Word Segmentation in the Spoken Dutch Corpus

**Jean-Pierre Martens [1], Diana Binnenpoorte [2], Kris Demuynck[3],
Ruben Van Parys [1], Tom Laureys [3], Wim Goedertier [1], Jacques Duchateau [3]**

[1]ELIS, University of Ghent, Sint-Pietersnieuwstraat 41, B-9000 Ghent, Belgium
{martens,odul,rvparijs}@elis.rug.ac.be
[2]Dept Language & Speech, University of Nijmegen, P.O. Box 9103, 6500 HD Nijmegen, The Netherlands
binnenpoorte@let.kun.nl
[3]ESAT, K.U.Leuven, Kasteelpark Arenberg 10, B-3001 Heverlee, Belgium
{kris.demuynck,tom.laureys,jacques.duchateau}@esat.kuleuven.ac.be

Abstract

This paper describes the aims of the word segmentation in the Spoken Dutch Corpus (Corpus Gesproken Nederlands, CGN), and the procedures to create it. For one million words, a manually verified segmentation will be created, whereas the remaining nine million words will only come with an automatically generated segmentation. Described are our efforts to create the best possible automatic word segmentation from an auditory verified phonetic transcription, and the development of a protocol for the manual verification of that automatic segmentation. The paper also mentions some figures concerning the manual verification of the first hundred thousand words.

1. Introduction

The Spoken Dutch Corpus (Corpus Gesproken Nederlands, CGN) project is a joint Flemish-Dutch initiative aimed at the compilation and annotation of a large corpus - ten million words - of contemporary standard Dutch as it is spoken in the Netherlands and Flanders. The project started in 1998 and will end in 2003. The goal is to create an important resource for research in various linguistic disciplines and for the development of applications in language and speech technology (for further details, see (Oostdijk et al., 2002)).

The corpus will contain speech material gathered from a great variety of socio-situational settings using various recording conditions. The recordings will range from spontaneous conversations, dialogues and multilogues to well-prepared read aloud texts by professional speakers, such as book readings (library for the blind) and broadcast news bulletins. Most recordings are sampled at 16 kHz and digitised with a 16-bit resolution. Only the recordings made through the telephone are sampled at 8 kHz.

All speech material will be orthographically transcribed, lemmatised and enriched with part-of-speech information. For about one million words more detailed information will be provided, such as a syntactic annotation, a broad phonemic transcription, a prosodic annotation and a manually verified word segmentation. The remaining nine million words will also be provided with an automatically generated word segmentation.

The goal of the word segmentation, or time alignment on the word level, is the introduction of time-markers delimiting the words. For the CGN with its diverse user groups, this word segmentation will enable easy access to the enormous amount of material. Users will be able to retrieve and to listen to relevant words and their context, when e.g. a specific linguistic phenomenon is queried. Speech technologists may use the manually verified word segmentation to get an acceptable (Oostdijk, 2000) time-alignment for the initial acoustic model training in automatic speech recognition (ASR) development. Or, they can use it for producing constrained alignments in the process of modelling pronunciation variation for ASR, or for prosodic research (e.g. for measuring word lengths, for identifying prosodic boundaries, etc.). Thus, the word segmentation will be an indispensable annotation layer for anyone using the corpus (Oostdijk et al., 2002).

The manually verified word segmentations are obtained in two stages. In stage 1, an automatic segmentation is generated, and in stage 2, this segmentation is checked and corrected by a human transcriber. For performing its task, the automatic segmentation system has access to a broad phonemic transcription produced by a phonetician. This transcription is also synchronized with the orthographic transcription. During the manual verification, the human transcriber uses *Praat* (http://www.praat.org). The word segmentation has to satisfy a number of criteria which were formulated beforehand in a written protocol. It will appear that for some of these criteria we deviate to some extent from the practice, as adopted in for example Switchboard (Godfrey et al., 1992) and Verbmobil (http://verbmobil.dfki.de/). For instance, we decided to delimit not only linguistic words, but also clear pauses between these words. Even more unusual is that we also permit words to overlap under certain conditions.

This paper first describes our attempts to produce the best possible automatic word segmentation (AWS) as a starting point for the manual checking stage (sections 2 and 3). Then, in section 4, it reviews the procedures for creating a manually verified word segmentation (MWS) starting from the AWS. Finally, in section 5, it reports some results with respect to the time consumption of the manual checking stage, and some experience that was gathered while processing the first one hundred thousand words.

2. Automatic Word Segmentation (AWS)

In order to maximize the efficiency of the manual verification step, we have tried to supply the human transcribers with the best possible AWS of the speech files. A good AWS is one providing accurately positioned time-markers, as well as good estimates of the reliability of these time-marker positions. In fact, such a confidence measure may help the human transcriber in deciding whether or not he should try to move a boundary in order to improve the AWS. The following two circumstances facilitate the creation of a good AWS:

1. The CGN protocol for orthographic transcription (Goedertier et al., 2000) states that long speech files have to be segmented in short chunks (maximum 3 seconds) before starting the transcription. Moreover, chunk boundaries are only allowed to occur in audible pauses between words. As a result, the AWS can be obtained chunk by chunk.

2. The AWS can be derived from the best possible input viz. a manually verified broad phonemic transcription which is synchronized with the words of the orthographic transcription.

The AWS is obtained in two steps. First, an aligner performs a forced alignment of the speech with the string of phonemic symbols representing that signal. Second, the emerging automatic phonemic segmentation (APS) is converted into an AWS that meets all the specifications outlined in the protocol for manual verification (see section 4). Confidence measures of the word boundaries are obtained from information that is internally available in the aligner.

The consortium responsible for performing the word segmentations within the CGN has access to two aligner technologies: one based on Hidden Markov Models (HMMs) (Steinbiss et al., 1993; Duchateau, 1998; Demuynck, 2001) and one based on Stochastic Segment Models (SSMs) (Vorstermans et al., 1996). The CGN contains Dutch and Flemish files which need aligners with specific acoustic models. Therefore, the Dutch data are processed in Nijmegen, whereas the Flemish files are processed in Leuven and Gent.

Since none of the two aligners was previously evaluated on a corpus covering the diverse conditions encountered in the CGN (different types of noise, room acoustics, recording conditions, speech modes, etc.) our first aim was to assess the accuracies of the two technologies in view of the envisaged task. The second aim was to investigate whether the AWS could be improved by combining the two aligner outputs in a so-called fusion system.

Before describing the evaluation and combination issues, we review the main characteristics of the HMM and SSM technologies that we have used in our experiments.

2.1. The HMM Aligner

The HMM aligner operates according to the following principles :

1. The speech is converted into a parametric representation which consists of a 39-dimensional feature vector produced every 10 ms. For each vector, a mel scaled spectrum, augmented with first and second order time derivatives, is calculated and transformed linearly in order to adjust it to modelling with diagonal covariance gaussians (Duchateau et al., 2001).

2. The HMMs are three-state left-to-right context-dependent phone models with partially tied gaussian densities. They were trained on read speech from speakers who did not participate in the CGN data collection.

3. Whole chunk acoustic models are obtained by mapping the phones of the phonemic transcription onto their respective HMMs, and by simply concatenating the different phone HMMs.

4. The single best assignment of feature vectors (frames) to acoustic model states is identified by means of a Viterbi search. From this assignment, it is straightforward to derive time-markers corresponding to a state, phoneme or word level segmentation.

Serious mis-alignments only occur when (1) the phonemic transcription is inaccurate, or (2) the acoustic HMMs fail to adequately model certain speech effects. An example of the first type of errors is the occurrence of non-speech events (eg. coughing, laughing) which are not properly transcribed phonemically. The main cause of errors of the second type is the minimal duration constraint (30 ms) on the phones, imposed by the three-state left-to-right HMM.

In order to provide more than just the time-markers, the HMM aligner is extended to produce confidences for these time-markers. The basic principle underlying the calculation of these confidences is that in case the hypothesized word boundary is inaccurate, the data in the vicinity of that boundary will not match well with the corresponding acoustical states.

Assume that the observation likelihood of state q given the feature vector x of a particular frame is described by means of an observation density function $f_q(x) = f(x|q)$. Assume further that the states are connected by transitions with transition probabilities $a_{ij} = P(q^{(t+1)} = q_j | q^{(t)} = q_i)$, with $q^{(t)}$ being the HMM state at frame t. To detect possible errors, we calculate the normalised acoustic log likelihood of the speech given the phones at both ends of a word boundary as

$$\begin{aligned}\log(\lambda(X|t,l,r,\bar{q})) &= \frac{1}{l+r}\sum_{i=t-l}^{t+r-1}\log\left(\frac{f(x_i|q^{(i)})}{f(x_i)}\right),\\ f(x_i) &= \sum_q f(x_i|q)\,P(q),\end{aligned}$$

In this expression, l and r represent the length (in frames) of the phones left and right of the boundary, $q^{(i)}$ is the best matching state (according to the Viterbi alignment) for frame i, $P(q)$ the a-priori probability of state q, and t the boundary (a frame number) for which the confidence score is calculated.

It was established experimentally that a window of 1 phone left and 1 phone right yielded the best results, but the above formula can easily be modified to incorporate more phones on each side. A more detailed description of the technique can be found in (Laureys et al., 2001).

2.1.1. The SSM Aligner

The SSM aligner was formerly described in (Vorstermans et al., 1996). It operates according to the following principles :

1. The speech is analyzed by an auditory model (Immerseel and Martens, 1992). Every 10 ms, it produces a 23-dimensional auditory spectrum (loudness contributions in 23 channels), a voicing evidence and a fundamental frequency (pitch), as well as 5 samples (one per 2 ms) of the total loudness as a function of time.

2. The total loudness function is used to produce landmarks in the speech. Landmarks which are scored by a neural network as potential phonemic boundaries are retained, and any subset of these retained landmarks constitutes a possible phonemic segmentation of the speech.

3. The best phonemic segmentation is obtained by a Viterbi search that uses two multi-layer perceptrons (MLPs) for scoring the acoustic features of segments between retained landmarks. The derivation of an AWS from that segmentation is straightforward.

One MLP, called the segmentation MLP, estimates the posterior probability that landmark t_{j+k} (with $k = 1, .., 5$) is the endpoint of a phonemic segment, given the acoustic features, and given that t_j has already been identified as the endpoint of the previous phonemic segment. By restricting k, the number of segmentations is kept at an acceptable level. The other MLP, called the classification MLP, estimates the posterior probability that segment (t_j, t_{j+k}) is of phonemic class C_m ($m = 1, ..., M$), given the acoustic features, and given that (t_j, t_{j+k}) has already been identified as a phonemic segment. It was demonstrated (Vorstermans et al., 1996) that by using just 5 broad phonemic classes describing the manner of articulation, one has sufficient phonemic detail for performing a good segmentation.

There is also experimental evidence (Vorstermans et al., 1996) that acoustic models trained for one language (the source language) are easy to adapt to another language, without the need for hand labeled data in the target language. Consequently, the SSM system should also be easy to adapt to new acoustic conditions. This could definitely be an advantage in view of the diversity of conditions encountered in the CGN material.

In order to provide more than just phonemic boundaries, the SSM system was extended to produce confidences for the hypothesized word boundaries. After some trials, the average posterior probability of the correct broad phonemic class in two phonemic segments left and two phonemic segments right of the word boundary was defined as the confidence measure.

2.2. Experimental Evaluation of Aligners

In order to eliminate possible biases towards any of the two aligners, we performed tests on two subcorpora, called SC-01a and SC-01b. The SC-01a files (3548 words) were verified in Leuven starting from the AWS of the HMM aligner (HMM-AWS), the SC-01b files (3945 words) were verified in Gent starting from the AWS of the SSM aligner (SSM-AWS). The material in both subcorpora was selected from the 6 of the 14 main components of the CGN (Oostdijk et al., 2002).

For each word w_i, the aligner produces a *word boundary solution* $s_i = (t_e(i), t_b(i+1), c_i)$ consisting of the endpoint of word w_i, the starting point of the next word and the confidence measure computed for the hypothesized solution. If $t_b(i+1) > t_e(i)$, it means that a silence was found between w_i and w_{i+1}. If $t_b(i+1) < t_e(i)$, it means that the words share a phoneme (see section 4 for more details on phoneme sharing). We introduce the local distance

$$d_i = \frac{1}{2}\,|t_b(i+1) - t_{Mb}(i+1)| + \frac{1}{2}\,|t_e(i) - t_{Me}(i)|$$

as a measure of the deviation between the computed boundary solution s_i and the solution s_{Mi} emerging from the manual verification.

In the protocol for manual verification, people are instructed to put a boundary back at its original position when moving it did not yield an improved segmentation. However, as *Praat* did not support this feature, the boundaries were not put back **exactly** at the original position. Consequently, $d_i = 0$ can be assumed to indicate that no time was spent on trying to change the boundary solution s_i.

The main properties of the local distance statistics for the two aligners are listed in Table 1. The average distance is the arithmetic mean of the d_i. From the figures it follows that globally speaking the HMM aligner is the best. Its AWS forms a good starting point for the manual verification (only 23 % of the boundaries need to be analyzed, and just having to move a boundary that is already more or less at the right position is considered an easy task). Since the outcome of this experiment, all manual verifications were done starting from the HMM aligner output.

local distance property	HMM-AWS (SC-01a)	SSM-AWS (SC-01b)
average d_i	11 ms	24 ms
percent $d_i = 0$	77 %	81 %
percent d_i <20 ms	81 %	84 %
percent d_i >100 ms	2 %	8 %

Table 1: Statistics of the local distances d_i measured on the two subcorpora SC-01a (HMM-AWS) and SC-01b (SSM-AWS).

Another conclusion that was drawn from our experiments is that confidence measures do not say very much about the correctness of the boundary solution. Nevertheless, there is **some** separation between the confidence measure histograms for altered ($d_i > 0$) and unaltered ($d_i = 0$) boundaries in the case of SSM.

2.3. Agreement between Aligners

In a second series of experiments, we investigated the degree of agreement or disagreement between the two aligners, and the potential use of the disagreement to produce an even better alignment.

For these experiment we used a corpus SC-02 (about 7000 words) of files that were selected from the various components (Oostdijk et al, 2002) the CGN. They were all verified starting from an HMM-AWS.

First of all it appears that when the two aligner solutions differ considerably, one of the solutions is usually much closer to the manual solution than the average of the two. Therefore we have investigated the potential of a system that would try to select the best of two solutions (lowest d_i) at every word junction. Looking at the data in this perspective, it appeared that the SSM aligner offers a better solution (smaller d_i) for 45 % of the altered boundaries. Selecting this solution would reduce the average d_i from 8.6 to 6.1 ms (a reduction of nearly 32 %). Consequently, it seemed worthwile to investigate the possibility of developing a data fusion system to create a better alignment.

3. A Data Fusion System

3.1. Methodology

If $\mathbf{w}$ and $\mathbf{f}$ represent the orthographic and phonemic transcription of the speech, and $\mathbf{s}_1 = \{s_{11}, .., s_{1i}, .., s_{1N}\}$ and $\mathbf{s}_2$ the solutions of aligners 1 and 2, then the fusion system aims at determining the most likely solution $\hat{\mathbf{s}}$, given by

$$\hat{\mathbf{s}} = \arg\max_{\mathbf{s}} P(\mathbf{s}|\mathbf{s}_1, \mathbf{s}_2, \mathbf{w}, \mathbf{f}) \tag{1}$$

with the maximum taken over all solutions consisting of s_i which are either equal to s_{1i} or s_{2i}. In order to obtain a feasible expression for the right hand side of Eq. (1), the following assumptions were made :

1. **The Markov assumption**
   Of all the previous decisions, only the one about s_{i-1} is relevant for making a decision about s_i.

2. **The independency assumption**
   The solution s_i only depends on the boundary solutions s_{1i} and s_{2i} taken from the observed solutions $\mathbf{s}_1$ and $\mathbf{s}_2$.

3. **The word identity assumption**
   The identity of word w_i cannot be used in the decision process. Only its phonemic transcription $\mathbf{f}_i$ can be used to compute expected minimum, average and maximum word durations.

By making these assumptions, the probability to evaluate can be written as

$$P = \prod_{i=1}^{N} P(s_i|s_{i-1}, s_{1i}, s_{2i}, \mathbf{f}_i)$$

For determining $\hat{\mathbf{s}}$ we first introduce a two-state ergodic Markov automaton (see figure 1) with word junction dependent transition probabilities P_{kli}. Visiting state 1 at time i means selecting the solution of aligner 1 at that time (and likewise for state 2). If the state of arrival at time i is denoted q_i, the transition probabilities P_{kli} are given by

$$P_{kli} = P(q_i = l | q_{i-1} = k, s_{k,i-1}, s_{1i}, s_{2i}, \mathbf{f}_i)$$

with $s_{k,i-1}$ representing the solution of aligner k at time $i-1$. At any time, transitions may be prohibited because they would yield negative word lengths. However, for the transitions that are allowed, the transition probability is derived from the output of an MLP that was trained to estimate the probability of being in state 1 at time i, given the elements in the condition part. The search for the best state sequence is a straightforward Viterbi-search.

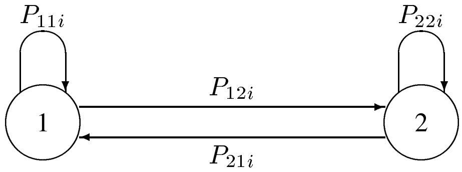

Figure 1: Ergodic Markov automaton for deciding which of two solutions (emerging from two aligners) to select at word junction i.

The MLP has 15 inputs. The first ones are the previous state (q_{i-1}), and the time differences $t_{e1}(i) - t_{e2}(i)$ and $t_{b1}(i+1) - t_{b2}(i+1)$. Next we have the number of phonemes in $\mathbf{f}_i$, the expected average duration of w_i, and the expected minimal and maximal duration with respect to that average. All this duration information is computed on the basis of $\mathbf{f}_i$ and some over-all phoneme duration statistics. Finally, there are two times 4 inputs describing what would be the situation if aligner 1 or aligner 2 were selected respectively. These variables are: the duration of word w_i relative to its expected average duration, the confidence measure c_i, the duration of the pause between w_i and w_{i+1} (0 in case there is no pause), and the duration of the overlap between the two words (0 in case there is no overlap).

3.2. Experiments and Results

For this experiment, the SC-02 subcorpus was divided into training and test files. The test set consisted of about 1500 words, and the speakers in this set did not occur in the training files. The fusion system was trained in two phases:

1. **Phase 1.** The MLP is trained to produce an output y_i that is either 1, 0 or 0.5. If $d(s_{1i}, s_{Mi})$ is denoted as d_{1i}, then the teachin output is

   $$\begin{aligned} y_i &= 1 \quad \text{if } d_{1i} < d_{2i} - d_o \\ y_i &= 0 \quad \text{if } d_{2i} < d_{1i} - d_o \\ y_i &= 0.5 \quad \text{in all other cases} \end{aligned}$$

   The distance d_o was set to 10 ms and was introduced to avoid a forced choice in case there is no significant gain to attain.
   Note that since s_{i-1} is a necessary input to the MLP, only those word junctions were considered for which the above rule had produced a teaching output of 0 or 1 at junction $i-1$.

2. **Phase 2.** Based on the alignments that were produced by the Viterbi-search (based on the MLP of phase 1), new training examples are collected and the MLP can be retrained/updated on the basis of these examples.

In table 2 we have collected the performances of the HMM aligner and the data fusion system (F) in phases 1 and 2 on the test set of SC-02. The data show that the fusion system does cause a small improvement, but this improvement is too small to justify the considerable extra effort of sending the files through the two aligners and the fusion system.

system	times SSM	average	times
	selected	distance	d <20 ms
HMM	0.0 %	8.60 ms	81.1 %
F(phase1)	1.7 %	8.39 ms	81.7 %
F(phase2)	1.1 %	8.32 ms	81.8 %

Table 2: Performances of the HMM aligner and the fusion system after phases 1 and 2 on the test files of SC-02.

Even though it will not be used for producing the MWS of the one million words for which there is a verified phonemic transcription, the fusion system may still appear to yield a benefit for the production of the best possible AWS for the remaining nine million words for which no such transcription is available, and for which the individual aligners will produce much more errors.

4. Manual Verification

The CGN subcorpus that needs manual verification represents approximately 100 hours of speech. To verify such an amount of data in a restricted time, it is inevitable to distribute the job over more than one person. Moreover, since Flanders and The Netherlands will each do their share of the work, these persons will also work at different sites (2 sites in Flanders and 1 in The Netherlands). This calls for a set of rules and agreements aiming at an identical data handling in identical situations.

4.1. Defining Rules and Principles

The protocol for manual verification of an AWS (Binnenpoorte, 2002) serves both as a user manual and a reference. The following basic principles were adopted :

- Maintain the consistency with both the orthographic transcription and the broad phonemic transcription that specifies what one should hear when listening to the words. For this reason, changing any of these transcriptions is prohibited.
- Be pragmatic when verifying the time-markers. Do not move markers when they are well placed, and do not move markers in order to make short words – such as *de* (the) and *en* (and) – sound intelligible in isolation.
- Consider pauses and non-linguistic sounds as words, in such a way that they are delimited by time-markers as well.

The user manual describes the procedures to follow, and it provides examples to illustrate better what to do while checking the supplied AWS. Some general rules are:

- Listen to every segment, whether it is a linguistic word or not (e.g. a pause).
- Decide whether the segment sounds acceptable given the orthographic and phonemic transcription.
- Move time-markers only when necessary, and do so according to the rules described in the protocol.

In spite of these rules, a number of problems can arise as a consequence of working with continuous speech in which words are usually not separated by pauses. Therefore, two issues are exhaustively described in the protocol, namely that of phonemes that are shared by two words, and that of phonemes that are inserted between words. Both phenomena form a potential cause of problems as they can affect the consistency between the orthographic and phonemic transcription, and possibly change the number of words.

Shared Phonemes

It is very common in Dutch that words share phonemes on word boundaries. This is the case when the final phoneme of a word is the same, or becomes the same after assimilation, as the initial phoneme of the following word. The phenomenon is known as degemination (Booij, 1995). When a phoneme is shared, this is displayed in the phonemic transcription. Here are two examples in which we display the orthographic transcription, the phonemic transcription and a translation in English (between brackets).

> ik wil naar Rotterdam
> Ik wIl nar_rOt@rdAm
> (i want to go to Rotterdam)

> hij komt terug
> hE+ kOmt_t@rYx
> (he comes back)

The phoneme sharing is indicated by the underscore.

When dividing continuous speech into words, it is not clear where to put the word boundary in the case of a shared phoneme. One solution would be to put it in the middle of that phoneme. In principle, this results in 'incomplete' words when made audible, but in practice this usually does provide an acceptable solution. Only in the case of a shared plosive (e.g. /p/, /t/, /k/, /b/, /d/ and /g/), there are reasons for adopting another strategy. In fact, a plosive cannot be split up into two parts such that it sounds acoustically acceptable in both words that share it. It was therefore decided to treat a shared plosive as a separate segment (marked with an underscore). When making either of the two words audible, this separate segment has to be considered as part of that word. This means that word segments can overlap. It also means that the automatic segmentation must generate a segment for each shared plosive, and that this segment must be verified according to the instructions described earlier. Figure 2 is a detail of a screen in the program *Praat* showing an example of a shared plosive.

Note that there remain some 'difficult' cases to handle, like a plosive that represents in itself a word and that is shared with either the preceding or the subsequent word or both, or, a sequence of two or more phonemes that is shared by two words.

> dus 'k keek naar hem
> dYs k_kek nar hEm
> (so I looked at him)

For all these cases, examples and solutions are presented in the protocol.

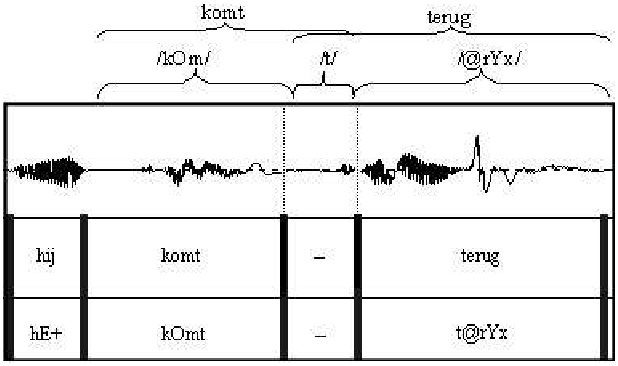

Figure 2: Shared phoneme in 'hE+ kOmt_t@rYx' (he comes back).

Inserted phonemes
Another problem concerns phonemes that are inserted between words.

> daarom doe ik het nu
> darOm du-w-Ik @t ny
> (that's why I do it now)

These phonemes do not have an orthographic equivalent but they do appear in the phonemic transcription. The inserted phoneme is thus transcribed in both word segments in the MWS layer, and marked with a hyphen to serve as a flag for the human transcriber when verifying the word segmentation. As the inserted phoneme does not actually contribute to the identity of the words all inserted phonemes are assigned to both words (like shared plosives).

4.2. Human Correctors

For reasons of cost and availability, we decided to hire students for the job. These student were neither acquainted with speech technology, phonetic science, etc., nor did they have any experience with the program *Praat*.

During a two hour instruction session they were made familiar with the material, *Praat* and the goal of their work. They were then asked to study the protocol very carefully and to keep it with them as a guideline and reference during their verification work. The students are hired on a half time basis to prevent complaints of RSI and loss of concentration during their work. All students started by verifying relatively easy material (monologues recorded in rather 'clean' environments) before they moved to the verification of more difficult stuff.

5. Results and Experiences

Thus far, the manual verification has been performed on about one hundred thousand words. However, this material does not yet include any telephone speech, nor does it contain many difficult dialogues already.

By comparing the manual with the automatic word segmentations we could verify that between 15 and 30 % of the boundaries in a file needed manual verification, the others were immediately accepted as being correct.

Although no systematic measurements are available yet, we do have some good indications about the time that is needed to perform the manual verification of one minute of speech. For the easy material (clean read speech), this may take less than 20 minutes, but for interactive multilogues, this can go up to 50 minutes.

6. Acknowledgements

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

7. References

D. Binnenpoorte. 2002. Protocol voor manuele verificatie van automatisch gegenereerde woordsegmentaties. *Internal publication Spoken Dutch Corpus (CGN) project.*

G. Booij. 1995. *The Phonology of Dutch.* Clarendon Press, Oxford.

K. Demuynck. 2001. Extracting, modelling and combining information in speech recognition. Phd, ESAT, KULeuven.

J. Duchateau, K. Demuynck, D. Van Compernolle, and P. Wambacq. 2001. Class definition in discriminant feature analysis. *Proceedings of Eurospeech*, 2001:1621–1624.

J. Duchateau. 1998. Hidden markov model based acoustic modelling in large vocabulary speech recognition. Phd, ESAT, KULeuven.

J. Godfrey, E. Holliman, and J. McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. *Proceedings ICASSP*, 1992:517–520.

W. Goedertier, S. Godijn, and J. Martens. 2000. Orthographic transcription of the spoken dutch corpus. *Proceedings of LREC*, (Athens):909–914.

L. Van Immerseel and J. Martens. 1992. Pitch and voiced/unvoiced determination with an auditory model. *Journal of the Acoustical Society of America*, 91:3511–3526.

T. Laureys, K. Demuynck, J. Duchateau, P. Wambacq, and A. Bogan-Marta. 2001. Assessing segmentations: Two methods for confidence scoring automatic HMM-based word segmentations. *Proceedings 6th Int. Conf. Engineering of Modern Electric Systems*, (Oradea):116–121.

N. Oostdijk, W. Goedertier, F. Van Eynde, L. Boves, J. Martens, M. Moortgat, and H. Baayen. 2002. Experiences from the spoken dutch corpus. *Proceedings LREC*, 2002:(this issue).

N. Oostdijk. 2000. The spoken dutch corpus. overview and first evaluation. *Proceedings LREC*, (Athens):887–893.

V. Steinbiss, H. Ney, R. Haeb-Umbach, B. Tran, U. Essen, R. Kneser, M. Oerder, H. Meier, X. Aubert, C. Dugast, and D. Geller. 1993. The philips research system for large-vocabulary continuous-speech recognition. *Proceedings of Eurospeech*, (Berlin):2125–2128.

A. Vorstermans, J. Martens, and B. Van Coile. 1996. Automatic segmentation and labeling of multi-lingual speech data. *Speech Communication*, 19:271–293.