, *SMOOTH*-49, both the SVF and the delta-cepstral change function led to more new insertions than corrected deletions compared to the standard HMM. The result is a slightly improved %correct at the cost of degraded %accuracy.

Of the two automatic methods, the SVF had higher accuracy while the delta-cepstral change function had a higher %correct. This difference becomes more pronounced as the smoothing is improved.

The more detailed method of smoothing yielded the best results for all change functions considered in this study. Compared to the more traditional method of mixing a context-dependent distribution with only the uniform distribution and the corresponding context-dependent distributions, the more general method consistently improved accuracy by one to two percent for all change functions.

7. DISCUSSION

The incorporation of explicit segmentation led to very modest increase in %correct for well smoothed models. However, the results fall considerably short of the upper bound improvement, as measured using hand-labeled segmentation.

One possible explanation for the discrepancy is that the cost of phone transitions is in general much smaller than other terms that are combined during Viterbi alignment. The log likelihood score is usually dominated by the output probabilities, for example. To emphasize the change function knowledge source, the segmentation information can be weighted in the same way as other HMM knowledge sources (e.g., "language match factor"). The results shown in table 1 reflect a weight of five. After evaluating several other weights as well as a number of additive offsets, we conclude that recognition accuracy is not sensitive to small variations in these constants.

A second explanation for the difference between ideal segmentation and automatically generated segmentation is that the change functions do not accurately predict a transition between some pairs of phones. Both of the change functions considered in this paper are oversimplified measures of change. It seems likely that some cepstral coefficients are better indicators of change than others, so that it would be reasonable to include weights w_k, $k = 0, ..., K$, in equation 3. It may also be beneficial to increase the resolution of the change function by using a higher frame rate for estimating the change function than is used in the HMM.

Currently, the supplemental segmentation information is knowledge-driven. A data driven approach would require that the HMM paradigm be modified so that nulls could be produced from one stream while other streams are non-null. Specifically, the stream corresponding to the change function would need to generate nulls for states that represent the interior of a phone segment. We are exploring some of these extensions.

8. REFERENCES

[1] G. Flammia, P. Dalsgaard, O. Anderson, and B. Lindberg. Segment-based variable frame rate speech analysis and recognition using a spectral variation function. In *Proc. International Conference on Spoken Language Processing*, 1992.

[2] J.N. Marcus. Phonetic recognition in a segment-based HMM. In *Proc. ICASSP*, 1993.

[3] K.F. Lee and H.W. Hon. Speaker-independent phone recognition using hidden Markov models. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 37(11), November 1989.

[4] K. Shikano. Evaluation of LPC spectral matching measures for phonetic unit recognition. Technical report, Carnegie Mellon University, 1986.

[5] K.F. Lee. *The Development of the SPHINX System.* Klewer Academic Publishers, 1989.

[6] F. Jelinek and R.L. Mercer. Interpolated estimation of Markov source parameters from sparse data. In E.S. Gelesma and L.N. Kanal, editors, *Pattern Recognition in Practice*, pages 381–397. North-Holland, Amsterdam, The Netherlands, 1980.