# A Comparison of Different Approaches to Automatic Speech Segmentation

Kris Demuynck and Tom Laureys*

K.U.Leuven ESAT/PSI
Kasteelpark Arenberg 10
B-3001 Leuven, Belgium
{kris.demuynck,tom.laureys}@esat.kuleuven.ac.be
http://www.esat.kuleuven.ac.be/~spch

**Abstract.** We compare different methods for obtaining accurate speech segmentations starting from the corresponding orthography. The complete segmentation process can be decomposed into two basic steps. First, a phonetic transcription is automatically produced with the help of large vocabulary continuous speech recognition (LVCSR). Then, the phonetic information and the speech signal serve as input to a speech segmentation tool. We compare two automatic approaches to segmentation, based on the Viterbi and the Forward-Backward algorithm respectively. Further, we develop different techniques to cope with biases between automatic and manual segmentations. Experiments were performed to evaluate the generation of phonetic transcriptions as well as the different speech segmentation methods.

## 1 Introduction

In this paper we investigate the development of an accurate speech segmentation system for the Spoken Dutch Corpus project. Speech segmentations, on phoneme (e.g. TIMIT) or word level (e.g. Switchboard, CGN), have become a standard annotation in speech corpora. Corpus users can benefit from the fact that the segmentation couples the speech signal to the other annotation layers (orthography, phonetics) by means of time stamps, thus providing easy access to audio fragments in the corpus. In addition, for the speech technologist segmentations are indispensable for the initial training of acoustic ASR models, the development of TTS systems and speech research in general.

Some speech corpora only provide automatic segmentations, obviously requiring an accurate segmentation algorithm. In other corpora speech segmentations are checked manually. The latter case requires a high-quality automatic segmentation system as well, since a better base segmentation speeds up the manual verification procedure which is time-consuming and expensive.

Some segmentation systems are based on specific acoustic cues or features for the segmentation task [1,2,3] focusing for instance on transient behaviour

* This publication was supported by the project 'Spoken Dutch Corpus' (CGN-project), which is funded by the Flemish Government and the Netherlands Organisation for Scientific Research (NWO).

or specific differences between phoneme classes. Others use general features and acoustic modeling which are common in ASR [4,5]. The method proposed in this paper is of the latter type.

Most speech segmentation systems take as input both the speech signal and its phonetic transcription. This phonetic transcription might be produced manually, which is again a time-consuming and expensive process. As a result, manual phonetic transcriptions are not always available for speech corpora. Orthographic transcriptions, on the other hand, make up the corpus' base annotation. This is the reason why we propose a segmentation system starting from a phonetic transcription that is automatically generated on the basis of its orthography.

Consequently, the complete segmentation process is composed of two subtasks. First, a number of alternative phonetic transcriptions is produced on the basis of a given orthographic transcription and an automatic speech recognizer is used to select the acoustically best matching phonetic representation. Then, this single phonetic transcription serves as input to a segmentation system based on the Viterbi or the Forward-Backward algorithm.

## 2 From Orthography to Phonetics

The automatic conversion from an orthographic to a phonetic transcription takes two steps. First, several techniques are applied to produce a network of plausible pronunciation variants. In a second step, the single best matching phonetic transcription is selected. We performed the conversion on material from the Spoken Dutch Corpus, in which the orthographic annotation is enriched with codes to indicate certain spontaneous speech effects [6].

A full network of alternative phonetic transcriptions is generated on the basis of orthographic information. Lexicon lookup is a simple but efficient way to acquire phonetic word transcriptions. Yet, not every orthographic unit is a plain word. Some speech fragments contain sloppy speaking styles including broken-off words, mispronunciations and other spontaneous speech effects. Different techniques are introduced to handle these phenomena and a grapheme-to-phoneme system (g2p) was developed as a fall-back. We will first describe the g2p system. Then we focus on the other techniques and resources employed.

**g2p:** The g2p system is based on the Induction Decision Tree (ID3) mechanism [7] and was trained on the Flemish Fonilex pronunciation database (200K entries) [8]. Each phoneme is predicted based on a vector of 10 variables: the grapheme under consideration, a context of 4 left and 4 right graphemes and the last decoded phoneme (feedback). Phonetic transcriptions are generated from back to front so that the last decoded phoneme corresponds to the right neighbour which turned out to be most informative. We performed a ten-fold cross validation on Fonilex and achieved a 6.0% error rate on the word level.

**lexicon lookup:** Fonilex provides (multiple) phonetic transcriptions for most of the *standard* Flemish words. In addition, we developed rules to deal with non-covered compounds, derivations and inflections. At this early stage, 5376 *proper nouns* (often foreign) were manually transcribed. A new g2p convertor may be trained on these transcriptions to deal with future proper noun input. Lexicon lookup is also the first option for *foreign words*. We build upon the

COMLEX English database, the CELEX German database and the BRULEX French database. If a foreign word is part of more than one of these lexica, the different phonetic realizations are put in parallel since the orthography does not specify which foreign language was used.

**spontaneous speech effects:** For *broken-off words*, also with broken-off orthography, we first retrieve all lexicon words starting with the given orthographic string. Then, a grapheme-phoneme alignment is produced for the retrieved words which allows us to select the phoneme sequence(s) corresponding to the given orthography. *Mispronounced words* are fed to the g2p convertor. *Dialectical pronunciations*, orthographically represented by the standard Flemish word marked with a code, are dealt with by first selecting a phonetic transcription for the standard word. Next, we generate dialectical pronunciation variants for the word by means of context-dependent rewrite rules. Finally, *cross-word* phonological phenomena such as assimilation, degemination and inserted linking phonemes are treated by context-dependent rewrite rules as well.

The outcome of the above techniques is a compact pronunciation network [9]. To select the transcription matching best with the speech signal, all phonetic alternatives are acoustically scored (maximum likelihood) in a single pass (Viterbi) through our speech recognition system and the most probable one is retained. The phoneme models are statistically represented as three-state left-to-right Hidden Markov Models (HMMs).

## 3 Speech Segmentation: Viterbi vs. Forward-Backward

Once a phonetic transcription has been selected, automatic segmentation can proceed in the following way. First, sentence models are then generated by simply concatenating all relevant phoneme models. Next, the speech data are assigned (hard or soft, by respectively Viterbi or Forward-Backward) to the acoustic model of the complete phoneme sequence.

### 3.1 Viterbi Segmentation

The Viterbi algorithm returns the single best path through the model given the observed speech signal $x_1^T$ (the corresponding sequence of feature vectors):

$$s_i^T = \arg\max_{s_i^T \subset S} \prod_{i=1}^{T} f(x_i \mid s_i) p(s_i \mid s_{i-1}), \tag{1}$$

with $s_i^T$ a sequence of HMM states (one state for each time frame) which is consistent with the sentence model $S$, $T$ being the number of time frames. Thus, the Viterbi algorithm results in the segmentation which reaches maximum likelihood for the given feature vectors.

### 3.2 Forward-Backward Segmentation

The Viterbi algorithm only provides us with an approximation of the quantity that is really looked for. This is illustrated in figure 1. The Viterbi algorithm generates the boundary corresponding to (1), whereas the best boundary in a least squares sense matches with (2).

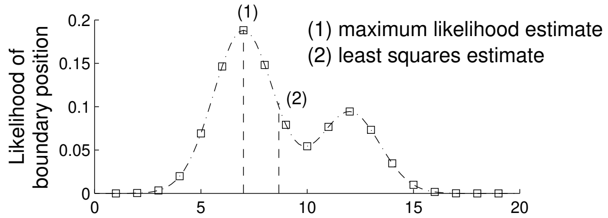


**Fig. 1.** Viterbi and Forward-Backward boundaries

To find the best possible estimate of the boundary in a least squares sense the probability function of each boundary must be calculated:

$$P(b|S, x_1^T) = \frac{f(x_1^b|S_l) f(x_{b+1}^T|S_r)}{f(x_1^T|S)}, \tag{2}$$

with

$$f(x_a^b|S_x) = \sum_{s_a^b \subset S_x} \prod_{i=a}^{b} f(x_i|s_i)^{1/\beta} p(s_i|s_{i-1})^{1/\beta}. \tag{3}$$

In the above equations, sentence $S$ is divided in part $S_l$ left and part $S_r$ right of the boundary of interest. The extra parameter $\beta$ compensates for the ill-matched assumption made by HMMs that the observations $x_i$ are independent. The optimal value for $\beta$ in our experiments was 10, but its exact value was not at all critical. The same compensation factor can be found in recognition systems [9] as well as in confidence scoring of recognized words [10] for balancing the contribution of acoustic and language model. The Forward-Backward algorithm allows for an efficient calculation of the density functions for all boundaries in a sentence. Given the probability density function of each boundary, the least squares estimate now equals:

$$E\{b\} = \sum_{b=1}^{T} P(b|S, x_1^T)\, b. \tag{4}$$

### 3.3 Post-Processing Techniques for Segmentation

A detailed comparison of automatic and corresponding manual segmentations revealed the occurrence of biases between the respective segmentations. These biases depend on the classes of the phonemes left and right of the boundary, and can be attributed to the fact that humans use different cues than HMMs for finding the boundary between consecutive phonemes [2]. For the transition to a vowel, for example, the average difference can be more than halved when compensating for these biases. An equally big improvement can be obtained for the transitions to noise. We discerned 9 phoneme classes in total and analyzed the biases on the boundary position between each pair of classes. Those biases in the automatic segmentations are removed in a post-processing step.

In a first approach, we shift the boundaries purely on the basis of the average biases. This simple technique is applicable to both Viterbi and Forward-Backward segmentations. A second method tries to compensate for the biases in a more advanced way by taking into account the confidence interval for the boundary. These confidence intervals are derived from the Forward-Backward method. Since the Forward-Backward algorithm calculates the probability density function for each boundary, we can regard the variance of this function as a confidence interval for the respective boundary:

$$\mathrm{Var}(b) = \sum_{b=1}^{T} p(b \mid S, x_1^T)(b - E\{b\})^2. \tag{5}$$

So we estimate the bias as a function on the boundary's confidence interval. This function is determined empirically with a polynomial fit on a train set. In section 4.3 we will discuss the results for both the simple and more advanced way of post-processing.

## 4 Experiments

### 4.1 Description

Experiments were performed on data taken from the Spoken Dutch Corpus. Three test sets were selected, representing different degrees of difficulty for the segmentation process. Test set 1 (50 speakers) accounts for the cleanest speech in the corpus, namely the read-aloud texts. It consists of 14176 words, resulting in 17976 boundaries since pauses exceeding 50 ms were also part of the segmentation. Broadcast material (documentaries, news shows, etc.) and public speeches belong to test set 2 (23 speakers, 7135 words, 9189 boundaries). They are harder to process than the read-aloud texts as background noise might be present and the speaker's style becomes more disfluent. Finally, test set 3 (11 speakers) consists of informal interviews, discussions and school lessons for a total of 27878 words and 36698 boundaries. They pose the hardest problem for the segmentation system as they are riddled with overlapping speech, dialectical pronunciations, etc.

For the experiments we used the LVCSR system developed by the ESAT-PSI speech group at the K.U.Leuven [9,11]. The acoustic models employed in the experiments were estimated on a separate database with 7 hours of dictated speech in Flemish.

### 4.2 Automatic Phonetic Transcription

We evaluated the automatic generation of phonetic transcription by counting the number of insertions, deletions and substitutions of the automatic transcription with respect to a hand-checked reference transcription. This reference transcription was produced by a trained phonetician who corrected a baseline transcription generated by a g2p system different from the one described in this paper. The results of the comparison are summarized in table 1. These results were obtained by using context-dependent models, which outperformed corresponding context-independent models for this task.

| **test set** | **ins** | **del** | **sub** | **total** |
|---|---|---|---|---|
| test set 1 | 0.77% | 1.27% | 2.95% | 4.99% |
| test set 2 | 1.15% | 1.59% | 3.41% | 6.15% |
| test set 3 | 1.82% | 2.18% | 4.26% | 8.26% |

**Table 1.** Deviations between automatic and manual phonetic transcription

A detailed analysis revealed three main causes for the deviations. First, certain infrequent assimilation rules were not included in our conversion system so that the corresponding pronunciation variants did not occur in the network. Second, the acoustic models were sometimes problematic because they impose a minimal duration constraint of 30 ms (causing schwa-deletion in particular) and because train and test conditions differ (especially for test sets 2 and 3). Third, not every deviation unambiguously corresponded to an error in the automatic transcription. For example, the automatic transcription typically incorporates more connected speech effects than its manual counterpart. This might be due to the fact that human transcribers, having to work at a considerable speed, sometimes overlook these phenomena not present in the base transcription they were offered. For example, especially schwa and linking phonemes were inserted in the automatic phonetic transcription. In Dutch schwa can be inserted in coda position in nonhomorganic coda position (e.g. /kAlm/ → /kAl@m/) [12]. Yet, this schwa-insertion is not part of the baseline phonetic word transcription provided by the g2p system. Similarly, schwa and syllable-final /n/ were often deleted in the automatic phonetic transcription. Again both phenomena are typical of Dutch connected speech.

### 4.3 Automatic Word Segmentation

The automatic segmentations were evaluated by counting the number of boundaries for which the deviation between automatic and manual segmentation exceeded thresholds of 35, 70 and 100 ms. Manual segmentation was performed by two persons and started from an automatic segmentation produced by the Viterbi algorithm (sect. 3.1). The persons were instructed to position boundaries so that each word would sound acoustically acceptable in isolation. Shared phonemes at the boundary (e.g. he is_sad) were split in the middle, except for shared plosives (e.g. stop_please), which were isolated altogether. Noticeable pauses (> 50 ms) were segmented in the same way as words, thus producing empty chunks.

| **test set** | **manual phon. trans. deviations exceeding** | | | **automatic phon. trans. deviations exceeding** | | |
|---|---|---|---|---|---|---|
| | **35ms** | **70ms** | **100ms** | **35ms** | **70ms** | **100ms** |
| **Viterbi** | | | | | | |
| test set 1 | 7.8% | 1.7% | 0.7% | 8.5% | 1.9% | 0.7% |
| test set 2 | 14.4% | 6.0% | 3.4% | 15.8% | 6.3% | 3.5% |
| test set 3 | 14.3% | 9.3% | 7.7% | 16.1% | 9.4% | 7.5% |
| **Viterbi post-processed** | | | | | | |
| test set 1 | 7.8% | 1.5% | 0.6% | 8.5% | 1.8% | 0.7% |
| test set 2 | 14.2% | 5.5% | 3.3% | 15.0% | 5.8% | 3.4% |
| test set 3 | 12.7% | 8.6% | 7.3% | 14.3% | 8.8% | 7.1% |
| **Forward-Backward** | | | | | | |
| test set 1 | 8.1% | 1.5% | 0.6% | 8.8% | 1.7% | 0.6% |
| test set 2 | 14.4% | 5.6% | 3.0% | 15.6% | 5.8% | 3.1% |
| test set 3 | 16.7% | 9.6% | 7.6% | 17.9% | 9.5% | 7.2% |
| **Forward-Backward post-processed** | | | | | | |
| test set 1 | 7.1% | 1.3% | 0.6% | 7.7% | 1.5% | 0.6% |
| test set 2 | 13.8% | 5.0% | 2.9% | 14.7% | 5.3% | 3.0% |
| test set 3 | 14.8% | 9.0% | 7.3% | 15.8% | 8.9% | 7.0% |

**Table 2.** Results: Viterbi vs. Forward-Backward

We performed experiments for both Viterbi and Forward-Backward segmentation, starting from a manual and automatic phonetic transcription. As can be seen from the post-processed results in table 2, the forward-backward method clearly outperforms the Viterbi approach on test set 1 and 2. The different behaviour on test set 3 is mainly due to the combined effect of using the Viterbi segmentation as a starting point for the manual verification process and the low quality of the material in test set 3 for which the human correctors quickly learned that only in few cases tangible improvements could be obtained by moving boundaries. This behaviour is reflected in the number of boundaries for which alternative positions were tried by the correctors: 37.1% and 51.7% for test set 1 and 2 versus only 32.7% for test set 3.

A detailed analysis of the results showed that the majority of the deviations in the automatic post-processed segmentations are transitions to and from noise and transitions to unvoiced plosives (45%, 11% and 15% of the remaining 35 ms errors respectively). Since these boundaries also show large variation between the corresponding manual segmentations of different correctors, we cannot expect an automatic system to give more consistent results.

Post-processing using confidence intervals showed no improvement and hence only the results for the simplest post-processing proposed in section 3.3 are given in table 2. The confidence intervals can be used to predict misplaced boundaries (e.g. more than 50% of the 75 ms deviations can be found by checking only the 10% boundaries with the largest predicted variance) but since the sign of the deviation (shift boundary to the left or right) cannot be predicted, no better boundary positions could be devised. However, the confidence intervals may be useful for other applications such as TTS systems for which the segments with reliable boundary positions now can be selected automatically.

Using the automatically derived phonetic transcriptions results in a limited degradation in the accuracy of the boundary positions. This again reflects the fact the the automatic phonetic transcription is of a high quality.

## 5 Conclusions and Future Research

We presented a system which first generates a phonetic transcription on the basis of orthographic information and then uses the obtained transcription to produce automatic speech segmentations. Different approaches to segmentation and bias compensation were explained and tested. The forward-backward segmentation, proposed as an alternative to the commonly used Viterbi algorithm, shows very good results, especially when considering that the Viterbi segmentation was used as starting point for the manually verified segmentation. The obtained phonetic transcriptions are also of high quality, showing the potential of ASR techniques for phonetic research. To further improve the automatic system, the following actions can (and will) be taken: (1) eliminating the mismatch between training an testing conditions by retraining of the acoustic models on the corpus that must be annotated itself using the automatically generated phonetic transcriptions, (2) the introduction of single state models for phonemes that tend to be pronounced very rapidly, and (3) the derivation of more assimilation rules based on what is observed in the corpus. But even without these modifications, the results obtained by the automatic system are up to standard.

## References


1. Vorstermans, A., Martens, J.P., Van Coile, B.: Automatic segmentation and labelling of multi-lingual speech data. Speech Comm. **19** (1996) 271–293
2. van Santen, J., Sproat, R.: High-accuracy automatic segmentation. In: Proc. EUROSPEECH. Volume VI., Budapest, Hungary (1999) 2809–2812
3. Husson, J.L.: Evaluation of a segmentation system based on multi-level lattices. In: Proc. EUROSPEECH. Volume I., Budapest, Hungary (1999) 471–474
4. Ljolje, A., Riley, M.: Automatic segmentation and labeling of speech. In: Proc. ICASSP. Volume I., Toronto, Canada (1991) 473–476
5. Beringer, N., Schiel, F.: Independent automatic segmentation of speech by pronunciation modeling. In: Proc. ICPhS, San Francisco, U.S.A. (1999) 1653–1656
6. Goedertier, W., Goddijn, S., Martens, J.: Orthographic transcription of the Spoken Dutch Corpus. In: Proc. LREC, Athens, Greece (2000) 909–914
7. Pagel, V., Lenzo, K., Black, A.W.: Letter to sound rules for accented lexicon compression. In: Proc. ICSLP. Volume I., Sydney, Australia (1998) 252–255
8. Mertens, P., Vercammen, F.: The Fonilex Manual. (1997)
9. Demuynck, K.: Extracting, Modelling and Combining Information in Speech Recognition. PhD thesis, K.U.Leuven, ESAT (2001)
   Available from http://www.esat.kuleuven.ac.be/∼spch.
10. Wessel, F., Ralf, S., Macherey, K., Ney, H.: Confidence measures for large vocabulary speech recognition. IEEE Trans. on SAP **9** (2001) 288–298
11. Duchateau, J.: HMM Based Acoustic Modelling in Large Vocabulary Speech Recognition. PhD thesis, K.U.Leuven, ESAT (1998)
    Available from http://www.esat.kuleuven.ac.be/∼spch.
12. Booij, G.: The Phonology of Dutch. Clarendon Press, Oxford (1995)