Effective Calculation of Optimal Segmentation of Speech Data for VQ-Based Hidden Markov Models

Boris Dimitrov Andreev and Assoc.Prof.Ph.D.Jordan Nikolov Kolev

Department of Electronics
Technical University of Varna

Abstract

This paper is focused on the effective computation of an optimal automatic segmentation procedure. The latter can be applied in speaker-independent isolated word recognition using Hidden Markov Models. First we discuss the segmentation scheme based on additional information from the vector quantization stage. Then an effective algorithm for iterative calculation of the segment boundaries is presented. The algorithms are tested on a speaker-independent Bulgarian data base and results are reported.

I. Introduction

During the last few years vector quantization (VQ) and hidden Markov models (HMMs) have been gaining increasing popularity in speech recognition systems over alternative approaches like dynamic time warping (DTW) and artificial neural networks (ANN) [1,2,3]. Training of HMMs is essentially a search for an optimal set of coefficients calculated on the basis of the training data. Starting from some initial parameters, an iterative optimization procedure can be applied using the Baum-Welch or the Viterbi reestimation algorithm. This procedure leads to a set of parameters which yield a local maximum of the probability P(O|S). In general, however, a number of different local maxima exists and the final result depends on the proper choice of the initial parameter values. Different algorithms have been developed for intelligent selection of this starting set of coefficients [4-7]. In [4] an alternative method is put forward which uses elaborate statistical analysis from the vector quantization stage to calculate initial model parameters, reflecting the global maximum of the probability function. The authors of the current research have aimed at proposing an effective algorithm for computation of the segment boundaries and obtaining some experimental results in support of the method discussed.

II. Training of HMMs and segmentation of the training sequences

In this section the segmentation procedure from [4] and its place in the overall HMM training process are briefly explained. Applying vector quantization to feature vectors results in a loss of information [8]. The representation of a feature vector can be improved by using more information from the VQ processing: defining the closest L codebook vectors instead of only the nearest one to the analyzed vector. This supplemental information can be used for optimal segmentation of the training utterances resulting in a better initialization and optimized HMMs.

We consider to have a sequence of T speech frames $\{o_1, o_2, ..., o_T\}$ with corresponding descriptions (coefficient vectors: spectral, LPC, cepstral, etc.) $\{O_1, O_2, ..., O_T\}$. This input sequence must be divided into N consecutive segments (N<T), where N is the number of states in the Markov model, and then the frequencies of the codebook entries in the individual segments and the lengths of the segments are used to calculate the initial values of the probability coefficients a_{ij} and $b_j(k)$ for a first order HMM.

The segmentation of an utterance with T feature vectors (T frames) is based on the definition of a cost function C (named total distortion [4]) for a given set of segment boundaries $k_0, k_1, ..., k_i, ..., k_N$, with $k_0=0$ and $k_N=T$. For each individual segment an intrasegment distortion is introduced which reflects the variations among the vectors within the segment. The optimal boundaries are chosen among all possible sets so that the total distortion C is minimized. The analysis results prove to better significantly if an elaborated vector distance measure is used, based on the quantization information. Let $\pi_t = \{\pi_t(1), \pi_t(2), ..., \pi_t(M)\}$ be the resulting order from the VQ of a given feature vector y_t, where $\pi_t(1)$ is the index of the closest codebook entry, $\pi_t(M)$ – of the most distant and M is the codebook size. Using these designations the distance d(s,t) between two feature vectors y_s and y_t is defined as

$$\mathbf{d(s,t)} = \sum_{\lambda=1}^{L} \omega_\lambda \cdot \pi_s^{-1}(\pi_t(\lambda)) \quad , \quad L \geq M \quad , \tag{1}$$

where π^{-1} denotes inverse permutation with $\pi_t^{-1}(\pi_t) = \{1, 2, ..., M\}$, ω_λ is a set of weighting factors and L is the number of positions that are considered in the algorithm (i.e. all codebook entries are sorted in a descending order according to their proximity to the analyzed feature vector and only the closest L are considered in the computation of d(s,t)). Without the weights ω_λ, d(s,t) would not take into account the different orders of the first L entries. In order to avoid the time and memory consuming sorting of all codebook indices only a limited number of L_{max} positions can be used which leads to $\pi_s^{-1} = \{\pi_s^{-1}(1), ..., L_{max}, L_{max}, ...\}$. The local distortion $D(k_{i-1}, k_i)$ in the i^{-th} segment $S_i=[k_{i-1}, k_i]$ is given by a summation over the mutual distances between all vectors in the segment,

$$\mathbf{D(k_{i-1}, k_i)} = \sum_{s,t \in S_i} \mathbf{d(s,t)} \quad . \tag{2}$$

Finally, the total distortion (3) is calculated as a sum of the local distortions in all segments and the assignment is to reach such a set of the model parameters which minimizes function (3), i.e. $C=C_{min}$.

$$\mathbf{C} = \sum_{i=1}^{N} \mathbf{D(k_{i-1}, k_i)} \tag{3}$$

III. Effective computation algorithm and experimental results

The algorithm discussed is aimed at optimizing the computation of $D(k_{i-1}, k_i)$ and C. Calculations of all d(s,t) for $1 \leq s \leq (T-1)$, $(s+1) \leq t \leq T$, cannot be avoided so

they are separated in a preliminary procedure and for the case of a 40-frame sequence their number is

$$C_n^m = \frac{n!}{m!.(n-m)!} \qquad \Rightarrow C_{40}^2 = \frac{40!}{2!.38!} = 780 \qquad (4)$$

To avoid repeated calculations a restriction of s<t is imposed which does not change the essence of the algorithm and as can be seen the number of elementary vector distances is comparatively small. Let us view a simple division of a T-frame utterance into two segments, assuming the requested boundary to be **k**, i.e the first segment contains vectors 1÷k, and the second (k+1)÷T. The problem is to find the optimal **k** leading to the minimum of the total distortion C. To make the algorithm ultimately fast we aim to find iterative formulas both for $D(k_{i-1},k_i)$ and C. Let us introduce two auxiliary matrices

$sum_1(x,t)$ - contains the sum of all d(s,t) for $x \le s \le (t-1)$, s=var

$sum_2(s,y)$ - contains the sum of all d(s,t) for $(s+1) \le t \le y$, t=var .

Both matrices adhere to the general conditions $1 \le s \le (t-1)$, $(s+1) \le t \le T$ and the supplementary conditions $s \le x$, for sum_1 , and $t \le y$, for sum_2 . With regard to these designations the intrasegment distortions are defined as:

$$D_{1_k} = \sum_{s,t \in 1_1} d(s,t) \qquad 1 \le s \le (t-1),\ (s+1) \le t \le k$$

$$D_{2_k} = \sum_{s,t \in 1_2} d(s,t) \qquad (k+1) \le s \le (T-1),\ (s+1) \le t \le T \qquad (5)$$

$$C_k^{(2)} = D_{1_k} + D_{2_k}$$

By induction :

$$D_{1_{k+1}} = D_{1_k} + \sum_{s=1}^{k} d(s,k+1) = D_{1_k} + sum_1(1,k+1)$$

$$D_{2_{k+1}} = D_{2_k} - \sum_{t=k+2}^{T} d(k+1,t) = D_{2_k} - sum_2(k+1,T) \qquad (6)$$

$$C_{k+1}^{(2)} = C_k^{(2)} + \sum_{s=1}^{k} d(s,k+1) - \sum_{t=k+2}^{T} d(k+1,t) = C_k^{(2)} + sum_1(1,k+1) - sum_2(k+1,T)$$

Using the obtained inductive formulas all possible $C_k^{(2)}$ can be calculated and a boundary **k** defined, where $C=C_{min}$. Analogous reasoning may be applied when working out the formulas for a 3-segment division, where the main difference is the variation of both boundaries k_1 and k_2 between the three segments. On figure 1 the surface $C(k_1,k_2)$ is shown, calculated for an example utterance of the word "chetiri". All conducted experiments confirm that the total distortion function C has an only minimum (without any local ones) which allows the development of a better algorithm aimed at a straightforward search of this minimum without calculating the entire surface. The advantages of such an approach become essential when a segmentation into an arbitrary number of segments is needed where the calculations and memory required increase in a geometrical progression with quotient N.

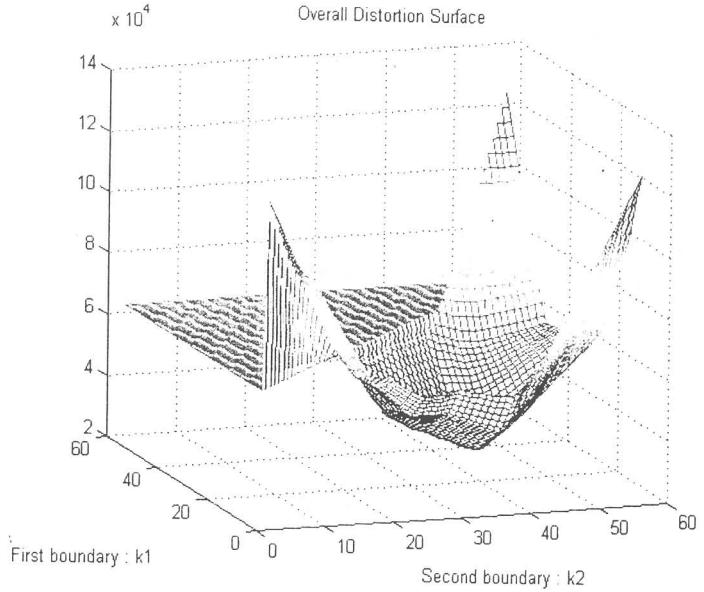

Fig. 1

The formulas worked out for $C_k^{(2)}$ and $C_{k1,k2}^{(3)}$ may be generalized for the division of a T-frame sequence into N segments with an optimal set of boundaries $\{k_0, k_1, k_2, ..., k_i, ..., k_{N-1}, k_N\}$. The boundaries $k_0=0$ and $k_N=T$ are only formally assumed and a given segment S_i starts with the $(k_{i-1}+1)^{-st}$ vector and ends up with the k_i^{-th}. The main idea here is that a variation in the intrasegment distortion D_i is possible only if one of the boundaries k_i or k_{i-1} is changed, and the unilateral alteration of k_i affects D_i and D_{i+1}, but does not influence the remaining D_j for $j \neq i$ and $j \neq (i+1)$. With regard to these designations the inductive formulas for the case when only k_i is changed can be expressed as

$$\mathbf{Di}_{k1,k2,...,k_i,...,k_{N-1}} = \mathbf{Di}_{k1,k2,...,k_{i-1},...,k_{N-1}} + \sum_{s=k_{i-1}+1}^{k_i-1} \mathbf{d}(s,k_i)$$

$$\mathbf{D(i+1)}_{k1,k2,...,k_i,...,k_{N-1}} = \mathbf{D(i+1)}_{k1,k2,...,k_{i-1},...,k_{N-1}} - \sum_{t=k_i+1}^{k_{i+1}} \mathbf{d}(k_i,t)$$

$$\mathbf{Dj}_{k1,k2,...,k_i,...,k_{N-1}} = \mathbf{Dj}_{k1,k2,...,k_{i-1},...,k_{N-1}} \qquad j \neq i,\ j \neq (i+1) \qquad (7)$$

$$C^{(N)}_{k_1,k_2,...,k_i,...,k_{N-1}} = C^{(N)}_{k_1,k_2,...,k_{i-1},...,k_{N-1}} + \sum_{s=k_{i-1}+1}^{k_i-1} \mathbf{d}(s,k_i) - \sum_{t=k_i+1}^{k_{i+1}} \mathbf{d}(k_i,t) =$$

$$= C^{(N)}_{k_1,k_2,...,k_{i-1},...,k_{N-1}} + \mathbf{sum}_1(k_{i-1}+1,k_i) - \mathbf{sum}_2(k_i,k_{i+1})$$

Generally, all above calculations may be expressed by an all-round computation of the N-dimensional surface C. In practice, however, the number of operations and memory required grows intolerably high. It becomes clear that the algorithm proposed is impractical in this form for more than 5 segments which instigated a new research for a faster approach directly leading to the global minimum of the surface C. Profound analysis of experimental results from segmentations into 2,3,4 and 5 segments led to the conclusion that on each step (adding a new segment) the boundaries of all segments are kept unchanged, except for the one that has had the greatest intrasegment distortion and optionally of one of its neighbors.

On this basis a new iterative algorithm was developed. It starts with a three-segment division of the input sequence and at each step a new segment is added, adhering to the minimum of the total distortion C. In the beginning of each iteration the highest intrasegment distortion D_i belonging to the segment S_i is defined. The analysis is conducted with two 3-segment divisions: one for the segments S_{i-1},S_i (within the boundaries k_{i-2},k_i) and the other for $S_i,S_{i+1}(k_{i-1},k_{i+1})$. The final iteration decision is assumed to be the one (of the two) which yield to a lower total distortion C. The speed of this algorithm is much improved by the difference in the computational costs of a N-segment and two 3-segment divisions for N>3.

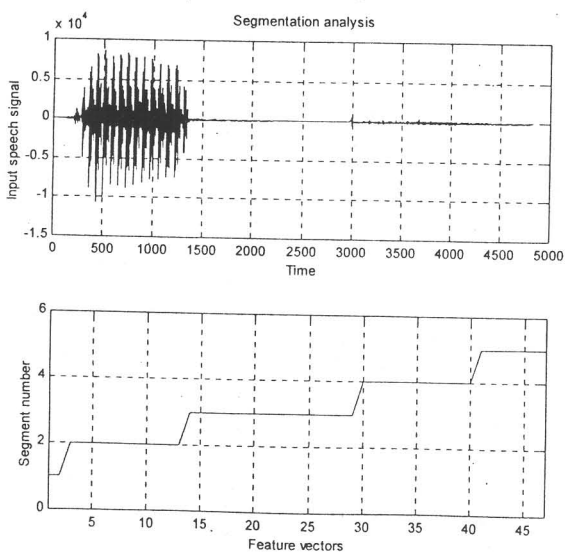

Fig. 2

On figure 2 results obtained through the new algorithm for a 5-segment division of the word "pet" are shown. Actually, the derived algorithm proves to give, though rarely, better but different results from the global minimum of the surface C.

So it turns out that the best matching of speech and segments is obtained not through a search of the very minimum of the total distortion but of the most heterogeneous of all current segments and a proper split as described above.

Utilizing the discussed algorithm a speaker-independent speech recognition system based on VQ and hidden Markov models was trained. The recognition error for a set of 10 isolated words is in the limits of 5-10%, where the lower recognition rates are due mainly to flaws in the speech-pause classification procedure. In the case of a manual elimination of the pauses the system error drops down to below 5%.

IV. Conclusion

A computational algorithm is put forward, based on the properties of mathematical induction and representing an alternative approach for calculation of the segment boundaries in the training utterances. The experiments conducted show a remarkable increase in the recognition rate. One of the main advantages of the method is the possibility for automated training of hidden Markov models. It should be noted that the obtained initialized HMMs are optimized in accordance with the maximum likelihood criterion for the training data set and require no further processing. Though research was conducted for the purposes of speech recognition, the algorithms discussed may be applied in any HMM application where optimal segmentation is needed. It may be concluded that the experimental results confirm the effectiveness of the method [4] and lay the ground for new research for additional improvements in the algorithms proposed.

Bibliography

1. L.Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proceedings of the IEEE, '2, February 1989, pp.257-286
2. L.Rabiner and B.-H.Juang, "An Introduction to Hidden Markov Models", IEEE Acoustics, Speech and Signal Processing Magazine, vol.3, '1, 1986, pp.4-16
3. S.Levinson, "Structural Methods for Automatic Speech Recognition", Proceedings of the IEEE, '11, November 1985
4. M.Falkhausen, S.Euler and D.Wolf, "Improved Training and Recognition Algorithms with VQ-Based Hidden Markov Models", IEEE-ICASSP 1990, vol.1, pp.549-552
5. T.Svendsen and F.Soong, "On the Automatic Segmentation of Speech Signals", IEEE-ICASSP 1987, vol.1, pp.77-80
6. J.Bridle, "A Method for Segmenting Acoustic Patterns, with Application to Automatic Speech Recognition", IEEE-ICASSP 1997, vol.1, pp.656-659
7. B.-H.Juang and L.Rabiner, "The Segmental K-Means Algorithm for Estimating Parameters of HMMs", IEEE Trans. on Acoustics, Speech and Signal Processing, vol.38, '9, September 1990, pp.1639-1641
8. J.Makhoul, S.Roucos and H.Gish, "Vector Quantization in Speech Coding", Proceeding of the IEEE, November 1985, pp.1551-1584