ON MODELING DURATION IN CONTEXT IN SPEECH RECOGNITION

Joseph Picone

Speech and Image Understanding Laboratory
Texas Instruments, Inc.
Dallas, Texas 75265

ABSTRACT

Although Hidden Markov Models (HMMs) can effectively encode duration information, seeding of an HMM with the correct duration information is important in obtaining high performance speech recognition. Hierarchical clustering techniques have been shown to be a powerful tool in Dynamic Time Warping systems for building speaker independent reference models, but direct application of clustering techniques to HMMs is not straightforward. In this paper, a clustering algorithm is introduced that allows clustering of HMM models directly. This clustering algorithm also determines the appropriate duration profile for a recognition unit. High performance speaker independent digit recognition on a studio-quality connected digit database is demonstrated using this algorithm.

INTRODUCTION

Hierarchical clustering techniques have been shown to be a powerful tool for building reference templates for speaker independent speech recognition using Dynamic Time Warping (DTW) [1]. Application of clustering techniques to continuous distribution Hidden Markov Model (HMM) based systems is not straightforward. Many existing HMM systems have bootstrapped existing tried and true reference template generation procedures for DTW. Clustering techniques, however, are typically most successful when they duplicate the distance comparison process used in the recognition system. The problem we examine in this paper is that of building optimal reference models for Hidden Markov Model systems.

Our experiences with HMM based speech recognition [2] indicate that one of the primary reasons HMMs give better performance than corresponding DTW systems is that the HMM system does a better job of modeling the time course of the speech signal (otherwise known as the problem of segmentation). We believe that this is a byproduct of the supervised training procedures that strictly enforce model optimization by optimizing the probability of the orthographic transcriptions of the utterances given a set of reference models. HMM models are continually refined in supervised training until they can quite accurately describe the time course of the training data. This has been verified using our HMM trace development tool [3]. Generally, when we observe the spectrographic trace of a recognition error, it is a result of a corresponding error in time alignment.

This time alignment problem is confounded by the other free dimension in HMM based recognition using continuous densities: choice of the number of states in an initial seed model. Performance, as we will show, even in the simplest of tasks is sensitive to the seeding process. Appropriate hand seeding of reference models generally requires an extensive knowledge of the recognition units and the application. We seek, therefore, a clustering procedure that, in addition performing some type of basic clustering for generation of spectral information in reference models, computes an optimal number of states for reference seed models.

Since the number of states in a model does not change in supervised training, it is crucial that the initial duration of a model be correct. Adding free variables to account fro more flexible duration models can many times degrade performance. Let us begin by demonstrating the importance of duration in seed models for an isolated digit recognition task.

THE IMPORTANCE OF SEED MODEL CONSTRUCTION

For the remainder of this paper we will consider digit recognition tasks on studio quality data, using the TI continuous digit database [3]. This is a 225 speaker database that has been dialectically balanced. Consider first the problem of isolated digit recognition. The recognition system we will use consists of a standard continuous density HMM system. One important feature of this system is that it uses a top-down control structure which models every frame of speech data. Explicit endpointing is not used.

The acoustic front-end is also a standard speech recognition front-end that uses LPC-derived principal spectral components, auxiliary energy measures, and differential features, as shown in Figure 1. The system uses a single, or pooled, transformation matrix to de-correlate features [2].

CH2673-2/89/0000-0421 $1.00 © 1989 IEEE

The histogram depicting the overall durations for the isolated digit portion of the database is shown in Figure 2. These were computed by handmarking the isolated digits. We see that the average duration of a digit is 400 msec, or 20 frames at a 20 msec frame rate. In this first experiment, we will use one model per digit. We choose a nominal exemplar for each digit, and sample it at frames rates of 20, 30, 40, 50, 60, 70, and 80 msec. This results in models that have 20, 14, 10, 8, 7, 6, and 5 states, respectively. Each model will have the same number of states (fixed state models).

Simple progressive HMM models were used (at each state either a self-transition can occur, or a transition to the next state). Hence, the implied duration model is a single exponential density at each state. These seed models were re-estimated on the database using supervised training, and evaluated. Performance is shown in Figure 3.

We see that best performance results when using 14 or 20 state models. Recall that the 20 state models correspond to representing every frame of reference data in the model. We have repeatedly seen that, with HMM recognition systems, performance is optimal when reference models model every frame of data. We also see that the initial number of states in a reference model strongly correlates with performance. Initial seeding of models with the proper duration model is important.

In the next experiment, we add skip states to these same models. The results are shown in Figure 3. Note that performance actually degrades for all models except the 14 and 20 state models. These models, because they model every frame of data, when trained, converge to the correct duration models. The other models, when skip states are added, have trouble converging to a solution as good as without skip states. This further highlights the tendency of HMM re-estimation to get stuck in local optimum, and not converge to a global optimum.

It is interesting to note that the 30 state model included in this experiment actually begins to take on the appearance of two models in one, because the initial 30 states are more than required for modeling a digit. The typical path through this model usually alternates between choosing even numbered states and odd numbered states.

Next, we compared the fixed state models with skip states to variable duration models. The variable duration models were constructed by choosing an initial seed model whose length (number of states) was the average duration for that digit. This effectively introduces a penalty into the HMM recognition for deviations from the nominal average duration of a digit. Performance is also shown in Figure 3. Here we see that variable duration models perform as well as the fixed state models.

HMM CLUSTERING WITH DURATION

Since we have demonstrated that it is important to seed an HMM with the models that have the optimal number of states, we seek a clustering algorithm that will automatically determine the number of states for seed models. K-MEANS clustering procedures [5] have been highly successful in both VQ applications [6], and speech recognition [1,2]. Here, we propose a simple modification to the K-MEANS procedure that allows duration to be included in the clustering procedure.

Each token in the training database for a particular digit is converted into an HMM with skip states by seeding each state in the model with an observation vector corresponding to each feature vector in the token. Since we model every frame of data, the transition probabilities are seeded such that the probability of progressing is equally as likely as staying or skipping. Thus, the mean duration of this model will be the actual duration of the token.

The K-MEANS clustering algorithm requires a distance matrix as input, which contains a distance value comparing every token to every other token. In previous DTW systems, this amounted to performing DTW matching and entering the resulting distance score in this matrix. We apply an analogous procedure to HMM's, except that the distance score now is computed as the probability of the model for token A given token B:

$$dist(A,B) = PROB(\text{model A/obs. for B})$$

Note that this distance metric is not symmetric, $dist(A,B) \neq dist(B,A)$.

This distance matrix is the input into the standard K-MEANS clustering procedure. The output of the K-MEANS clustering is a set of cluster centers, which are actually elements of the training set. The HMM seed models are then generated by performing re-estimation using only the elements in the cluster. The seed model is the cluster center, and the number of states in this seed model is taken to be the number of states in the cluster center. The cluster center is re-estimated by using supervised training over all elements in the cluster. The output of the process, the recognition seed models, are then typically re-estimated over the entire training database.

This clustering algorithm represents two significant departures from previous algorithms. First, a probabilistic distance measure is used. This distance measure computes the distance between token A and token B as the probability of token A given an HMM model for token B. Second, and most importantly, the

duration of a model is factored into the distance measure, thereby allowing the clustering to compute cluster centers that are optimal with respect to spectral information and duration. Since this clustering is performed on recognition units, the duration is captured in the context that it was produced.

Rather than decoupling duration from the model, such as using some postprocessing type measures [7], duration profiles for recognition units can be explicitly modeled. This is important for large recognition units such as whole word models, where duration has a strong acoustic correlate. Capturing duration differences in context allows the HMM model to effectively encode this information. We believe such duration in context issues are important even at the phonetic level.

EXPERIMENTAL RESULTS

The experiment described above was repeated, this time using one cluster per digit. The training database was clustered using the duration in context clustering. The performance, shown in Figure 3, was slightly better, than the variable state system.

There are three important observations to be made. First, in the case of one cluster, the number of states in each model was not simply equal to the average duration of the digit in frames. The models produced by clustering had an average of 16.5 states per digit, while the variable number of states experiment had an average of 21.2 states per digit. Clustering produced models in an automated way that had on the average 20% fewer states. Recalling that these digits were handmarked, this difference could not be attributed to experimental error.

The second observation is that performance improves, of course, with the number of clusters. As the number of clusters increases, the duration profile for the digit models more than just the duration histogram for the digit. For instance, in the case of the digit "ONE". The mean duration of the digit "ONE" in the training database is 18.4 frames, with a standard deviation of 3.8. The clustering algorithm, for the case of seven clusters, chooses six clusters that are within one standard deviation of the mean (20, 19, 21, 19, 15, 13, and 17 states), and one cluster that is outside one standard deviation (13 states).

Thus, the clustering, in addition to modeling spectral variations, is modeling duration in an "optimal" sort of way. (This is verified by the fact that 4 of the seven clusters were associated with male speakers, and the other three females. Also, the performance with two clusters (one male/one female of approximately the same number of states) indicates that spectral variations take precedence over duration until there are enough degrees of freedom to model both.

This clustering procedure has been evaluated on a connected digit recognition task. Duration information is, in fact, far more significant in continuous digit recognition. Using the 8 cluster per digit system, and the pooled covariance approach, a 4.7% string error rate was achieved. This was reduced to 3.5% by invoking state specific transformations [2]. This is an unknown string length experiment in which recognition utterances can be any length. Performance in the 8 cluster case was further improved to 3.1% by incorporating a sex constraint, that is, forcing an input utterance to be recognized with either all male or all female models.

CONCLUSIONS

We have introduced a clustering procedure that allows direct generation of HMM seed models. Good performance was demonstrated on a digit recognition task. future work will be directed towards developing more powerful methods of investigating contextual effects. We have recently developed an architecture for speech recognition that appears to be a very promising approach for modeling acoustic context [8].

Efficiently modeling contextual information in the speech recognition process is very important to clustering. Clustering seems to consistently be able to identify and represent various acoustic contexts. However, we introduce these models into the recognition process with no guarantee that because they are good representational models, they will also be good discrimination models. On the other hand, even modeling a simple contextual parameter, such as sex, requires an unacceptable increase in complexity in an HMM. Future research will be directed towards using more powerful grammar formalisms to model acoustic context.

REFERENCES

[1] S.E. Levinson, L.R. Rabiner, A.E. Rosenburg, and J. Wilpon, "Interactive Clustering Techniques for Selecting Speaker-Independent Reference Templates for Isolated Word Recognition," IEEE Trans. Acoust., Speech, and Signal Proc., Vol. ASSP-27. No. 2, pp. 134-141, April 1979.

[2] J. Picone, G.R. Doddington, and J.J. Godfrey, "A Layered Grammar Approach to Speaker Independent Speech Recognition", presented at the 1988 IEEE Speech Recognition Workshop at Harriman, NY, June 1988.

[3] G.R. Doddington, J. Picone, and J.J. Godfrey, "The LPC trace as an HMM development tool," Journal Acoust. Soc. Am., Suppl. 1, Vol. 84, Fall 1988.

[4] R.G. Leonard, "A Database For Speaker-Independent Digit Recognition," Proc. 1984 IEEE Int. Conf. Acoust., Speech, and Signal Proc., pp.42.11.1-42.11.4, April 1984.

[5] M.R. Anderberg, Cluster Analysis For Applications, Academic Press, New York, 1973.

[6] J. Picone and G.R. Doddington, "Low Rate Speech Coding Using Contour Quantization," Proc. 1987 IEEE Int. Conf. Acoust., Speech, and Signal Proc., pp.1653-1656, April 1987.

[7] L.R. Rabiner, J.G. Wilpon, and B.H. Juang, "A model-based connected-digit recognition system using either hidden Markov models or templates," Computer Speech and Language, Vol.#1, pp.#167-197, 1986.

[8] C. Hemphill and J. Picone, "Robust Speech Recognition in a Unification Grammar Framework," to be published in the Proc. of the 1989 Int. Conf. on Acoust., Speech, and Signal Proc. in Glasgow, Scotland.

PRECONDITIONING
- Sample at 8 kHz
- First-Order Difference

ACOUSTIC MODEL
- 10-th Order LPC Autocorrelation-Based
- 20 ms Frame Period
- 30 ms Window Length, Hamming Weighted

SPECTRAL CONVERSION (14 MEASURES)
- 14 MEL-Spaced Filters
- Normalized Filter Amplitude (Log)

AUXILIARY MEASURES (3 MEASURES)}
- Speech Level
- Normalized Frame Energy (Log)
- T-Function

DYNAMIC MEASURES (15 MEASURES)}
- 40 ms Difference Of Frame Energy
- 40 ms Difference Of Spectral Amplitude

FEATURE SELECTION - PRINCIPAL COMPONENTS
- Select 10 Most Significant
  - Spectral Eigenvectors
- Select 4 Most Significant
  - Spectral Difference Eigenvectors

FEATURE TRANSFORMATIONS (18 FEATURES)
- Pooled Covariance Transformation

Figure 1. The Acoustic Front-End

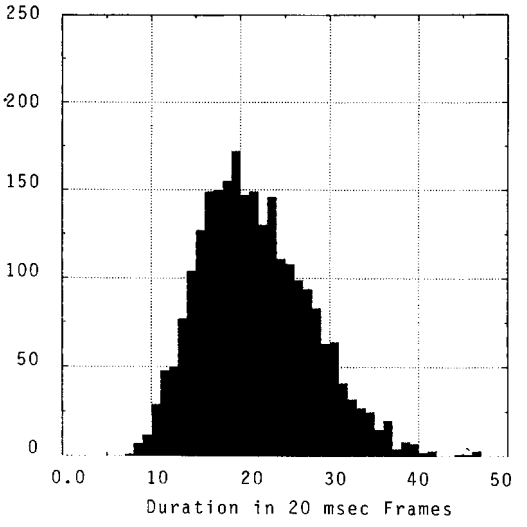

Figure 2. A Histogram of Durations For Isolated Digits

(a) Fixed States:

# of States	Prog. Models	With Skips
5	6.2%	7.4%
6	4.4%	6.7%
7	3.6%	4.4%
8	2.2%	2.9%
10	1.4%	1.6%
14	1.3%	0.7%
20	1.7%	0.7%
30	High	0.7%
Variable		0.8%

(b) Clustering Performance

No. of Clusters:	Word Error Rate
1	0.7%
2	0.3%
3	0.4%
4	0.3%
5	0.3%
6	0.3%
7	0.2%
8	0.2%
16	0.3%

Figure 3. Performance On Isolated Digit Recognition Using Clustering (Percent Word Errors)