Filled Pause Modeling

Ricardo Nunes & Luís Neves
L^{2}F – Spoken Language Systems Laboratory
INESC ID Lisboa, Rua Alves Redol 9, 1000-029, Lisboa, Portugal
{rjfn,lmln}@l2f.inesc-id.pt

This document presents a streamlined approach to modeling filled pause distribution in spontaneous speech and populating a large clean *corpus*, making use of only the SRILM toolkit and a small training set. Although used for filled pause modeling, it can be fairly general and may be used to model other types of disfluencies, punctuation or sentence boundaries, with a minimal set of changes.

August 8, 2006

Table of Contents

List of Tables

List of Figures

1 Introduction

The motivation for this work resided in the need to build an automatic speech recognition (ASR) system that handled typical spontaneous speech disfluencies, more precisely filled pauses.

Filled pauses are hesitation sounds that speakers employ to indicate uncertainty or to maintain control of a conversation while thinking of what to say next. They can occur anywhere in the stream of speech, although there are some systematic distributions and well-defined functions in the discourse. For European Portuguese, the most common filled pauses were transcribed as mm (when the sound is a nasal murmur), aam (when there is evidence either of a nasal vowel or a vowel followed by a nasal murmur), and aa (when the filled pause corresponds to a non-nasal vowel, close to the mid-center vowel). The most common filled pause in our *corpus* of lectures is by far aa, very similar to the Portuguese article and preposition "a", one of the top 5 most frequent words in Portuguese [1].

This type of disfluency is not correctly modeled in our ASR trained for Broadcast News (BN), because most of the elements that constitute the language model training *corpus* are written documents, namely newspapers. These documents do not have any kind of disfluency markers since there are hardly any spontaneous speech transcriptions in them. The frequent usage of filled pauses in spontaneous speech along with the fact that the ASR is not prepared to handle this type of disfluencies, causes systematic errors in the recognition result. Furthermore error bursts, that are sequences of wrongly recognized words, are fairly frequent and usually occur in a 3-9 words sequence, which means that a wrongly recognized filled pause may be responsible for errors in the neighbouring words. This situation has the consequence of drastically increasing the word error rate (WER) of the ASR for spontaneous speech.

In an attempt to circumvent this problem, one may artificially populate the training newspaper *corpus* with filled pauses in order to make it more similar to spontaneous speech. This task is accomplished by inserting filled pause markers in the newspapers texts, using stochastic n-gram models similar to the ones used in the ASR (models that predict the word spoken based on the preceding words). The only difference between the two processes is that the ASR models predict the current spoken word, whereas the filled pause models predict if the current location is appropriate for inserting a filled pause. In order to predict the probability values for the insertion of filled pauses after a certain word sequence, one needs a training set that is representative of their location, obtained from manual transcriptions of spontaneous speech.

The set of tools used to build language models and to populate the texts with filled pauses based on those models are included in the Stanford Research Institute Language Modeling Toolkit (SRILM) version 1.4.5.

In the second chapter we present a detailed description of the several *corpora* that were used for language model training and testing. The third chapter describes the process and tools that were used to insert filled pauses in the texts. The two last chapters present the results, along with a discussion of how to improve them.

2 Corpora Collection

2.1 Training Corpus

In order to obtain a more robust training language model we have included various sources of transcribed material. Our main source was the corpus of Broadcast News transcriptions built in the scope of the ALERT project, but we have also included a *corpus* of lecture transcriptions obtained in the scope of the LECTRA project (corresponding to the "Production of Multimedia Contents" (PMC) course), a *corpus* of oral interviews collected by the Center of Linguistics of the University of Lisbon (CLUL) entitled "Fundamental Portuguese" (FP) and the *corpus* of transcriptions of dialogues on a pre-defined domain (map task) built in the scope of the CORAL project . Overall, the *corpus* has 769k words, corresponding to many different speakers, and addressing a wide variety of situations (BN, lectures, chats between friends, map directions).

	ALERT	**LECTRA-PMC**	**CLUL-FP**	**CORAL**
Words	517 047	25 778	166 368	59 836
Sentences	33 236	1 160	12 746	11 809
Filled Pauses	5 937	904	3 784	928

Table 1. Constitution of the training *corpus.*

It is worth noting that the main part of this training *corpus* derives from BN transcriptions, where many segments correspond to read speech, spoken by professional anchors. This justifies the relative low frequency of filled pauses in this *subcorpus* (1.1%), which contrasts with the high percentage of filled pauses in the lecture corpus (3.5%). Unfortunately there is a very limited number of transcribed material of this type available. For the other subcorpora, we obtained intermediate values, averaging 1.5% for all the training corpus.

2.2 Evaluation Corpus

For the target of the filled pauses insertion process we chose a newspapers *corpus* corresponding to 9 Portuguese newspapers: "A Bola", "O Jogo", "Público", "Expresso", "Diário Económico", "O Independente", "Jornal de Notícias", "Diário de Notícias", "Expresso Diário", collected over several years. Altogether, there were 45 files, each corresponding to a one-year collection of a single newspaper. Globally there are approximately 578 million words (1.2 GB) without any type of punctuation, with the exception of the tags <s> and </s> signaling the start and end of each sentence.

3 Process Description

The process for the insertion of filled pauses uses the information contained in a language model built entirely from manual transcriptions to replicate that information on a larger *corpus* but less representative of spontaneous speech.

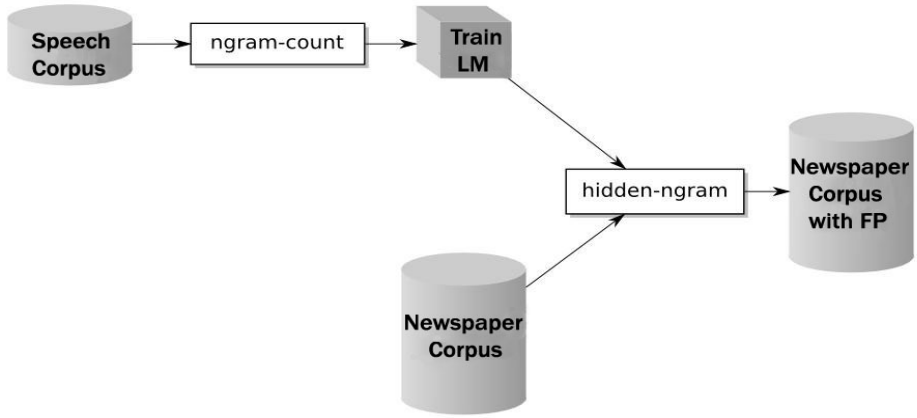

Figure 1. Overall schematic of the filled pause insertion process

The entire process can be summarized in these three steps:

a) Formatting the training *corpus*

Filled pauses were encoded differently in the *subcorpora* that were used for training. Hence, the first task consisted of formatting these *subcorpora* using characters for representing filled pauses that were well supported by the SRILM toolkit. Thus the 3 most common filled pauses were represented by the characters @aa, @aam and @mm. The orthographic annotation conventions were also significantly different in the 4 *subcorpora*. Hence this formatting task also eliminated all types of unpronounced punctuation and other annotation tags denoting broken forms, repetitions, contractions, specific pronunciations, etc.

b) Building the language model

This stage was done using the ngram-count program enclosed in the SRILM toolkit. This program can be used with multiple options. In our first experiments, the options were set so as to build a language model of the 4th order, without effecting cuts in any order and using the modified Knesser-Ney smoothing and back-off distribution method:

```
ngram-count   –text <Speech Corpus>
              –order 4
              –gt1min 0 –gt2min 0 gt3min 0 –gt4min 0
              –kndiscount1 –kndiscount2 –kndiscount3  –kndiscount4
              –lm <Training LM>
              –read–with–mincounts
              –meta–tag __tag__
              –debug 2
```

In later experiments, we modified these options in order to achieve a higher percentage of filled pause insertion, closer to the one that is verified in the training *corpus*. Thus the Good-Turing discounting method was used (given that it as a lesser effect on the n-grams with a higher number of counts) and the n-grams that contained filled pauses were multiplied by a factor of 50, in order to increase their relative frequency.

Filled pause insertion

Using the hidden-ngram program (also enclosed in the SRILM toolkit), we performed the insertion of filled pauses in the newspaper texts, repeating the command for each of the 45 files. The program takes as input arguments the previously constructed language model, the newspaper texts file and a file containing the vocabulary that is intended to be inserted. In this case this FP file contains the three chosen filled pauses (one word per line).

```
hidden-ngram –lm <Training LM>
              –text <Newspaper Corpus>
              –hidden–vocab <FP file>
              –keep–unk
              –order 4        >  <Newspaper Corpus with FP >
```

As a result, we obtained the 45 files populated with filled pauses.

4 Results

The results of populating the newspaper *corpus* with filled pauses are shown in Table 2. The first row (Without FP) shows the initial contents of the newspaper *corpus*. The second row shows the modified contents after filled pause insertion (With FP). Finally, the third row shows the modified contents after implementing the modifications that aimed at increasing the probability of the n-grams with filled pauses, described in the previous chapter (WithxFP).

	Words	**@aa**	**@aam**	**@mm**	**Total**	**%**
Without FP	578 M	0	0	0	0	0
With FP	578 M	451 521	3 747	5 029	460 297	0.08
WithxFP	587 M	8 600 001	221 655	390 148	9 211 804	1.55

Table 2. Contents of the newspaper corpus before and after filled pause insertion.

As can be verified the automatic process introduces a lower percentage of filled pauses in comparison to the average of the training set (1.5%). Modifying the counts and the discount method allowed us to obtain a similar percentage, maintaining grossly the same relation between all types of filled pauses.

The original and modified newspaper *corpora* were used in recognition experiments in order to test the potential advantages of filled pause insertion. As an evaluation set we selected a lecture of the PMC *corpus* that had not been used for training (12-1). Each of the language models built with the 3 newspaper corpora was interpolated with the manual transcriptions of the BN and the training material of PMC. For further information on the construction processes/adaptation of language models and recognition process see [3] and [4]. The recognition results are shown in Table 3.

	TOTAL	CORR	SUB	DEL	INS	WER %
WithoutFP	6626	3820	1560	846	245	42.6
WithFP	6226	3835	1544	847	244	42.3
WithxFP	6626	3896	1501	829	257	41.6

Table 3. Overall recognition results

The introduction of filled pauses led to a very small reduction in the WER. As predicted, the quantity of filled pauses inserted by the program was clearly insufficient to correctly model their distributions in the discourse. Only adding some relevance to the ngram counts containing filled pauses were we able to obtain some significant results.

The PMC lecture chosen as evaluation set has 165 marked filled pauses. The majority are of the @aa type, that is most common in the Portuguese language [1]. Tables 4, 5 and 6 show the number of hits, misses and false alarms in the detection of filled pauses in this lecture, for each of the 3 scenarios (WithoutFP, WithFP, WithxFP, respectively).

Although there are some filled pauses represented in the language model (through the

interpolation with the BN *corpus*), they have associated very small probabilities, as can be verified by the low number of detection and false alarms in the first scenario.

	Reference	Hypothesis		
		Hit	Miss	False Alarms
@aa	156	13	143	7
@aam	9	0	9	0
@mm	0	0	0	0

Table 4. Filled pauses detection results in the WithoutFP scenario .

	Reference	Hypothesis		
		Hit	Miss	False Alarms
@aa	156	34	122	9
@aam	9	0	9	0
@mm	0	0	0	0

Table 5. Filled pauses detection results in the WithFP scenario.

	Reference	Hypothesis		
		Hit	Miss	False Alarms
@aa	156	75	81	50
@aam	9	2	7	0
@mm	0	0	0	0

Table 6. Filled pauses detection results in the WithxFP scenario.

After the filled pause insertion process, we obtain a small increase in the number of detections, and the number of false alarms remains practically unchanged. This indicates an improvement towards the goal of this work, although these changes were not reflected in the final WER.

One should also point out that no @aam filled pause was detected. On one hand, its phonetic contents should lead to an easier detection, but on the other hand, none of the filled pauses enclosed in the BN *corpus* were of that specific type.

The modifications introduced in the insertion process resulted in a significant increase in the number of detections, although the number of false alarms also increased. Nevertheless, a 50% detection rate was almost achieved, together with an improvement in the detection of the second type of filled pause, indicating that they had very low relative frequencies.

Another source of errors is the fact that prior to the automatic speech recognition stage, the system includes an automatic segmentation stage whose goal is to exclude segments that do not contain any form of speech. This segmentation is based on the analysis of features such as dynamism and entropy. Due to the fact that filled pauses are characterized by a relative statitonarity of pitch and formant contours, some segments containing filled pauses (specially the long ones surrounded by pauses) are automatically discarded.

5 Discussion and Conclusions

This study explored a streamlined approach into modeling an important phenomena associated with spontaneous speech. This type of approach can be fairly general allowing not only the modeling of filled pauses, but also other elements such as sentence boundaries, punctuation, repetitions, etc. This can be done simply by changing the vocabulary specified in section 3 (<FP file>) and possibly with some preprocessing of the training material.

There are other types of approach for dealing with this kind of problems, such as neural networks or iterative processes based on the maximum likelihood method. Although offering more flexibility (allowing not only to model the preceding words, but also the suceeding words, word classes, etc.), they require considerably more resources. The adopted process, described in this document, took an overall time of three hours to process a 1.2GB *corpora*, with 578 million words.

The best results obtained in this work led to a very small reduction in WER (minus 1% absolute score), although the increase in the number of counts associated with filled pauses was totally artificial. The main relevance of these tests was in showing that there are problems with the modeling the correct localization of filled pauses, based solely on the preceeding words. Also if we only consider sequences of words, we obtain a wide variety of n-grams, which lead to very low probabilities, and therefore, to very few inserted filled pauses.

In order to increase the effectiveness of the language model, class n-grams could be adopted instead of word n-grams, using the PoS class of each word. This is one of the approaches we are currently considering, together with the expansion of the manualy transcribed spontaneous speech corpora.

6 References

[1] Trancoso, I., Nunes, R. and Neves L., "*Classroom Lecture Recognition*", Proc PROPOR 7th Int.Workshop on Computational Processing of the Portuguese Language, Brazil, 2006.

[2] Trancoso, I., Nunes, R., Neves L., Viana, C., Moniz., Helena, Caseiro and D., Mata, A. I., "*Automatic Speech Recognition of Classroom Lectures*", Proc. In INTERSPEECH, Pittsburgh USA, 2006.

[3] Nunes, R. and Neves L., "Tanscrição de aulas para e-learning" - TFC Report 101/2005/L

[4] SRILM manual pages "http://www.speech.sri.com/projects/srilm/manpages/"

[5] Stolcke, A., Shriberg, E., "Statistical Language Modeling for Speech Disfluencies", *Proc. IEEE ICASSP* , Atlanta, GA 1996.