Automatic Segmentation of Speech at the Phonetic Level*

Jon Ander Gómez and María José Castro

Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia, Valencia, Spain
{jon,mcastro}@dsic.upv.es

Abstract. A complete automatic speech segmentation technique has been studied in order to eliminate the need for manually segmented sentences. The goal is to fix the phoneme boundaries using only the speech waveform and the phonetic sequence of the sentences.
The phonetic boundaries are established using a Dynamic Time Warping algorithm that uses the *a posteriori* probabilities of each phonetic unit given the acoustic frame. These *a posteriori* probabilities are calculated by combining the probabilities of acoustic classes which are obtained from a clustering procedure on the feature space and the conditional probabilities of each acoustic class with respect to each phonetic unit.
The usefulness of the approach presented here is that manually segmented data is not needed in order to train acoustic models. The results of the obtained segmentation are similar to those obtained using the HTK toolkit with the "flat-start" option activated. Finally, results using Artificial Neural Networks and manually segmented data are also reported for comparison purposes.

1 Introduction

The automatic segmentation of continuous speech using only the phoneme sequence is an important task, specially if manually pre-segmented sentences are not available for training. The availability of segmented speech databases is useful for many purposes, mainly for the training of phoneme-based speech recognizers [1]. Such an automatic segmentation can be used as the primary input data to train other more powerful systems like those based on Hidden Markov Models (HMMs) or Artificial Neural Networks (ANNs).

In this work, two different Spanish speech databases composed of phonetically balanced sentences were automatically segmented. The phonetic boundaries are established using a Dynamic Time Warping algorithm that uses the *a posteriori* probabilities of each phonetic unit given the acoustic frame. These *a posteriori* probabilities are calculated by combining the probabilities of acoustic classes which are obtained from a clustering procedure on the feature space and the conditional probabilities of each acoustic class with respect to each phonetic unit.

* Work partially supported by the Spanish CICYT under contract TIC2000-1153.

T. Caelli et al. (Eds.): SSPR&SPR 2002, LNCS 2396, pp. 672–680, 2002.

2 Description of the System

The core of the approach presented here is the estimation of $P(ph_u|x_t)$, that is, the *a posteriori* probability that the phonetic unit ph_u has been uttered given the feature vector x_t, obtained at every instant of analysis t. When this probability is broken down using the Bayes rule, we obtain:

$$P(ph_u|x_t) = \frac{P(ph_u) \cdot p(x_t|ph_u)}{\sum_{i=1}^{U} P(ph_i) \cdot p(x_t|ph_i)} \tag{1}$$

where U is the number of phonetic units used in the system, and $P(ph_u)$ is the *a priori* probability of ph_u. In this approach, we assume $P(ph_u) = 1/U$ for all units, so it can be removed from expression (1).

Now, we need to calculate $p(x_t|ph_u)$, which is the conditional probability density that x_t appears when ph_u is uttered. To do so, a clustering procedure to find "natural" classes or groups in the subspace of $\mathbb{R}^d$ formed by the feature vectors is done. From now on, we will refer to this subspace as "feature space". Once the clustering stage has been completed, we are able to calculate $P(w_c|x_t)$, that is, the *a posteriori* probability that a class w_c appears given an input feature vector x_t, applying the Bayes rule as follows:

$$P(w_c|x_t) = \frac{P(w_c) \cdot p(x_t|w_c)}{\sum_{i=1}^{C} P(w_i) \cdot p(x_t|w_i)} \tag{2}$$

where C is the number of "natural" classes estimated using the clustering procedure, $P(w_c)$ is the *a priori* probability of the class w_c, and $p(x_t|w_c)$ is the conditional probability density calculated as Gaussian distributions. In this work, we assume $P(w_c) = 1/C$ for all classes.

At this point, the conditional probability densities $p(x_t|ph_u)$ from equation (1) can be estimated from the models learned using the clustering procedure. The "natural" classes make a partition of the feature space which is more precise than the phoneme partition. Since we already have $p(x_t|w_c)$ from the clustering procedure, $p(x_t|ph_u)$ can be approximated as

$$p(x_t|ph_u) \approx \sum_{c=1}^{C} p(x_t|w_c) \cdot P(w_c|ph_u) \tag{3}$$

where $P(w_c|ph_u)$ is the conditional probability that the class w_c is observed when the phonetic unit ph_u has been uttered (see how to obtain these conditional probabilities in section 2.2).

Given that the *a priori* probabilities of the phonetic units $P(ph_u)$ are considered to be equal, we can rewrite equation (1) using (3) as

$$P(ph_u|x_t) = \frac{\sum_{c=1}^{C} p(x_t|w_c) \cdot P(w_c|ph_u)}{\sum_{i=1}^{U} \sum_{c=1}^{C} p(x_t|w_c) \cdot P(w_c|ph_i)}, \tag{4}$$

which is the *a posteriori* probability we were looking for.

2.1 Clustering Procedure

One of the underlying ideas of this work is that we do not know how many different acoustical manifestations can occur for each phoneme from a particular parametrization. The obtained acoustical feature vectors form a subspace of $\mathbb{R}^d$. We assume that this subspace can be modeled with a Gaussian Mixture Model (GMM), where each class or group is identified by its mean and its diagonal covariance matrix. In our case, the *a priori* probabilities of each class or group, $P(w_c)$, are considered to be equal to $1/C$. The unsupervised learning of the means and the diagonal covariances for each class have been done by maximum likelihood estimation as described in [3, chapter 10].

The number of classes C has been fixed after observing the evolution of some measures which compare the manual segmentation with the automatic one (see section 3 and Figure 1). Once the number of classes is fixed and the parameters which define the GMM are learned, we can calculate the conditional probability densities $p(x_t|w_c)$. Then, the probabilities $P(w_c|x_t)$ are obtained as shown in equation (2).

2.2 Coarse Segmentation and Primary Estimation of the Conditional Probabilities

We need a segmentation of each sentence for the initial estimation of the conditional probabilities $P(w_c|ph_u)$. This first coarse segmentation has been achieved by applying a set of acoustic-phonetic rules knowing only the phonetic sequence of the utterance. The phonetic sequence of each sentence is automatically obtained from the orthographic transcription using a grapheme-to-phoneme converter [4].

The coarse segmentation used at this stage is done by:

1. Searching for relative maxima and minima over the speech signal based on the energy.
2. Associating maxima with vowel or fricative units and minima with possible silences.
3. Estimating the boundaries between each unit by simple spectral distances.

Searching for relative maxima and minima over the speech signal based on the energy. The location of relative maxima is restricted to those instants t where the energy is greater or equal to the energy at the interval of ± 30 ms. around t. Each maximum is considered to be more or less important depending on whether its energy is greater or smaller than a threshold for maxima calculated specifically for each sentence. The importance of a maximum is used to properly weight its deletion. The location of relative minima is done by searching for intervals where the energy is under a threshold for minima, which is also calculated for each sentence. After this step, we have a list of maxima and minima $(m_1, m_2, \ldots, m_{|m|})$ for the sentence.

Association of maxima with vowel or fricative units and minima with possible silences. The association of phonetic units (vowels or fricative consonants) is performed by a Dynamic Time Warping (DTW) algorithm that aligns the list of maxima and minima $(m_1, m_2, \ldots, m_{|m|})$ with the phonetic sequence $p_1 p_2 \ldots p_{|p|}$. The DTW algorithm uses the following set of productions:

- $\{(i-1, j-1), (i, j)\}$: Location of the phonetic unit p_j around the instant which m_i occurs. If m_i is a maximum, p_j is a vowel or a fricative consonant; if it is a minimum, p_j is a silence.
- $\{(i, j-1), (i, j)\}$: Insertion of phonetic unit p_j, which is not associated with any maximum or minimum.
- $\{(i-1, j), (i, j)\}$: Deletion of maximum or minimum m_i.
- $\{(i-1, j-\delta), (i, j)\}$, with $\delta \in [2..5]$: To align consecutive vowels (such as diphthongs).

Each production is properly weighted; for instance, the weight of the insertion of possible silences between words is much lower than the weight of the insertion of a vowel. In the case of several continuous vowels, the association of this subsequence with a maximum is also allowed.

The association of vowels and fricative consonants with a maximum is weighted using a measure which is related to the first MFCC (CC1). Fricative consonants, when associated with a maximum, have a cost which is calculated by differentiation of the CC1 with a threshold for fricatives, which is estimated for each sentence. This differentiation is also used for the vowel "i", and inverted in the case of the vowels "a", "o" and "u".

Estimation of the boundaries between each phonetic unit by simple spectral distances. After the association is done, we have some phonetic units (vowels and fricative consonants) located around the instant where its associated event (maximum or minimum) was detected. For instance, we could have the following situation:

$$\begin{array}{ccccccc} \ldots & m_i & & & & m_{i+1} & \ldots \\ \ldots & \text{a} & \text{r} & \text{d} & & \text{o} & \ldots \end{array}$$

We then take subsequences of units to locate the boundaries. These subsequences are formed by two units which are associated with an event (in the example, "a" and "o") and the units between them ("r" and "d" in the example). The boundaries are located by searching for relative maxima of spectral distances. The interval used to locate the boundaries begins at the position where the event m_i is located, and it ends where m_{i+1} is located. The Euclidean metric is used as the spectral distance, which is calculated using the feature vectors before and after instant t, as $||x_{t-1} - x_{t+1}||$.

At this point, with a segmented and labeled sentence, the estimation of each joint event (w_c, ph_u) can be carried out as its absolute frequency. The conditional probabilities $P(w_c|ph_u)$ are calculated by normalizing with respect to each ph_u. Now, we can calculate $p(x_t|ph_u)$ as in equation (3).

2.3 Conditional Probability Tuning

At this point, we can apply both a DTW algorithm, which uses the *a posteriori* probabilities $P(ph_u|x_t)$ obtained as in equation (4), and the phonetic sequence to segment a sentence. This algorithm assigns a phonetic unit ph_u to an interval of the signal in order to minimize the measure

$$\sum_{f=1}^{F} \sum_{t=t^0_{ph_f}}^{t^1_{ph_f}} -\log P(ph_f|x_t) \tag{5}$$

where F is the number of phonetic units of the sentence, and $t^0_{ph_f}$ is the initial frame of ph_f and $t^1_{ph_f}$ the final frame.

When the DTW algorithm is used to segment all the sentences of the training corpus, we obtain a new segmentation, which is used to make a new estimation of the absolute frequency of each joint event (w_c, ph_u). Then, the conditional probabilities $P(w_c|ph_u)$ are recalculated by normalizing with respect to each ph_u. This process is repeated until the difference between all the conditional probabilities $P(w_c|ph_u)$ of two continuous iterations is smaller than an ϵ (we use $\epsilon = 0.01$).

To perform this iterative tuning process do the following:

1. Initialize the absolute frequencies to 0.
2. For each sentence of the training corpus:
   (a) Estimate $P(ph_u|x_t)$ using equation (4).
   (b) Segment minimizing equation (5).
   (c) Increment the absolute frequencies.
3. Calculate the new conditional probabilities $P(w_c|ph_u)$ from the new absolute frequencies.
4. If the difference between the conditional probabilities is smaller than ϵ, then finish, otherwise go to step 1.

3 Evaluation

The measures used to evaluate the performance of the segmentation were extracted from [5]. The percentage of correctly located boundaries (*PB*) compares the location of automatically obtained phoneme boundaries with the location of manually obtained reference boundaries. The *PB* is the percentage of boundaries located within a given distance from the manually set boundaries. Tolerance intervals of 10 ms. to 30 ms. are considered.

The second measure used in this work is the percentage of frames (*PF*) which matches both segmentations, the automatic one and the manual one. Other measures are calculated using the ratios C_{man}, and C_{aut} for each phonetic unit:

$$C_{man} = \frac{correct}{tot\text{-}man} \times 100 \qquad C_{aut} = \frac{correct}{tot\text{-}aut} \times 100$$

where *correct* is the number of frames matching both segmentations, *tot-man* is the total number of frames in the manual segmentation for each phonetic unit, and *tot-aut* is the total number of frames in the automatic segmentation.

These ratios allow us to determine the type of segmentation error for each phonetic unit. A low value of C_{man} indicates a tendency of the system to assign shorter segments than needed to the unit under consideration. A low value of C_{aut} indicates a tendency to assign longer segments than needed.

4 Experiments and Results

The experiments performed in this work were performed using two Spanish speech databases composed of phonetically balanced sentences. The first one (*Frases*) was composed of 170 different sentences uttered by 10 speakers (5 male and 5 female) with a total of 1,700 sentences (around one hour of speech). The second one was the *Albayzin* speech database [2], from which we only used 6,800 sentences (around six hours of speech) which were obtained by making subgroups from a set of 700 distinct sentences uttered by 40 different speakers.

Each acoustic frame was formed by a d-dimensional feature vector: energy, 10 mel frequency cepstral coefficients (MFCCs), and their first and second time derivatives using a ±20 ms. window, which were obtained every 10 ms. using a 25 ms. Hamming window. A preemphasis filter with the transfer function $H(z) = 1 - 0.95z^{-1}$ was applied to the signal.

For the *Frases* database, 1,200 sentences were used for training. First of all, the feature vectors of these sentences were clustered to find "natural" classes. The phonetic sequence of each sentence was obtained using a grapheme-to-phoneme converter [4]. The coarse segmentation of each sentence was obtained using the sequence of phonetic units uttered and the acoustic-phonetic rules explained in section 2.2. The initial values of the conditional probabilities $P(w_c|ph_u)$ were estimated using this coarse segmentation and the clusters. Next, the tuning process to re-estimate the conditional probabilities is iterated by segmenting the sentences with a DTW algorithm. Finally, the segmentation of the test sentences was carried out for evaluation purposes.

A subset of 77 manually segmented sentences was used for testing. Table 1 shows the results obtained with 300 "natural" classes and the results reported in [5] using HMMs for the same corpus and the same test sentences. In the case of HMMs, the same 77 manually segmented sentences were also used for training. From Table 1, it can be observed that our automatic system performs slightly better than the HMM approach. The same segmentation task was carried out using the HTK toolkit [6] with the "flat-start" option activated. Our automatic procedure and the HTK toolkit led to similar results.

In addition, we trained a Multilayer Perceptron (MLP) with the 77 manually segmented sentences to estimate the *a posteriori* probabilities of the phonetic units given the acoustic input. In this case, no derivatives were used: the input to the MLP was composed of a context window of nine acoustic frames, each of which was formed by energy plus 10 MFCCs. An MLP of two hidden lay-

Table 1. Percentage of frames (*PF*) matching both segmentations, and percentage of correct boundaries (*PB*) within tolerance intervals of different lengths (in ms.) for the *Frases* database.

	Frases ($C = 300$)			
	PF	*PB*		
		10	*20*	*30*
Automatic	82.1	67.9	85.1	93.0
HMMs	81.7	67.7	82.4	91.1
HTK	82.2	69.8	85.1	90.1
MLP+DTW	94.2	93.1	97.2	98.3

ers of 100 units each was trained achieving a classification error of around 6% (at frame level). In order to have a biased result to compare our system to, we resegmented the same 77 sentences using the trained MLP and the DTW segmentation algorithm. The result of this experiment is also shown in Table 1. As might be expected, the results of the closed-experiment (the same manually segmented training data and testing data) using the MLP were much better than our automatic approach, which did not use manual segmentation at all.

The same procedure was applied to the phonetic corpus of the *Albayzin* database. In this case, the number of sentences used was 6,800. They were divided into two subsets, one of 5,600 sentences used for training, and the other of 1,200 sentences used for testing. These 1,200 test sentences were manually segmented.

A subset of 400 sentences was selected out of the training sentences to do the clustering and to obtain the initial conditional probabilities. All the 5,600 training sentences were used in the iterative tuning process to adjust the conditional probabilities. In order to select the number of classes C, the whole experiment was carried out for increasing values of C, from 80, 100, 120, ..., 500 (see Figure 1). From this graph, it can be seen that performance is similar for values of C above 120.

The results obtained for the *Albayzin* database are shown in Table 2 and Figure 2 (performance of our automatic procedure is given for 400 "natural" classes). As before, the same task was performed by using the HTK toolkit with the "flat-start" option activated. An experiment with a MLP was also performed using the same 1,200 manually segmented sentences for training and testing. The results were quite similar to those obtained by the other speech database. Thus, the system for automatic segmentation can be scaled to any speech database.

5 Conclusions

In this work, we have presented a completely automatic procedure to segment speech databases without the need for a manually segmented subset. This task is important in order to obtain segmented databases for training phoneme-based speech recognizers.

Table 2. Percentage of frames (*PF*) matching both segmentations, and percentage of correct boundaries (*PB*) within tolerance intervals of different lengths (in ms.) for the *Albayzin* database

	Albayzin ($C = 400$)			
	PF	*PB*		
		10	*20*	*30*
Automatic	81.3	70.5	87.1	93.4
HTK	82.8	72.9	84.3	87.7
MLP+DTW	83.8	80.6	89.0	92.5

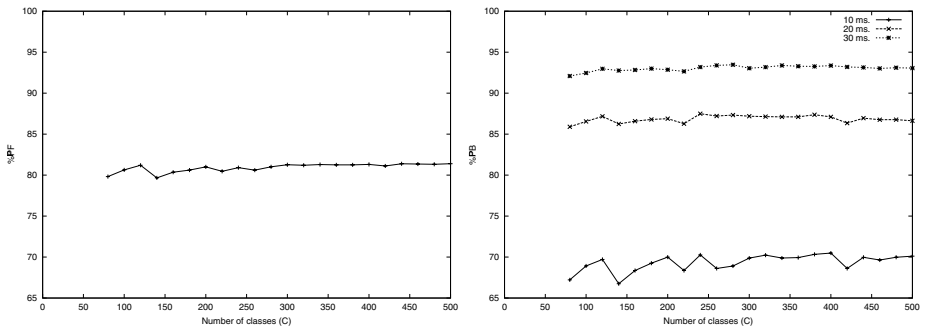

Fig. 1. *Left:* Percentage of frames (*PF*) matching manual segmentation and automatic segmentation versus the number of classes C for the *Albayzin* database. *Right:* Percentage of correct boundaries (*PB*) within tolerance intervals of different lengths (in ms.) versus the number of classes C for the *Albayzin* database

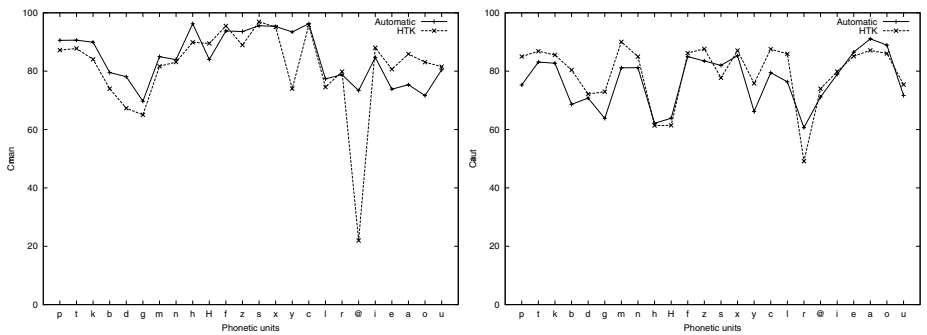

Fig. 2. C_{man} and C_{aut} for the phonetic units (SAMPA allophones) for the automatic segmentation (with 400 classes) and for the segmentation obtained using the HTK toolkit for the *Albayzin* database

As future extensions, we plan to increment the ratio of analysis from 10 ms. to 5 ms. to obtain better representations of the acoustical transitions, specially the burst of the plosive consonants. We also plan to extend the feature vectors with a contextual window of acoustic frames. We hope that the incorporation of these extensions will significantly increase the accuracy of the obtained segmentation.

References

1. B. Angelini, F. Brugnara, D. Falavigna, D. Giuliani, R. Gretter, and M. Omologo. Automatic Segmentation and Labeling of English and Italian Speech Databases. In *Eurospeech93*, volume 3, pages 653–656, Berlin (Germany), September 1993. 672
2. A. Moreno, D. Poch, A. Bonafonte, E. Lleida, J. Llisterri, J. B. Mariño, and C. Nadeu. Albayzin Speech Database: Design of the Phonetic Corpus. In *Eurospeech93*, volume 1, pages 653–656, Berlin (Germany), September 1993. 677
3. R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, 2001. 674
4. María José Castro, Salvador España, Andrés Marzal, and Ismael Salvador. Grapheme-to-phoneme conversion for the Spanish language. In *Proceedings of the IX National Symposium on Pattern Recognition and Image Analysis*, pages 397–402, Benicàssim (Spain), May 2001. 674, 677
5. I. Torres, A. Varona, and F. Casacuberta. Automatic segmentation and phone model initialization in continuous speech recognition. *Proc. in Artificial Intelligence*, I:286–289, 1994. 676, 677
6. Steve Young, Julian Odell, Dave Ollason, Valtcho Valtchev, and Phil Woodlan. *The HTK Book*. Cambridge University, 1997. 677