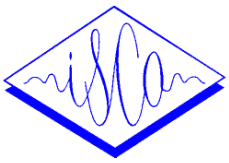

2nd European Conference on
Speech Communication and Technology
EUROSPEECH '91
Genova, Italy, September 24-26, 1991

# ENTROPIC TRAINING FOR HMM SPEECH RECOGNITION

Antonio M. Peinado, Ramon Roman *, Jose C. Segura,
Antonio J. Rubio, Pedro Garcia, Jesus E. Diaz

Departamento de Electronica y Tecnologia de Computadores
Universidad de Granada, 18071 GRANADA (Spain)

**Abstract** - *The segmentation of the training data has become the most used initialization method for HMM training. The Viterbi reestimation has been widely applied for that purpose. We introduce a new segmentation method, based on the maximization of the Entropic Cohesion Measure (ECM) between segments and observations, which is equivalent to minimize the entropy model. This maximization is carried out by looking for the optimal boundaries between segments of the training utterances. Thus, we obtain an optimal segmentation (in the ECM sense) that achieves similar performance to the Viterbi reestimation with a considerable computational saving.*

10.21437/Eurospeech.1991-160

## 1. Introduction

In the last years, several training methods for HMMs have been proposed. A standard method [1] consists of using the Baum-Welch reestimation. This reestimation is sensitive to the initial model (specially the **B** parameters). A manual or automatic segmentation can be used to solve the initialization problem. The Viterbi reestimation procedure (started with a linear segmentation) has been widely applied as initialization method.

Also, the complexity of the algorithms forces algorithm designers to look for computational savings. In this work, we consider a new training procedure based on an entropic segmentation criterion, with a considerable computational cost saving. This criterion is the maximization of the Entropic Cohesion Measure (ECM) [2] among segments and observations or, equivalently, the minimization of the entropy model over the segment (or state) boundaries. This criterion can be used to perform an entropic classification, based on the idea that when a segment is split, the cohesion lost by the data must be minimum. The computational saving is based on the fact that this minimization can be performed in a differential way. We develop this method with discrete models, but it can be straightforwardly extended to continuous ones.

* Dpto. de Fisica Aplicada
Universidad de Granada, Spain

## 2. Theoretic basis of the segmentation

Given a set of symbols $V=(v_1, \cdots, v_M)$ and the set of segments (that we identify with states) $S=(s_1, \cdots, s_N)$, the cohesion (ECM) between the components $V$ and $S$ of the probability space $V \otimes S$ is defined [2] as,

$$W(V \otimes S; V, S) = H(V) + H(S) - H(V \otimes S) \qquad (1)$$

where $H$ is the entropy function. According to this definition, the ECM is equal to the Mutual Information between $V$ and $S$,

$$W(V \otimes S; V, S) = I(V, S) = H(V) - H(V \mid S) \qquad (2)$$

where,

$$H(V \mid S) = \sum_{i=1}^{N} p(s_i) h_i \qquad (3)$$

$$h_i = H(V \mid s_i) = -\sum_{k=1}^{M} p(v_k \mid s_i) \log(p(v_k \mid s_i)) \qquad (4)$$

(we take the occurrence frequencies as probabilities).

Our segmentation criterion consists of maximizing the ECM, or, equivalently, of maximizing the Mutual Information $I(V, S)$, or, equivalently, of minimizing the entropy $H(V \mid S)$ (since $H(V)$ depends

only on the training data and not on the segmentation). If we consider $H(V|S)$ as the Markov source entropy [2], the probability of the typical utterances emitted from such a source is about $2^{-LH}$ ($L$ is the utterance length) and $2^{LH}$ the number of them. Our segmentation criterion consists on minimizing the model entropy, which is equivalent to maximize the probability of the typical utterances.

## 3. Entropic Segmentation

The problem now is how to minimize $H(V|S)$. For a given segmentation of the training data, we can compute $H(V|S)$. We have observed that if we move the boundary between two segments of a given training utterance, $H(V|S)$ varies with it, as it is shown in figure 1. That graphic presents a minimum in the range of variation of boundary (that is, from the previous to the next boundary).

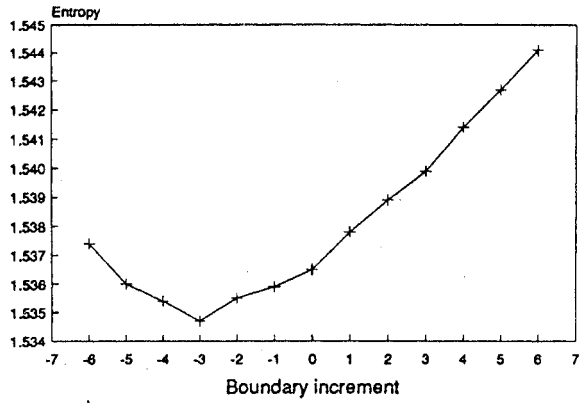


*Figure 1.- Model entropy vs. the boundary increment of a split training utterance.*

Based on the above fact, we propose an *entropic segmentation* algorithm that minimizes $H(V|S)$. This algorithm has the structure of a k-means algorithm, limited by the sequential nature of the problem. We suppose that we have $P$ training utterances $O^j=O^j_1,O^j_2,\cdots,O^j_{T^j}$ $(j=1,...,P)$. We assume left-to-right models. The algorithm consists of the following steps:

1) A linear segmentation of the training data is performed for getting $N$ initial segments and $N$ boundaries $(l_j(1),l_j(2),\cdots,l_j(N)=T^j)$ on each training utterance $O^j$, as depicted in figure 2. From this segmentation, we have to compute the matrices $p(v_k|s_i)$ $(k=1,...,M\ \ i=1,...,N)$ and $p(s_i)$ $(i=1,...,N)$, and the entropy $H(V|S)$.

2) For each segment $s_i$ $(i=1,...,N-1)$ and for each utterance $O^j$, probabilities $p(s_i)$ and $p(s_{i+1})$, and vectors $p(v_k|s_i)$ and $p(v_k|s_{i+1})$ are temporally recomputed for $\Delta l_j(i)=-\delta,\cdots,+\delta$. We set $l_j(i)=l_j(i)+\Delta l^*_j(i)$, where $\Delta l^*_j(i)$ is the increment on $l_j(i)$ that minimizes the sum (with the restriction of $l_j(i-1)\le l_j(i)\le l_j(i+1)$),

$$p(s_i)h_i+p(s_{i+1})h_{i+1} \qquad (5)$$

which is equivalent to minimize $H(V|S)$, since all the segments (except $s_i$ and $s_{i+1}$) are not modified. It is important to note that $p(s_i)$, $p(s_{i+1})$, $p(v_k|s_i)$ and $p(v_k|s_{i+1})$ are only temporally recomputed to estimate (5) for each $\Delta l_j(i)$. We have used $\delta=1$ in order to avoid unnecessary computation.

3) The matrices $p(s_i)$ and $p(v_k|s_i)$ are computed from the segmentation obtained in the previous step.

4) The entropy $H(V|S)$ is calculated from the matrices obtained in step 3, and is compared with the value of $H(V|S)$ from the previous iteration. If the relative difference between them is under a certain threshold, the procedure finishes, and, otherwise, goes to step 2 again.

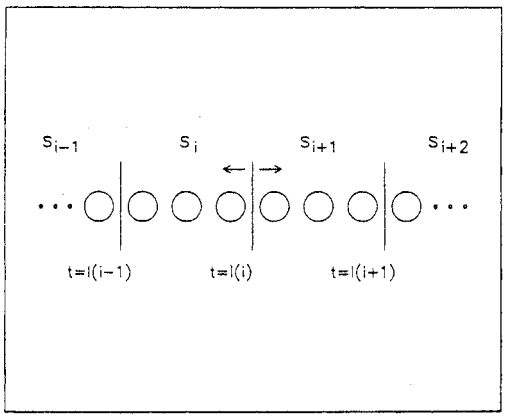


*Figure 2.- Segmentation of a training utterance* **O**.

One of the first differences we point out, related to the Viterbi algorithm, is that the *jump number* (that is, the number of forward transitions allowed from a given state to different states) can be dynamically determined, since we let $l_j(i)$ to be from $l_j(i-1)$ to $l_j(i+1)$, while the Viterbi algorithm usually uses a fixed *jump number*.

## 4. Determination of entropy increments

An important handicap of the above algorithm is the high computational cost of step 2. But what is really important is to know how the entropy $H(V|S)$ increases when the boundary between two segments $l(i)$ of a given training utterance **O** is shifted. Since we use $\delta$=1, the increments in $p(v_k|s_i)$ and $p(s_i)$ are very small, so we can approximate $\Delta H(V|S)$ by a first order increment.

$$\Delta H(V|S)=\Delta(p(s_i)h_i+p(s_{i+1})h_{i+1}) \quad (6)$$

We have to differentiate the expression (5) to get the increment (6). Taking into account that

$$\Delta(p(v_k|s_i)log(p(v_k|s_i))=log(ep(v_k|s_i))\Delta p(v_k|s_i) \quad (7)$$

we can rewrite,

$$\Delta H(V|S)=-\sum_{r=i}^{i+1}\sum_{k=1}^{M}\Big[p(s_r)log(ep(v_k|s_r))\Delta p(v_k|s_r) + p(v_k|s_r)log(p(v_k|s_r))\Delta p(s_r)\Big] \quad (8)$$

The increments $p(s_i)$ and $p(v_k|s_i)$ are calculated as,

$$\Delta p(s_i)=\frac{\Delta N_i}{N_T} \quad (9)$$

$$\Delta p(v_k|s_i)=\frac{\Delta n_i(v_k)}{N_i}-p(v_k|s_i)\frac{\Delta N_i}{N_i} \quad (10)$$

where $N_i$, $N_T$ and $n_i(v_k)$ are counts related to all the training data: $N_i$ is the total number of observations in segment $s_i$, $N_T$ is the total number of observations, and $n_i(v_k)$ is the total number of observations type $v_k$ in segment $s_i$. Substituting (9) and (10) in (8),

$$\Delta H(V|S)=-\sum_{r=i}^{i+1}\Big[p(s_i)\sum_{k=1}^{M}log(ep(v_k|s_i))\frac{\Delta n_i(v_k)}{N_i} - p(s_i)\frac{\Delta N_i}{N_i}loge\Big] \quad (11)$$

Also, with $\delta$=1, the boundary movement only affects to one observation $v^*$,

$$v^*=\begin{cases} O_{l(i)} & \Delta l(i)=-1 \\ nothing & \Delta l(i)=0 \\ O_{l(i)+1} & \Delta l(i)=+1 \end{cases} \quad (12)$$

and,

$$\Delta n_i(v_k)=\begin{cases} \Delta N_i=1 & v_k=v^* \\ 0 & otherwise \end{cases} \quad (13)$$

Taking into account (12) and (13),

$$\Delta H(V|S)=-\sum_{r=i}^{i+1}\log(p(v^*|s_r))\Delta p(s_r) \quad (14)$$

For computational purposes, eq. (14) must be written as

$$B^{N_T\Delta H}=\begin{cases} p(O_{l(i)}|s_i)/p(O_{l(i)}|s_{i+1}) & \Delta l(i)=-1 \\ 1 & \Delta l(i)=0 \\ p(O_{l(i)+1}|s_i)/p(O_{l(i)+1}|s_{i+1}) & \Delta l(i)=-1 \end{cases} \quad (15)$$

where B is the log-base we are using. The decision about the boundary position can be made only with 2 divisions, and a decision about the minimum of expression (15) for finding $\Delta l^*(i)$.

## 5. Experimental results

The data were sampled at 8.091 KHz, and preemphasized with a preemphasis factor $\mu$=0.95. Hamming windows were applied to blocks of 256 samples, with an overlapping of 64 samples. Liftered Cepstrum was computed for each frame (with 10 cepstral coefficients and length 12 for the liftering window) and Delta Cepstrum is approximated by linear regression on a ±3 frames environment. Frame energy is normalized to the peak of energy in the word and expressed in the dB scale. Delta Energy is computed from the normalized dB-scaled values of Energy. Finally, an average of all of these parameters is performed every other consecutive frames to compose the feature vectors. The final result is equivalent to have 256-samples frames overlapped 128 samples.

The utterances were coded with a 128-centroid codebook in all the experiences, using the MWDM distance measure [3]. We used one model per word and 7 states per model.

The vocabulary consists of the ten Spanish digits and the Spanish words {CUERPO, HOMBRO, CODO, MUNECA, MANO, DEDOS}, thought for controlling each motor of a Robot.

The database consists of 40 speakers and 3 utterances per speaker and per word (1920 words), and it was recorded under the normal conditions of work rooms, so certain level of noise, such as the computer

noise, is included. We develop our experiments in speaker-independent mode, using the leaving-one-out technique [4], consisting on training the system with all the training speakers, leaving one out for recognition, and doing this for all the speakers. In our case, we leave 8 speakers out (4 female and 4 male), so 5 trainings and recognitions (T&R) are performed. All the experimental results are the average obtained in each T&R.

The experiments consist of trying the following segmentation methods for training or for Baum-Welch initialization:

SL+VT:
Linear segmentation followed by Viterbi reestimation.

SENT1:
Entropic segmentation computing sum (5) for entropy increment determination.

SENT2:
Entropic segmentation using eq. (15) for entropy increment determination.

Table 1 shows the error rate and computation time results using the above mentioned segmentations for training. Table 2 is the same, but using the segmentations only as initialization procedures for a Baum-Welch reestimation.

| | ERROR RATE(%) | TIME(seg) |
|---|---|---|
| SL+VT | 5.31 | 434.6 |
| SENT1 | 5.26 | 565.4 |
| SENT2 | 5.41 | 132.4 |

*Table 1.- Error rate and time results using SL+VT, SENT1 and SENT2 for training.*

| | ER(%) | BW(seg) | TOTAL(seg) |
|---|---|---|---|
| SL+VT+BW | 5.05 | 171.1 | 605.7 |
| SENT1+BW | 4.89 | 233.7 | 799.1 |
| SENT2+BW | 5.05 | 244.6 | 377.0 |

*Table 2.- Error rate and time results using SL+VT, SENT1 and SENT2 for Baum-Welch initialization.*

From table 1, it is clear that the three methods provide similar error rates, but SENT2 is computationally more efficient (more than 3 times SL+VT). Table 2 confirms the above results when a Baum-Welch reestimation is added to the training process.

## 6. Summary

We have presented a new segmentation method based on the maximization of the cohesion between segments and observations, according to the Entropic Cohesion Measure. We have proved that that criterion is equivalent to the minimization of the conditioned entropy between segments and observations. The proposed segmentation algorithm is based on the above criterion. A serious handicap for this algorithm is the high computational cost. We have proposed a solution for that problem, consisting of calculating only entropy increments by a differential approximation.

In the experimental results section, the segmentation algorithm based on entropy increment calculation has shown to be as accurate as a Viterbi reestimation, but meaningfully better in computational cost. We think that this result can be useful for parameter estimation problems in which a training data segmentation is needed (like in continuous HMM training).

## REFERENCES


[1] L.R. Rabiner: "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition". *Proceedings of the IEEE*, vol. 77, no. 2, Feb 1989.

[2] S. Guiasu: "Information Theory with Applications", pp. 352, *McGraw-Hill*, 1977.

[3] A.M. Peinado, P. Ramesh, D.B. Roe, "On the use of energy information for speech recognition using HMM". *Proceedings of EUSIPCO-90*, vol. 2, pp. 1243-1246, Sept. 1990.

[4] R.O. Duda, P.E. Hart: "Pattern classification and scene analysis". *John Wiley & Sons*, pp. 75-76, 1973.