IMPROVING THE PHONETIC ANNOTATION BY MEANS OF PROSODIC PHRASING

H. Vereecken [1], *A. Vorstermans* [2], *J.-P. Martens* [1] *and B. Van Coile* [1,2]
[1]ELIS, University of Gent, Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium
[2]Lernout & Hauspie Speech Products NV, Sint-Krispijnstraat 7, B-8900 Ieper, Belgium
E-mail: halewijn@elis.rug.ac.be

ABSTRACT

It was established that the performance of our annotation system [8] is affected by the length of the utterances: the error rate, the CPU-load and the memory requirements tend to increase as the utterances get longer. In this contribution the speech signal is first segmented into speech, pauzes and noise (breaths, clicks, ...) and subsequently split in signal phrases prior to the annotation. Experiments on 3 different databases (3 languages) demonstrate that this stategy yields a significant improvement of the annotation accuracy.

1. INTRODUCTION

It is well known that high quality speech synthesis can only be achieved by incorporating accurate *prosodic models*. The development of prosodic models requires large speech databases which are labeled both at a *phonetic* and a *prosodic* level. For that purpose an automatic phonetic and prosodic labeling algorithm is incorporated in an interactive labeling tool. Obviously, the better the automatic labeling works, the less manual interventions are required to reach a certain quality of the final labeling. Furthermore, the importance of the phonetic information for the automatic prosodic labeling of speech is pointed out by several researchers [1, 10]. Therefore, the very first step towards a reliable automatic prosodic labeling is an accurate phonetic annotation (segmentation and labeling). Campbell e.g. had to throw away part of his database because the automatic phonetic segmentation had failed [2]. In this article, we focus on the automatic and semi-automatic *phonetic* annotation of speech passages.

In our automatic annotation system [8], the best annotation is found by maximizing the joint probability (given the acoustic observations) of the phonetic segmentation and the state sequence through a model that was derived from the phonetic transcription of the utterance. The maximization is done by means of a Viterbi alignment. To increase the speed of the search, only trellis nodes in a belt around the diagonal are investigated. However, if we are dealing with long utterances, this belt has to be quite large. Hence, annotating very long utterances requires a large amount of memory and causes a dramatic increase of the computational load. Furthermore, long utterances usually comprise breaths, microphone clicks and other non-stationary background noises which make the alignment error-prone. Experiments confirm that more errors occur as the utterances become longer. Moreover, the manual control of one long utterance is a more tedious task than the control of the composing shorter parts, at least if no errors occur across these parts.

One way to counter the mentioned disadvantages is to break up larger entities (discourses, paragraphs, sentences) into smaller ones (phrases), *prior to the automatic annotation*. Powerful cues, such as preboundary lengthening [1, 10], can therefore not be exploited.

In this contribution, we describe an algorithm for automatically chopping large paragraphs or sentences into (smaller) prosodic phrases. The phrase-length we are aiming at is of the order of 2 seconds. The input consists of the speech signal and its phonetic transcription. The strategy breaks down into 2 major parts:

1. Search for prosodic phrases in the signal.
   The goal is to locate silences, breaths and clicks in the signal and to use these to demarcate the phrases.
2. Perform a signal-to-transcription mapping.
   Having split up the signal, one has to identify the corresponding parts of the phonetic transcription.

An obvious restriction would be that prosodic phrases are delimited by word boundaries. However, some databases contain extremely long unvoiced stop closures, sometimes as long as 250 ms. In some cases these unvoiced stops are immediately followed by a vowel, making it hard to distinguish their closures from word boundaries solely on the basis of simple acoustic observations. Therefore, we have allowed phrases to be terminated by these unvoiced stop closures as well. I.e. all word boundaries and all word-internal unvoiced stop closures are marked as possible phrase boundaries in the phonetic transcription. It is clear that this definition of a 'prosodic phrase' clashes with all existing ones. We wish to emphasize however that our goals are just to speed up and to improve the automatic annotation of long utterances, to reduce the memory requirements, and to facilitate a computer-assisted manual annotation of long utterances. The concept of a phrase is merely a means of achieving this goal.

Experiments on 3 different languages demonstrate that splitting the input prior to the annotation yields a significant improvement of the annotation accuracy.

2. SPLITTING IN SIGNAL PHRASES

2.1. Principle

Searching for phrase boundaries in the signal boils down to looking for silences of more than 150 ms, and for breaths, clicks, and other non-stationary background noises which tend to manifest themselves as isolated syllable- or phone-sized unvoiced segments of relatively high energy. Wightman and Ostendorf [9, 10] used a frame-based likelihood classifier to detect breaths and large pauzes, thus requiring an explicit training stage. The absence of manually marked data forced us to adopt a strategy based on a blind syllabification of the signal: the acoustic observations are described as a sequence of voiced syllabic units (speech), unvoiced syllabic units (breaths, clicks, aspirated stops) and pauzes (silences, closures). Breaths can be separated from clicks and aspirated stops on the basis of duration criteria. Their presence will be signaled to the contemplated prosodic labeling module [10]. Unfortunately, we can not compare our approach to other approaches, because manually marked data are not available for this purpose.

2.2. Syllabification

The generation of potential syllable boundaries is based on an analysis of the energy contour. This implies that the detected syllables will not necessarely correspond to the syllables provided by a dictionary. Ideally, syllable boundaries are placed on those time instants where the coarticulation between the separated sounds is minimal. The syllabification is performed in 4 stages.

In a first stage, the energy contour provided by an auditory model [6] is smoothed by a first-order low-pass filter with a time constant τ. The extrema in the resulting energy contour are determined according to the following principles [4]:

- If searching for a maximum, continue to do so until a value is encountered which is smaller than max$-\delta$, where max is the maximum found since the beginning of the search. From then on, start searching for a minimum.
- If searching for a minimum, continue to do so until a value is encountered which is larger than min$+\delta$, where min is the minimum found since the beginning of the search. From then on, start searching for a maximum.

The value of δ is a fixed percentage of the maximum energy that can be encountered, and not a fraction of the active min or max. Obviously, the larger δ, the more pronounced an extremum has to be before it will be detected. The minima and maxima correspond to potential syllable boundaries and syllable nuclei. Thus we arrive at what Mermelstein [5] referred to as *syllabic units*, i.e. syllable-sized speech segments located by loudness criteria.

It is clear that the boundaries of these syllabic units do not necessarily coincide with phone boundaries. Therefore, in a second stage the potential syllable boundaries are aligned with potential phone boundaries provided by a presegmentation algorithm [8]. This presegmentation identifies potential phone boundaries as sudden increases in the derivative of the energy contour. A potential syllable boundary is moved to the nearest potential phone boundary having the smallest energy, taking into account however the following constraints:

- A potential syllable boundary can not be moved over more than 35 ms.
- The syllable structure (alternating minima and maxima) can not be disturbed.

If these constraints can not be fulfilled, the potential syllable boundary is removed, i.e. two potential syllables are merged.

In a third stage silence segments are inserted in the syllable structure. A potential phone segment is regarded as a silence if both its average energy and its average voicing evidence are small. The voicing evidence is provided by the AMPEX algorithm [6].

A syllable having insufficient voicing evidence (maximum voicing smaller than v_t) is likely to be an insertion. Therefore, in a final stage such a syllable is merged with an adjacent syllable or, if it is surrounded by silences, it is marked as an *unvoiced syllable*. The latter usually is an indication of an aspirated stop, a microphone click or a breath.

2.3. Locating phrase boundaries

For the purpose of detecting phrase boundaries in the speech signal, the unvoiced syllables have to be removed. If an unvoiced syllable is in the proximity of a regular (voiced) syllable, it is usually an aspirated stop, and it is merged with the accompanying syllable. The enclosed silence is absorbed in the resulting syllable. If the unvoiced syllable is not in the proximity of a regular syllable, it is replaced by a silence. Given this syllabification, the detection of phrase boundaries is straightforward: any silence larger than 150 ms is regarded as a phrase boundary. However, phrases as small as one voiced syllable are not allowed (same restriction as Campbell [1]). They are always merged with the previous phrase.

2.4. Experimental Evaluation

2.4.1. Syllabification

The syllabification is tested on a Flemish database comprising 3304 syllables. To take the delay induced by the low-pass filter into account, syllable boundaries detected at $t = t_s$ are moved to $t = t_s - \tau/2$, where $\tau/2$ is the group delay of the filter at its cut-off frequency. The syllable boundaries are then aligned with the manually marked

τ (ms)	v_t	Corr	Ins	Del
20	no	3188	1112	57
20	yes	3159	338	72
35	yes	3011	167	148

Table 1: *Performance of the syllabification with and without postprocessing: number of detected syllables that are correct, inserted or deleted.*

phones. If a syllable contains exactly one vowel, it is a correct one. If it does not contain a vowel, it is an insertion. If it contains $k > 1$ vowels, there are $k-1$ deletions. Table 1 depicts some results; δ and v_t (if applied) are chosen equal to 5% of the maximum energy and 10% of the maximum voicing, respectively. If no postprocessing is used (row 1), the number of insertions is extremely high. Postprocessing (row 2) reduces the number of insertions dramatically, while at the same time keeping the number of deletions low. The value of τ can be used to control the ratio deletions/insertions. Row 3 represents the settings minimizing the total error (insertions+deletions). They are the ones that will be used for the remaining of this article.

2.4.2. Locating Phrase Boundaries

The detection of phrase boundaries was manually verified on the English and Italian parts of EUROM0. Each corpus contains one paragraph, which is read by 4 speakers (about 8 minutes of speech). All detected phrase boundaries corresponded either to word boundaries or to unvoiced stop closures preceding a vowel, as intended. The latter only occurred in the Italian corpus.

The algorithm was also tested on paragraphs of the Flemish corpus COGEN (cfr. [3]). In our experiments we tested on 30 manually labeled paragraphs, each one read by a different speaker (about 15 minutes of speech). All phrase boundaries corresponded to word boundaries.

3. MAPPING TO PHONETIC PHRASES

Once the signal phrases are available, the phonetic transcription has to be split into the corresponding phonetic phrases. Suppose that we have retained a number of possible phonetic phrase endings at a particular signal phrase ending, and that we have computed a score for each of these combinations. We can then *simultaneously* align the next signal phrase to a number of phonetic phrases which are likely to correspond to this signal phrase, and obtain a set of new scores representing the probabilities of reaching the end of the analysed signal phrase while arriving at the ends of the considered phonetic phrases. In order to constrain the number of phonetic phrases to examine for each signal phrase, the number of syllables detected in the signal phrase (N_s) is compared to the expected number of detected syllables for the potential phonetic phrases. If the phonetic phrase consists of the phonemes $f_1, ..., f_N$, this expectation is defined as

$$N_p = \sum_{n=1}^{N} \Pr\left(\text{nucleus} \mid f_n\right)$$

where $\Pr\left(\text{nucleus} \mid f_n\right)$ is the probability of detecting a syllable nucleus on f_n. Only phonetic phrases whose N_p's are sufficiently close to N_s are examined.

At the end of a signal phrase, 3 situations can occur:

1. If all hypothesized phonetic phrases yield low scores, all hypotheses are retained. Else,
2. If one of the top-2 scores corresponds to a phonetic phrase terminated by a punctuation mark, *and* the following word boundary is not accompanied by a punctuation mark, only this phrase endpoint is retained for further processing. Else,
3. All paths with a reasonable score are kept alive.

Even for the first case, the number of remaining paths is rather limited. This path elimination thus results in a significant reduction of the computational load, compared to the alignment of the whole utterance. Finally, the chunk of phonetic phrases is retrieved from the best alignment of the whole signal to the complete transcription. It is important to note that the annotation itself is *not* retained: the backtracking is performed at the phrase level instead of at the segmental level. Thanks to this the alignment can proceed with a limited amount of memory.

4. EXPERIMENTAL EVALUATION

We have carried out 4 experiments on the English and Italian parts of EUROM0, and on COGEN:

Exp1: The *paragraphs* are automatically annotated by the baseline system of [8].

Exp2: The paragraphs are manually split into *sentences*, and the automatic annotation is performed using the correct phonetic sentences.

Exp3: The paragraphs are automatically split into *phrases* by our algorithm, and the automatic annotation is performed using the automatically found phonetic phrases.

Exp4: The paragraphs are automatically split into *phrases*, but the automatic annotation is performed using the manually provided correct phonetic phrases.

The input of the automatic annotation is a standard phonemic transcription (concatenation of canonical word pronunciations).

The experimental results are depicted in tables 2, 3 and 4. Clearly, the longer the utterances, the worse the automatic annotation system becomes. Comparing Exp1 and Exp4, we can see that the total error rate drops from 36.26% to 33.45% for the English corpus, from 32.55% to

Exp	1	2	3	4
Del	7.25	7.03	6.69	6.71
Ins	2.70	2.59	2.26	2.09
Sub	6.49	6.17	5.41	5.29
Far	19.82	19.84	20.02	19.35
Segm err	29.77	29.46	28.97	28.15
Total err	36.26	35.63	34.38	33.45

Table 2: *Error rates (%) for the English part of EUROM0 (5364 handlabels): deletions, insertions, substitutions, boundary deviations, segmentation error and total error.*

Exp	1	2	3	4
Del	9.07	8.88	8.52	8.36
Ins	1.00	0.75	0.94	0.92
Sub	2.98	3.08	3.22	3.27
Far	19.50	18.86	17.38	16.70
Segm err	29.57	28.48	26.84	25.98
Total err	32.55	31.56	30.07	29.26

Table 3: *Error rates (%) for the Italian part of EUROM0 (6173 handlabels).*

29.26% for the Italian corpus, and from 35.56% to 32.92% for the Flemish corpus. These drops imply significant improvements (95% confidence intervals). The improvements are equally divided among all types of errors. The decrease in segmentation error rate is only significant for the Italian and the Flemish corpus. The circumstances of Exp4 are typically those of the computer-assisted manual annotation we have in mind: the utterance is split into phrases by the tool described here, and the human labeler just corrects the signal-to-transcription mapping simply by listening to the signal phrases. If no manual correction is performed, the total error rate is expected to increase by about 1% (Exp4 versus Exp3).

The results on EUROM0 mentioned in [7, 8] were obtained via Exp2, but using a transcription derived from the manual label sequences.

5. CONCLUSION

We have shown that the phonetic annotation of long sentences and paragraphs can be improved by introducing prosodic phrasing based on a syllabification of the speech, prior to the annotation. The error rates, the memory requirements and the computational load all drop significantly. The algorithm also facilitates the task of the human labeler who is to verify the automatic segmentation and labeling: errors no longer occur *across* phrase boundaries, but *within* relatively short phrases.

What is described here, has to be seen as a first, indispensable step towards a fully automated prosodic labeling of speech corpora. The detected syllables are classified as

Exp	1	2	3	4
Del	5.08	6.01	5.43	5.43
Ins	5.87	4.74	4.89	4.86
Sub	12.11	11.60	11.52	11.31
Far	12.51	11.20	11.51	11.32
Segm err	23.46	21.95	21.83	21.61
Total err	35.56	33.55	33.35	32.92

Table 4: *Error rates (%) for COGEN (11622 handlabels).*

speech, silence or noise (breath, click) and the speech syllables can be corrected by taking the phonetic annotation into account. The final syllabic structure then provides a solid framework for the contemplated prosodic segmentation and labeling.

6. ACKNOWLEDGEMENT

This research was performed with support of the Flemish Institute for the Promotion of the Scientific and Technological Research in the Industry (contract IWT/AUT/950056).

7. REFERENCES

[1] N. Campbell (1993), *Automatic detection of prosodic boundaries in speech*, Speech Communication 13, 343-354.

[2] N. Campbell (1995), *Prosodic influence on segmental quality*, Proc Eurospeech, 1011-1014.

[3] K. Demuynck, J. Duchateau and D. Van Compernolle (1996), *Reduced semi-continuous models for large vocabulary continuous speech recognition in Dutch*, Proc ICSLP, 2289-2292.

[4] J.-P. Martens and L. Depuydt (1991), *Broad phonetic classification and segmentation of continuous speech by means of neural networks and dynamic programming*, Speech Communication 10, 81-90.

[5] P. Mermelstein (1975), *Automatic segmentation of speech into syllabic units*, J. Acoust. Soc. Am., vol 58, no 4, 880-883.

[6] L. Van Immerseel and J.-P. Martens (1992), *Pitch and voiced/unvoiced determination with an auditory model*, J. Acoust. Soc. Am., vol 91, no 6, 3511-3526.

[7] A. Vorstermans, J.-P. Martens and B. Van Coile (1995), *Fast automatic segmentation and labeling: results on TIMIT and EUROM0*, Proc Eurospeech, 1397-1400.

[8] A. Vorstermans, J.-P. Martens and B. Van Coile (1996), *Automatic segmentation and labeling of multi-lingual speech data*, Speech Communication 19, 271-293.

[9] C. Wightman and M. Ostendorf (1991), *Automatic recognition of prosodic phrases*, Proc ICASSP, 321-324.

[10] C. Wightman and M. Ostendorf (1994), *Automatic labeling of prosodic patterns*, IEEE Transactions on Speech and Audio Processing, vol 2, no 4, 469-481.