

# Coping with disfluencies in Spontaneous Speech Recognition

Frederik Stouten and Jean-Pierre Martens

ELIS-Ghent University  
Sint-Pietersnieuwstraat 41, B-9000 Ghent, Belgium  
{fstouten,martens}@elis.ugent.be

## Abstract

Nowadays, automatic speech recognizers have become quite good in recognizing well prepared fluent speech (e.g. news readings). However, the recognition of spontaneous speech is still problematic. Some important reasons for this are that spontaneous speech is usually less articulated and contains a lot of disfluencies. In this paper, a new methodology for coping with disfluencies is presented and evaluated. The basic idea is to detect disfluencies and to determine the nature of these disfluencies prior to the recognition, and to use that information to control/modify the search. At present, the methodology has been elaborated for filled pauses (FP) and word repetitions (WR). It enables us to eliminate about one associated *normal word* error per disfluency without introducing a significant augmentation of the computational load.

## 1. Introduction

The most frequently occurring disfluencies in spontaneous speech are filled pauses (FPs). If one has a sufficiently large spontaneous speech corpus with FP transcriptions available, one can create dedicated acoustic FP models to supplement the traditional triphone model set. One can also introduce lexical FP models (pronunciation variants in the lexicon). In [7], a combination of acoustic and lexical FP models yielded a reduction of the word error rate (WER) of 7.8 % relative for medical dictation (with respect to a baseline system with one FP model in the lexicon).

Even when good FP models are available, FPs may still confuse the recognizer because they lead to word sequences (*n*-grams) that are not observed in the kind of written material that is normally used for language model (LM) development. Extending this material with spontaneous speech transcriptions would be another means of improving a spontaneous speech recognizer. However, the necessary resources to accomplish this for a general task are usually not available (and certainly not for Dutch). For a limited domain like medical dictation, the situation is different, and experiments by Pakhomov [4] showed that the technique can work: the WER could be reduced from 32.2 to 28.5 %.

A simple way of dealing with FPs at the level of the LM is to ignore the FP in the LM context of the next word. However, Stolcke and Shriberg [8] come to the surprising conclusion that discarding FPs from the trigram context actually increases the perplexity. However, they looked at the speech stretches that were supplied to the recognizer and that were delimited on the basis of acoustic criteria. The FPs occurring at sentence boundaries often appeared in the middle of such stretches. By only discarding sentence internal FPs the perplexity did decrease, as expected. In [5] the speech stretches corresponded to sentences and all FPs were sentence internal. For this material, the discarding strategy resulted in a 4 % decrease of the overall perplexity and a 30 % decrease of the perplexity of the word after an FP.

The problems with the discarding of FPs in the LM are that the notion of a sentence is not always clear in spontaneous speech, and that even when it is, the sentence boundaries only become available after the recognition (with FPs) has been accomplished.

Apart from FPs, there are other important disfluency types in spontaneous speech, such as the word repetition (WR). It is defined as an event consisting of a completed (not interrupted) word (W) and a repetition (R) of that word. Clearly, WRs do not require any new acoustic or lexical models. However, just like FPs, they may disturb the recognizer at the level of the language model.

In this paper we investigate adaptive search methodologies that aim at improving the recognition of spontaneous speech in the vicinity of FPs and WRs without affecting the performance in other regions. In section 2 we briefly outline the basic principles of the new adaptive approach and we further elaborate and evaluate it in sections 3 and 4.

## 2. An adaptive search methodology

The basic principle of our approach is to modify the search on the basis of disfluency information that is extracted from the speech by means of an acoustic front-end. The necessary conditions for this approach to be attractive are (1) that the extracted disfluency information is sufficiently accurate, (2) that the extraction is computationally attractive (limited number of operations per frame) and (3) that the delay it introduces (number of future frames the front-end must have processed before the search at the current frame can be completed) remains limited.

If we can develop a front-end that detects time intervals which are likely to correspond to a FPs, and W and R-parts of WRs, we can try to use these intervals for altering the behavior of the search. So far we have investigated the following strategies:

- 1. FP frame dropping**  
If the front-end detects a filled pause, the search engine can simply discard the frames in that time interval.
- 2. WR frame dropping**  
If the front-end detects a word repetition (W + R) the search engine can discard the frames in either the W or R time intervals.
- 3. FP probability adaptation**  
If at a certain moment the front-end detects the start of a FP, the search engine can locally raise the probability of entering an FP model, and lower that of entering another model. Several options are discussed in section 3.
- 4. WR probability adaptation**  
If at a certain moment the front-end detects a WR, the search engine can raise the probability of paths confirming this fact and lower that of other paths. A particular implementation of this is discussed in section 4.

The advantage of frame dropping is that it can also work in combination with a traditional recognizer not incorporating any disfluency handling strategy. Its disadvantage is of course that it may throw away frames corresponding to so-called *intended words*. The advantage of probability adaptation is that it can be applied in a graceful manner: one can control the degree of adaptation in a continuous way.

In the following two sections we investigate the effects of different disfluency processing techniques on the recognition of spontaneous speech. As in [7], we only consider the intended words and we derive the intended WER as the performance criterion.

### 3. Recognition experiments with filled pauses

All the experiments presented in this section are performed on a test set (46 min 40 sec long) from the Spoken Dutch Corpus (CGN) [1]. It is composed of speech from 9 male and 9 female speakers, and it contains 7041 intended words and 445 filled pauses. Hence, the FP-rate is 5.94 %.

We used a speech recognizer [2, 3] that was originally designed to recognize prepared continuous speech, but we extended it with a standard FP handling strategy. The lexicon had 40K entries and the language model was a trigram LM, trained on a 35M words corpus of newspaper articles. The acoustic models were trained on the Flemish read speech part of the CGN (38 h of speech from 150 different speakers).

In what follows, disfluency processing methods are simulated by rescoreing the word string hypotheses embedded in a word lattice emerging from the system as it was delivered to us. This system was labeled BS-FP : a baseline system without FP handling. The lattice depth is made sufficiently large so as to achieve that the simulations produce very much the same results as the real systems would.

#### 3.1. Baseline system with traditional FP handling (BS)

In order to create a more realistic baseline system (labeled BS), we have first of all extended the 40K lexicon with some lexical FP models. Since in Dutch, a FP is usually realized as a steady /@/ (schwa) optionally followed by an /m/, we have created 36 FP pronunciation variants (without probabilities) : @, @m, @@, @@m, @mm, @@@, ... More variants did not help anymore.

The language model probability  $P[FP]$  of entering one of the FP models is one of the free parameters of the baseline system. Other free parameters are the pruning threshold (THR) (margin in  $\log P$ ) and the beam width (BW) (maximum number of paths to keep at any time).

The performances of the baseline system for three (THR,BW) settings and 5 choices of  $P[FP]$  are shown in Table 1. Clearly, FP

THR/BW	BS-FP	BS as a function of $P[FP]$				
		0.081	0.057	0.036	0.025	0.018
40/5000	50.13	49.00	48.86	48.59	48.32	48.42
45/10000	47.65	46.73	46.60	46.66	46.54	46.31
50/20000	47.35	45.80	45.99	45.58	45.63	45.75

Table 1: WER (%) for the baseline systems BS-FP and BS. For BS, different values of  $P[FP]$  were tested.

modeling causes a significant reduction of the WER. The effects for one (THR,BW) setting are well illustrated by Figure 1.

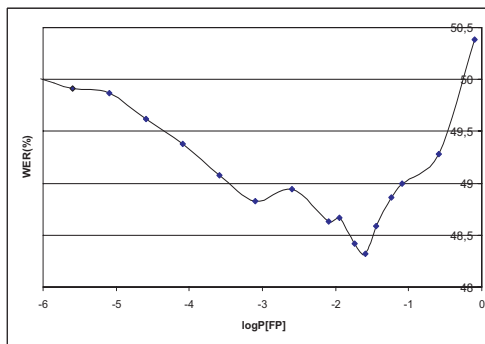


Figure 1: WER as a function of  $\log P[FP]$  (THR = 40, BW = 5000).

#### 3.2. Removal of FP from the LM context

A straightforward extension of the baseline system is obtained by discarding FPs in the computation of LM probabilities. Suppose that we have a word sequence: *op een uh stoel*. In the baseline system, the trigram probability of *stoel* is calculated as

$$\log P(\text{stoel}|\text{een, uh}) = \log P(\text{stoel}|\text{uh}) + \log D(\text{een, uh}) = -5.82$$

with  $D$  representing a bigram discount factor. Similarly, the word sequence *op uh een stoel* yields

$$\log P(\text{stoel}|\text{uh, een}) = \log P(\text{stoel}|\text{een}) + \log D(\text{uh, een}) = -4.18$$

Both probabilities are significantly lower than the -3.23 that follows from the clean word sequence *op een stoel*.

For three values of  $P[FP]$  we have tested a system that discards FPs in the LM probability computations. The results of this system (labeled BS+LM) are listed in Table 2. There is only a statistically

THR/BW	BS+LM as a function of $P[FP]$		
	0.057	0.036	0.025
40/5000	47.91	47.92	47.71
45/10000	45.70	45.93	45.85
50/20000	45.28	45.11	45.25

Table 2: WER (%) of a system that discards FPs in the LM context.

insignificant drop of the WER, which is in line with [8].

#### 3.3. FP frame dropping

In [9] we proposed a system for the detection of filled pause intervals prior to the recognition. It was based on an initial phone-like segmentation of the utterance, an acoustic/prosodic characterization of the segments, an estimation of the posterior probabilities  $P(FP|x)$  and a merging of consecutive FP segments to FP intervals (see [9] for more details). This FP detector is able to detect 74 % of the FPs with a precision of 83 %, and it has been used in [9] to show that frame dropping can reduce the WER. In this paper this is confirmed by more elaborated experiments with a better baseline system.

Applying FP frame dropping means that the search engine is to ignore all frames falling inside a FP interval. Since frame dropping can also be applied in combination with a speech recognizer without any FP modeling, we have performed tests on BS-FP as well as on BS+LM. Moreover, to assess the quality of our present FP detector we have also measured the results obtained by dropping the frames falling in the reference FP intervals emerging from the word

segmentations that are provided with the CGN data. The figures in Table 3 demonstrate that dropping the frames of the reference

THR/BW	BS-FP + drop		BS+LM + drop	
	REF	DET	REF	DET
40/5000	45.49	48.86	46.20	47.96
45/10000	43.53	46.64	43.32	45.41
50/20000	43.02	46.06	42.59	44.88

Table 3: WER (%) obtained by systems that ignore FP frames in the acoustic input. Both the removal of reference (REF) and detected (DET) FP frames is investigated.

FP intervals yields a significant reduction of the WER. Moreover, when dropping these frames one does not need the traditional FP handling, as expected. Unfortunately, dropping the frames of the detected FP intervals yields but a small improvement with respect to the traditional FP handling. In order to make FP frame dropping more attractive technique, one needs a better FP detector.

### 3.4. FP probability adaptation

In this section we investigate the potential of using the FP detector output to control (in a continuous way) the probabilities of entering a FP arc of the word lattice.

First we raise the LM probability of a FP arc when more than 50 % of the frames consumed by this arc fall inside a detected FP interval. The probabilities of all other arcs are left unaltered. Two different methods were tested : M1 in which the LM probability of the FP arc was modified to 1, and M2 in which it set to the mean  $P(FP|x)$  of the frames consumed by this arc. From Tables 4 and 3

THR/BW	BS+LM	M1-REF	M1-DET	M2-DET
40/5000	47.71	47.05	47.19	47.21
45/10000	45.85	44.64	45.11	45.11
50/20000	45.25	43.77	44.41	44.50

Table 4: WER (%) after adaptation of the FP model entering probability on the basis of reference and detected FP intervals.

one can see that with detected FP intervals, M1 and M2 are equivalent, and probability adaptation seems to outperform frame dropping. When reference FP intervals are used however, frame dropping seems to be the better option. (Note that in the case of REF only method M1 is relevant.) This probably means that we should further improve our probability adaptation strategy.

Before trying to, we have tested a strategy (M3) that is similar to M1, but easier to integrate in a time-synchronous search. If the search is at time  $t$  and if the FP-detector produces some FP interval starting in  $[t - D, t + D]$  (with  $D$  being a predefined length), then the probability for entering an FP model is raised to 1. For  $D = 0.1$  sec, method M3 provides the same results as M1.

In a next experiment we combined setting the FP log-probability to 1 (M3) with lowering the non-FP log-probabilities by an amount  $\Delta \log P$ . However (see Table 5) this did not cause any additional improvement.

The relative improvements of the WER are very modest, but by looking at the number of intended words being corrected per FP, we get quite a different picture. In Table 6 we have listed the overall and the per FP improvements of the different methods with respect to system BS-FP. Apparently, the best system (discard FP in LM and adapt LM) can correct 1.04 intended words per FP, the traditional FP handling strategy only 0.75. The maximal improvement (with a

THR/BW	M4 with $\Delta \log P$				M3
	0.5	1.5	2.5	10.0	
40/5000	47.31	47.31	47.34	47.37	47.31
45/10000	45.18	45.16	45.19	45.22	45.24
50/20000	44.51	44.47	44.45	44.45	44.58

Table 5: WER (%) with manipulation of both the FP and non-FP model probabilities.

discard FP in LM	drop FP frames	adapt LM (M1)	RI (%)	IWC/FP
no	no	no	3.63	0.61
yes	no	no	4.43	0.75
yes	no	yes (DET)	6.21	1.04
yes	no	yes (REF)	7.56	1.27
no	yes (REF)	no	9.14	1.54
no	yes (DET)	no	2.72	0.46
yes	yes (REF)	no	10.0	1.69
yes	yes (DET)	no	5.21	0.88

Table 6: Relative improvement (RI) and intended words corrected per FP (IWC/FP) for the different methods (w.r.t. BS-FP).

perfect FP detector) is estimated to be as large as 1.7 intended words per FP.

## 4. Recognition experiments with repetitions

Another type of disfluencies is the word repetition (WR). In a CGN subset of 12h 38min (130k words) of spontaneous speech we found 965 WRs, excluding intended identical word pairs of course. In 95 cases (= 10 % of the WRs) the same word was repeated twice. Figure 2 shows the top-15 of repeated words. This top-15 consists of very frequent monosyllabic function words, and it covers more than 70 % of all WRs. That not so many long words are repeated is due to the fact that they are usually interrupted before being repeated. Technically, this is a word interruption and no WR.

In our material, the WR rate, defined as the number of WRs per intended word, is only 0.8 % and thus much smaller than the FP rate. We therefore investigate first whether WR processing can have a significant impact on the WER. For that purpose we assume to have knowledge of the correct W and R intervals. We also perform our tests on a selected test set of 100 spontaneous sentences (1388 words) with a lot (116) of WRs (and only 7 FPs).

### 4.1. Acoustic/prosodic features of WRs

From the word segmentations delivered with the CGN data, it followed that 408 out of 965 WRs have a silence between the word (W) and its repetition (R) and 22 others have a filled pause between the two. The low number of repetitions combined with a FP reveals that repetitions and filled pauses are two different (person dependent) disfluency strategies. An interesting property of WRs is that W tends to be longer than R. Our findings for Dutch confirm the data of [6] for American English.

### 4.2. WR frame dropping

In analogy with FP frame dropping, the search engine can ignore all frames falling inside a W or R interval. By dropping the frames of the longest of these two intervals, a relative improvement of 5.3 %

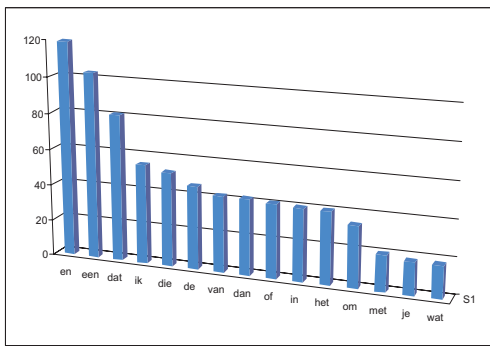


Figure 2: Most frequently repeated words in Dutch.

system	WER (%)	RI (%)	IWC/WR
BS	50.07	–	–
BS+Rdrop	48.03	-4.07 %	0.48
BS+Wdrop	47.80	-4.53 %	0.54
BS+Ldrop	47.41	-5.31 %	0.63
BS+Rdiscard	47.05	-6.03 %	0.72
BS+LM	45.75	-8.62 %	1.03

Table 7: WER (%), relative improvement and number of intended words corrected per WR due to different WR handling strategies. The search parameters were  $THR = 50$  and  $BW = 20000$ .

is attained (see Table 7). This corresponds to only 0.63 intended word corrections per WR.

#### 4.3. Discarding R from the LM context

For handling WRs at the level of the LM we first replace the LM probability of the repeated word by some constant  $P_o$  as to indicate that there is a repetition. Then we also discard the repetition in the LM context of the next word. By doing this, we can gain about 6 % relative with respect to the baseline (see Table 7, BS+Rdiscard) for  $P_o$  values between 0.03 and 0.06.

#### 4.4. WR probability adaptation

Finally, we also investigated whether changing the LM probabilities on the basis of WR detections could offer an additional improvement. The LM-probability of entering the same word as before was set to 1 when more than 50 % of the frames consumed by this repeated word arc belong to an R interval and more than 50 % of the previous word fell inside a W interval. If this condition is not met, the probability for re-entering the word is set to  $P_o$  which is expected to be smaller than the optimal  $P_o$  that is used in the previous section. In addition, the LM probability of the word following the repeated word is computed by discarding the repeated word in the LM context. By doing this, a gain of 8.62 % relative with respect to the baseline is attained for  $P_o$  in the range 0.005 to 0.01. This gain represents 1.03 intended word corrections per WR. Surprisingly, this is smaller than the 1.7 which we found for FPs.

Keeping in mind the low frequency of occurrence of word repetitions, the figures in Table 7 do not justify any large effort in search of a good WR detector. Maybe by considering word repetitions and word abbreviations as two forms of the same phenomenon, we may be able to obtain more significant improvements by the handling of that (more frequently occurring) phenomenon.

## 5. Conclusions

The experiments on recognition of spontaneous speech with filled pauses clearly show the benefit of embedding FP handling strategies in the recognizer. Moreover, it is shown that continuously adapting the LM probabilities on the basis of the outputs of an FP detector can induce a larger gain than just a priori optimizing the LM with respect to the possible occurrence of FPs. It is also shown that further improvements of the FP detector could increase the actual performance gain from the actual 1 intended word corrections per FP to maximally 1.7 corrections per FP.

The preliminary experiments on recognition of spontaneous speech with word repetitions show that none of the investigated methods can yield a sufficient improvement of the WER.

## 6. Acknowledgements

This research was supported by the Flemish Institute for the Promotion of the Scientific and Technical Research in the Industry (contract ADV/STWW/000151). The ESAT speech group is acknowledged for letting us use its speech recognition engine.

## 7. References

- [1] Goedertier W., Goddijn S., Martens J.-P., “Orthographic Transcription of the Spoken Dutch Corpus”, Procs LREC, 2000, pp 909–914.
- [2] Demuynck K., Duchateau J., Van Compernelle D. and Wambacq P., “An efficient search space representation for large vocabulary continuous speech recognition”, Speech Communication, Vol. 30, number 1, January 2000, pp 37–53.
- [3] Duchateau J., Demuynck K. and Van Compernelle D., “Fast and accurate acoustic modelling with semi-continuous HMMs”, Speech Communication, Vol. 24, number 1, April 1998, pp 5–17.
- [4] Pakhomov S. V., “Modeling filled pauses in medical dictations”, Procs Association for Computational Linguistics (ACL), 1999, pp 619–624.
- [5] Peters J., “LM Studies on filled pauses in spontaneous medical dictation”, Procs HLT-NAACL, 2003, pp 82–84.
- [6] Plauché M. C. and Shriberg E., “Data-driven subclassification of disfluent repetitions based on prosodic features”, Proc. of Int. Congress of Phonetic Sciences, Vol 2, 1999, pp 1513–1516.
- [7] Schramm H., Aubert X. L., Meyer C. and Peters J., “Filled-Pause modeling for medical transcriptions”, Procs ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition, 2003, paper TM06.
- [8] Stolcke, A. and Shriberg, E., “Statistical language modeling for speech disfluencies”, Procs ICASSP, Vol. 1, 1996, pp 405–408.
- [9] Stouten F. and Martens J.-P., “A Feature-Based Filled Pause Detection System for Dutch”, Procs Workshop for Automatic Speech Recognition and Understanding, 2003, pp 309–314.
- [10] Shriberg E. and Stolcke A., “Word predictability after hesitations: A corpus based study”, Procs ICSLP, Vol. 3, 1996, pp 1868–1871.
- [11] Siu M.-H and Ostendorf M., “Modeling disfluencies in conversational speech”, Procs ICSLP, Vol. 1, 1996, pp 386–389.