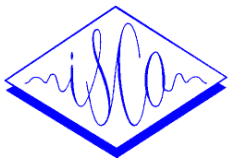

First International Conference on
Spoken Language Processing (ICSLP 90)
Kobe, Japan
November 18-22, 1990

AUTOMATIC ALIGNMENT OF PHONEMIC LABELS WITH CONTINUOUS SPEECH

Torbjørn Svendsen and Knut Kvale

Div. of Telecommunications
Dept. of Electrical Engineering and Computer Science
The Norwegian Institute of Technology
N-7034 Trondheim, NORWAY

ABSTRACT

Annotation of speech waveforms with phonemic labels and markers defining the position of the labels within the waveform is desirable for many purposes. In this paper we propose a method for automatic label alignment which can ease the task. The algorithm contists of two stages. The first stage segment the speech waveform into segments which contain acoustically similar speech frames. The second stage use the acoustic segment boundaries as anchor points in a phoneme based HMM type segmentation.

1 INTRODUCTION

For several purposes, segmentation of continuous speech into linguistically defined segments is desirable in speech processing. Speech waveforms annotated with phonemic labels and markers defining the beginning and end of the phoneme is needed e.g., for training phoneme based speech recognizers. Annotated waveforms are also useful for diagnostic purposes when optimizing speech recognizers as well as for excising prototypes for speech synthesis purposes. Our motivation for studying this problem was our participation in a European speech research project (ESPRIT-SAM) where there was a need for segmentation and labelling of a large, multilingual database.

The problem can be formulated as follows: Given the digitized speech waveform of a passage of continuous speech and a string of broad phonemic labels which correspond to the phonemic content of the passage, the task is to time-align the phoneme string with the speech waveform, explicitly defining the beginning and end sample of each phoneme.

The labelling of the speech can be done manually, but this has two major drawbacks: i) The process is both laborious and tedious requiring, e.g., extensive spectrogram reading and listening. ii) Due to the lack of an objective criterion, manual procedures unavoidably will exhibit some inconsistencies. Defining difference in labelling when phoneme boundaries differ by more than 10 ms, a recent study[1] reported that 28% of the phoneme boundaries differed when cross-comparing two different labelers and 24% of the boundary positions differed when comparing two segmentations made by the same person.

Within the ESPRIT-SAM project recordings are presently being done of a large, multilingual database. For each language, approximately 4 hours of speech will be recorded using 60 speakers per language. It is desirable to annotate the speech with phonemic labels explicitly marking the beginning and end of each phoneme. It is estimated that if this were to be done manually, 20-60 minutes would be required to transcribe one minute of speech and 600-2000 minutes would be required to manually align the phoneme labels with the speech waveform. Based on 40 working hours per week it would then take a minimum of 2 weeks to do the transcriptions and 60 weeks to do the label alignment for each language. Obviously, if the label alignment could be totally or partially done by automatic means, the workload could be drastically reduced.

Several methods for automatic label alignment have been proposed in the literature (e.g.,[2],[3],[4],[5],[6]). Some of the methods are entirely based on signal processing/statistics ([5],[6]), others are based on a two-level processing of the speech signal information where the first level performs an acoustic segmentation based on signal processing techniques and the second level applies knowledge based processing to obtain the final alignment ([2],[4]). In our case, we have not deemed it viable to resort to acoustic-phonetic knowledge, mainly due to the multilingual nature of the problem. One main goal was that the algorithm should be minimally dependent upon language and that once the language dependent characteristics are (automatically) learnt by the algorithm no further training should be required.

Several researchers have utilized the remarkable segmentation properties of Hidden Markov Models when training phoneme based speech recognizers. A typical approach has been to use a limited set of manually segmented speech to build initial models. These models are then used to perform segmentation of additional training material by Viterbi decoding of the optimal state sequence. The HMMs can then be improved by including the additional training material in the model estimation. For our purposes, the idea of using HMMs for automatic label alignment is certainly an appealing one. However, the available training material is very limited (some phonemes only occur 1-3 times), and it was foreseen that additional processing was required.

The basic idea of the present approach is founded in the observation that phoneme transitions as labeled manually tend to occur at instances of high acoustic variability. Secondly, not all acoustic transitions imply a phoneme boundary (e.g., plosives which contain two acoustic events, the pause and the burst). Our approach is thus based on a two-step procedure. In the first step, the speech signal is segmented into acoustically similar segments. These segment boundaries indicate the instances of acoustic change. Since some phonemes contain more than one acoustic event, the number of acoustic segments exceed the number of phonemes in the utterance. In the second step, HMM techniques are used in conjunction with the possible phoneme boundaries obtained in the first step to yield the final segmentation.

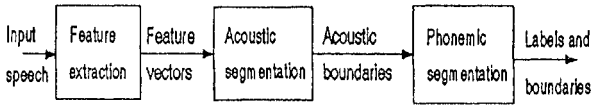

Figure 1. System overview

2 ACOUSTIC SEGMENTATION

As mentioned above, several of the phonemic label alignment methods use a first step for acoustic segmentation. In [2], each input speech frame is classified as belonging to a broad phonetic class. Spurious classifications are removed by a smoothing median filter, thereby making the operation a segmentation, albeit more phonetic than acoustic in character. In [5] a similar acoustic classification is performed by a phonotopic map. The SUMMIT system [4] uses an acoustic segmentation procedure which classifies acoustically similar consecutive speech frames as belonging to the same segment [3]. Finally, in [6] a statistical approach is taken to detect acoustic changes implying that a phonetic event has occurred.

Our approach is similar in spirit to the approach taken by Glass and Zue [3] and is based on a concept called constrained clustering vector quantization [7]. Assume that the acoustic segmentation algorithm is presented with a sequence of speech frames $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$. The task of the acoustic segmentation algorithm is to segment this utterance into m consecutive segments with boundaries given by $\{b_0, b_1, \ldots, b_m\}$, where we have defined $b_0 = 0$ and $b_m = T$. The optimal segmentation will then be found as the segmentation that maximizes the acoustic similarity of the speech frames within the segments. We thus wish to find the set of boundaries, $\{b_i\}$, which minimize

$$\sum_{i=0}^{m-1} \sum_{n=b_i+1}^{b_{i+1}} d(\mathbf{X}_n, \mathbf{c}_i) = \sum_{i=0}^{m-1} D_s(b_i+1, b_{i+1}) \qquad (1)$$

where $\mathbf{X}_n$ is the spectral representation of speech frame $\mathbf{x}_n$, $\mathbf{c}_i$ is the generalized centroid of the i^{th} segment consisting of the spectral sequence $\{\mathbf{X}_{b_i+1}, \mathbf{X}_{b_i+2}, \ldots, \mathbf{X}_{b_{i+1}}\}$ for a specific distortion measure $d(\mathbf{x}, \mathbf{y})$ and $D_s(b_i+1, b_{i+1})$ is the corresponding segment distortion.

In order to find the optimal acoustic segmentation we must evaluate (1) for all possible combinations of the segment boundaries. This can be efficiently be performed by the use of dynamic programming. Denoting the minimum accumulated distortion obtained by segmenting the spectral sequence $\{\mathbf{X}_{b_i+1}, \mathbf{X}_{b_i+2}, \ldots, \mathbf{X}_{b_{i+1}}\}$ in $i+1$ segments as $D(i+1, b_{i+1})$, the dynamic programming problem can be formulated as finding the minimum of $D(i+1, b_{i+1})$ as

$$D(i+1, b_{i+1}) = \min_{b_{i+1}} \{D(i, b_i) + D_s(b_i+1, b_{i+1})\} \qquad (2)$$

for all possible b_{i+1}. This can be done by first computing a distortion matrix,

$$\mathbf{D}_s = \{D_s(i,j)\} \quad ; \quad 1 \le i \le m \quad ; \quad 1 \le j \le T \qquad (3)$$

and defining

$$D(1, b_1) = D_s(1, b_1) \quad ; \quad 1 \le b_1 \le T \qquad (4)$$

whereafter the DP search is performed and the minimum distortion for a m-level segmentation is found as $D(m,T)$. The optimal segmentation is finally found by backtracking the DP grid along the optimal path.

In our case, the length of the input speech utterance can be rather long (more than 1000 frames). This means that the distortion matrix, $\mathbf{D}_s$, will be large and that the computation will be extensive. The storage size is manageable as the matrix values can be computed on the fly, but the backpointer array which gives us the optimal segmentation can not easily be reduced. A requirement for the segmentation algorithm was that it could be implemented on an IBM compatible PC under DOS, restricting the program size to a maximum of 600 kBytes. Thus, it was necessary to find simplifications to the algorithm which reduced the program size without compromising performance too severely.

Two modifications have been implemented. The first simply restricts the maximum length of an acoustic segment to a fixed value. Typically, the maximum segment length has been set to 250ms (50 frames).

The second modification restricts the number of paths to retain as possible optimal paths. In an optimal implementation, a back pointer should keep track of the optimal m-level segmentation for frames 1 through n. In practice it turns out that in most instances, successive ending frames (e.g. $\ldots, x_{n-1}, x_n, x_{n+1}, \ldots$) tend to yield the same acoustic segmentation (apart from the ending frame). Thus, these successive frames have the same back pointer value. In the current implementation, the back pointer values are only recorded when the value changes, and a maximum of 50 back pointer values are retained at each segmentation level. For most utterances, the degradation introduced by this restriction is non-existent or minimal. For longer utterances, the degradation is more noticeable, the major effect being a reduction in the number of manually determined segment boundaries detected within the maximum resolution of the system (5 ms, the frame shift). However, the deterioration when allowing a larger deviation from the manual segmentation is minimal.

3 PHONEMIC SEGMENTATION

The output from the acoustic segmentation is a set of possible frame boundaries $\{b_0, b_1, \ldots, b_m\}$. Using a suitable degree of oversegmentation, the majority of the "true" segment boundaries are included in this set. The task of the phonemic segmentation is then to merge consecutive acoustic segments into phonemic segments so that the final phoneme alignment is obtained.

From a set of manually segmented speech data, Hidden Markov Models for all phonemes occurring in a language were estimated. The phoneme HMMs were 3-state models where a jump from state 1 to state 3 was allowed. The models were continuous density Gaussian with one mixture using a diagonal covariance matrix. This choice was made due to the limited amount of training data available.

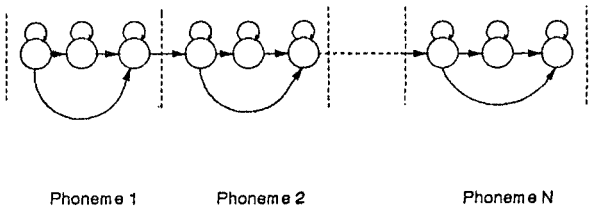

Figure 2. Phonemic segmentation

When doing the final alignment, a constrained Viterbi decoding was performed. The phoneme HMMs were concatenated according to the input string of phoneme symbols in order to obtain a HMM for the entire utterance. The boundaries obtained through the acoustic segmentation were also input to the algorithm. The standard Viterbi decoding was then performed with the following modification in the recursion:

$$\phi_t(i,m) = \begin{cases} \max[\phi_{t-1}(1,m)a_{11}^{(m)}, \phi_{t-1}(3,m-1)]b_i^{(m)}(\mathbf{X}_t) & ; \quad i = 1 \\ \max_{1 \le j \le I}[\phi_{t-1}(j,m)a_{ji}^{(m)}]b_i^{(m)}(\mathbf{X}_t) & ; \quad i = 2,3 \end{cases} \qquad (5)$$

where $\phi_t(i,m)$ is the accumulated probability of being in state i of phoneme model m at time t, $a_{ij}^{(m)}$ is the probability of making a transition from state i to state j for model m and $b_i^{(m)}(\mathbf{X})$ is the probability of observing the spectral vector X in state i of model m. The modification thus only allows jumps from one phoneme model to the next at the time instances specified by the acoustic segment boundaries, $\{b_0, b_1, \ldots, b_m\}$ and jumps are only allowed from the last state in the preceding phoneme model to the first state in the following model.

4 EXPERIMENTS

The experiments to be presented were performed on the EUROM0 corpus compiled by the ESPRIT SAM project. The portions of this corpus used for the experiments were a continuous passage of a duration of approximately 2 minutes containing an average of approximately 1.200 phonemes. The passage was read by four speakers per language, two male and two female speakers. The languages tested so far were Italian and British English. The passages are digitized using a sampling rate of 16.02 kHz and are stored on a CD-ROM.

4.1 Test preliminaries

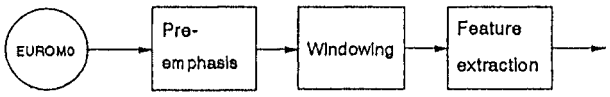

Figure 3. Pre-processing

The pre-processing of the speech is shown in fig. 3. The digitized speech is read from the CD-ROM and pre-emphasized with a first order filter with transfer function $H(z) = 1 - 0.95z^{-1}$. The British English recordings had a severe DC bias which was adjusted for prior to processing. A 15 ms Hamming window is applied every 5 ms and a 15-order LPC analysis is then performed on the windowed data. The LPC coefficients are then converted to 18 cepstral coefficients. The peak frame energy for each utterance is found and the normalized log energy and differential energy are included as the final parameters creating a 20-dimensional parameter vector for each frame.

The continuous passages have been manually segmented and labelled by experts native in the specific languages. The manual labelling is in the following used as the "true" segmentation and the coincidence between the automatic segmentation and the manual segmentation is used as a figure of merit for the automatic segmentation algorithm. The manual segmentation will unavoidably exhibit some inconsistencies, some phoneme boundaries have been more or less arbitrarily set since there are no definite acoustic cues showing the transition point between phonemes and there is a possibility that some boundaries are outright incorrect. We do not, however, have any better indication on what the "true" phoneme boundaries are than what human experts, to the best of their abilities, can determine.

The system does not perform any explicit endpointing. The accompanying label files are searched for the SAMPA (the SAM phonetic alphabet) symbols indicating silence and the speech samples between pauses are taken to be one utterance. The phoneme string between two pauses are also read from the label files and the task for the algorithm is to align the labels with the speech signal indicating beginning and end of each phoneme.

In the acoustic segmentation, the energy parameters are not used. The number of phonemes in the utterance is derived from the string of associated phoneme label. The number of acoustic segments is then set to be 2.5 times the actual number of phonemes. Setting the oversegmentation degree this high provides that 98% of the "true" segment boundaries have been found within a margin of ±20 ms. The acoustic segmentation does not require any training, all is based upon acoustic similarity within the utterance being analyzed.

The parameters for the HMM-based phonemic segmentation need to be estimated using a suitable set of labelled training data. Due to the lack of labelled training material in most languages, it was chosen to use the EUROM0 corpus for training as well as for testing. This was done in the following way:

-In order to test for speaker dependencies, the HMM models were trained using 3 speakers and tested using the remaining speaker. Approximately 3500 and 1200 phonemes were included in the training and test sequence respectively.

-In order to test text dependencies, the speech corpus was split in two (approximate) halves. One half (initial or final) uttered by all four speakers was used for training while the other (disjoint) half was used for testing.

The above arrangement will not be able to provide a valid evaluation of the algorithm. Ideally we should have tested using a different text and different speakers. However, the above test setup was the best we could manage with the available speech corpus.

4.2 Results

In Fig. 4, an example of the performance of the proposed algorithm is shown. The speech utterance was "In language", uttered by a female speaker. In the upper window, the acoustic segment boundaries are shown and in the wide, middle window, the speech waveform and the final phoneme boundaries are depicted. In the bottom window, the bars show the location of the manually placed phoneme boundaries. As can be seen, there is a good match between the automatic and the manual segmentation. The largest deviations are in the transitions between /n/ and /l/ and between /{/ and /N/. These phoneme boundaries are also difficult to determine manually.

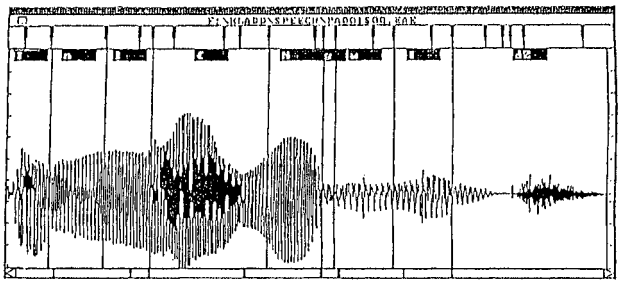

Figure 4. Automatic segmentation of the utterance "In language".

Performing a complete test of the algorithm, testing sequentially for each speaker (4 tests/language) and text (2 tests/language) dependence, the gross results shown in Table 1 and Table 2 were obtained. In the tables, the coincidence rates are given in percent as a function of the allowed deviation from the manually determined phoneme boundaries.

Table 1. Test for speaker dependence

Inside/ outside training	Language	Allowed deviation			
		< 5 ms	< 10 ms	< 20 ms	< 25 ms
Inside	English	54.6%	74.4%	88.1%	91.3%
	Italian	43.8%	63.2%	81.5%	85.6%
Outside	English	50.8%	68.9%	82.3%	86.0%
	Italian	38.6%	55.2%	71.7%	75.5%

The test for speaker dependency show that the difference in performance inside and outside the training corpus is rather large. This reflects the fact that the training material available is too small to properly train the model parameters. The continuous passage is not phonetically balanced and thus some phonemes are have relatively few occurrences (some occur only once). An analysis of the performance for the individual speakers show that there are indeed sheep and goats; the difference between the coincidence rates inside and outside the training corpus vary from as little as 2% to as much as 12-14%.

The speaker sensitivity is clearly higher in the Italian corpus. The results for Italian are also clearly poorer than for English when the speaker is included in the training material.

Table 2. Test for text dependence

Inside/ outside training	Language	Allowed deviation,			
		< 5 ms	< 10 ms	< 20 ms	< 25 ms
Inside	English	54.5%	74.3%	88.3%	91.6%
	Italian	43.6%	63.9%	81.2%	85.4%
Outside	English	51.7%	70.6%	84.6%	88.3%
	Italian	41.3%	59.9%	77.9%	82.2%

The dependency on which portion of the spoken passage to use for training is much smaller than the speaker dependency. Obviously, with a passage consisting of ~1200 phonemes, not all phoneme combinations can be included and thus quite a few coarticulatory phenomena are not present in the training sequence. The difference between the performance inside and outside the training corpus is uniformly on the order of 3-4%. A closer analysis of the individual results confirm the gross results; there is no major difference in performance whether the initial or the final portion of the passage is used for training and the deterioration when going outside the training corpus is at the same level for both languages.

We have also broken down the gross results to investigate the performance of the algorithm for finding the boundaries between phonemes belonging to different macro-classes (plosives, affricatives, fricatives, vowels, nasals, glides and liquids). The following discussion apply to results obtained for speakers outside the training corpus.

We will take a look at some phoneme-class combinations that constitute a major part of the speech corpus. The combinations plosive/vowel, fricative/vowel, vowel/fricative, fricative/plosive and affricate/vowel make up 36% of the phoneme combinations in the English and 31% of the Italian test corpus. The algorithm performs considerably better than the gross average on these phoneme-class combinations. For the combination vowel/plosive which constitute 10.5% of the English corpus and 11.5% of the Italian corpus, the algorithm performs slightly better than average. This is also reasonable since these phoneme classes tend to exhibit clearly different spectra. On the other end of the scale, the algorithm performs significantly worse than average on the combinations vowel/glide, glide/vowel and vowel/liquid which make up 5% of the English and 17% of the Italian test corpus. On the combination vowel/nasal which constitute 12% of the English and 9% of the Italian test corpus, the algorithm performs slightly worse than the average. Again, the result is not really surprising considering the phoneme classes involved.

In general, the results for the different phoneme-class combinations correspond very well across the two languages. The results for Italian are inferior to the results for English also on the phoneme-class level. This is amplified in the overall results by the fact that Italian has a greater number of the troublesome phoneme-class combinations.

5 CONCLUSIONS

We have presented an algorithm for automatically aligning broad phonemic labels with a speech waveform. The results obtained are promising for the utilization of automatic algorithms as an aid in the time-consuming task of labelling and segmentation.

FINAL COMMENTS

The proposed method for automatic label alignment is one of three proposals presently being evaluated by the ESPRIT SAM project. The results and interpretations presented in this paper are those of the authors and do not necessarily represent the views of the forthcoming formal evaluation being performed by SAM.

This work has been sponsored by The Royal Norwegian Council for Scientific and Industrial Research.

REFERENCES:

[1] A. Van Erp, L. Boves: "Manual Segmentation and Labelling of Speech" Proceedings of Speech'88, pp. 1131-1138, Edinburgh, Aug. 1988

[2] H. C. Leung, V. W. Zue: "A Procedure for Automatic Alignment of Phonetic Transcriptions with Continuous Speech", Proc. ICASSP-84, pp. 2.7..1-4, San Diego 1984

[3] J. R. Glass, V. W. Zue: "Multi-Level Acoustic Segmentation of Continuous Speech", Proc. ICASSP-88, pp. 429-432, New York 1988

[4] V. W. Zue et Al.: "Acoustic Segmentation and Phonetic Classification in the SUMMIT System", Proc. ICASSP-89, pp. 389-392, Glasgow 1989

[5] K. Torkkola: "Automatic Alignment of Speech with Phonetic Transcriptions in Real Time", Proc. ICASSP-88, pp. 611-614, New York, 1988

[6] R. André-Obrecht: "A New Statistical Approach for the Automatic Segmentation of Continuous Speech Signals", IEEE Trans. on ASSP, Vol. 36, No. 1, pp. 29-40, January 1988.

[7] T. Svendsen, F. K. Soong: "On the Automatic Segmentation of Speech", Proc. ICASSP'87, pp. 77-80, Dallas, April 1987.