Filled pauses in speech synthesis: towards conversational speech.

Jordi Adell[1], Antonio Bonafonte[1], and David Escudero[2]

[1] Universitat Politècnica de Catalunya, Barcelona 08034, Spain,
{jadell,antonio}@gps.tsc.upc.edu,http://www.talp.upc.edu
[2] Universidad de Valladolid, 47011 Valladolid ,Spain,
descuder@infor.uva.es,http://www.infor.uva.es

Abstract. Speech synthesis techniques have already reached a high level of naturalness. However, they are often evaluated on text reading tasks. New applications will request for conversational speech instead and disfluencies are crucial in such a style. The present paper presents a system to predict filled pauses and synthesise them. Objective results show that they can be inserted with 96% precision and 58% recall. Perceptual results even shown that its insertion increases naturalness of synthetic speech.

1 Introduction

Speech synthesis has already reached high naturalness, mainly due to the use of effective techniques such us unit selection-based systems [1] or other new rising technologies [2] based on the analysis of huge speech corpora. The main application of speech synthesis has been focused by now on read style speech as it can be assessed that read style is the most generalist style to be extrapolated to any other situation. But nowadays and future applications of text to speech (TTS) systems like film dubbing, robotics, dialogue systems, or multilingual broadcasting demand a variety of styles as the users expect the interface to do more than just reading information.

If synthetic voices want to be integrated in future technology, they must simulate the way people talk instead the way people read. This objective has been already tackled in several manners such as emotional speech synthesis [3], voice quality modelling [4] or even pronunciation variants [5]. In our opinion style is more important; it is desirable synthetic speech to be more conversational-like rather than reading-like speech. Therefore, we claim it is necessary to move from *reading* to *talking* speech synthesisers.

Both styles differ significantly from each other due to the inclusion of a set of a variety of prosodic resources affecting to the rhythm of the utterances. Disfluencies are one of these resources defined as phenomena that interrupt the flow of speech and do not add propositional content to an utterance [6]. Disfluencies are very frequent in normal speech [7] so that it is possible to hypothesise the need to include these prosodic events to approximate to talking speech synthesis. We have already presented experiences on synthesising disfluencies in TTS systems in previous works [8], now we present a work to predict where the disfluencies must be placed in the text.

In the goal of integrating disfluencies in conversational speech synthesis, it is not only important that the system is able to pronounce disfluencies, but also to give the system capabilities to predict them. Previous experiences in pauses prediction [9] lead us to give to this process a relevant role because wrong predictions can decrease dramatically the quality of the synthetic voices. Due to the complexity of the disfluencies phenomenon we focus on the simplest and more frequent type of disfluency: filled pauses. Filled pauses, in contrast with other disfluencies like repairs or repetitions, are easy to synthesise and permit the use of similar algorithms as the ones already used to predict pauses.

First, in Section 2 the framework is set, both the prediction algorithm and the corpus are described. Next section explains the synthesis method used for filled pauses. Finally, Section 4 presents objective and perceptual evaluation results and in Section 5 conclusions are discussed.

2 Prediction of Filled Pauses

In the framework of disfluent speech synthesis, the synthesis of filled pauses is essential. In some applications, such as dialogue systems, filled pauses position could already be given to the synthesiser. However, in some other applications such as speech translation, text is given in spoken style but with no disfluencies in it. Is in this kind of applications where filled pauses prediction makes its contribution. Filled pauses can be inserted, for example, in between the two utterances of a repetition: *no, **uh** no I won't go!* or after an acknowledge phrase: *well, **uh** I will go!*. The proposed approach to this task can be carried on due to the increasing availability of large corpora. Mainly for recognition purposes, many spoken corpora are currently being build. Machine learning techniques can take advantage from this corpora in order to perform tasks as the one proposed here.

2.1 Classifier description

The algorithm presented here is based on a combination of language modelling plus a decision tree. Both are well known machine learning techniques. The focus of the work done is on choosing proper features with respect to the task.

The decision tree performs a binary classification task. It classifies each word in the text whether it has a filled pause following it or not. The aim of the algorithm is to decide whether a filled pause has to be inserted or not after each word present in the text. The decision tree does the classification based on feature vectors which mainly contain language model probabilities and POS tags.

First of all, a language model is trained [10]. Afterwards, the text is tagged with Part-of-Speech (POS) labels [11]. Using the tagged text and the language model a feature vector is generated for each word. A special tag is used as sentence beginning and is treated as a word in order to allow insertions at beginning of sentences.

2.2 Feature sets

Several sets of features have been tested in order to see which one is better suited for the filled pause insertion task. Three classes of features have been used: text-based, ngram-based and a disfluency-based one. Text-based features are: w_i and pos_i, where w_i is the actual word to which is applied the decision tree, pos_i is the Part-of-Speech of word w_i. Ngram-based features are: $p(w_i|h_i)$ and $p(FP|h_{i+1})$ where $p(w_i|h_i)$ is the probability of word w_i given the history h_i and the language model, and $p(FP|h_i)$ is the probability of a filled pause given previous history h_i. Finally, the disfluency-based feature is binary and indicates whether the filled pause is to be inserted in between the two utterances of a repetition or not and will be referred as *repeat*. Repetitions of one and two words have been considered. The four sets used in the experiments are:

- *Set1*: w_i, pos_i, pos_{i-1}, pos_{i+1}, $p(w_i|h_i)$, $p(w_{i+1}|h_{i+1})$ and $p(FP|h_i)$;
- *Set2*: w_i, pos_{i-2}, pos_{i-1}, pos_i, pos_{i+1}, pos_{i+2}, $p(w_i|h_i)$, $p(w_{i+1}|h_{i+1})$, $p(w_{i+2}|h_{i+2})$ and $p(FP|h_i)$;
- *Set3*: w_i, *repeat*, pos_i, pos_{i-1}, pos_{i+1}, $p(w_i|h_i)$, $p(w_{i+1}|h_{i+1})$ and $p(FP|h_i)$;
- *Set4*: pos_i, pos_{i-1}, pos_{i+1}, $p(w_i|h_i)$, $p(w_{i+1}|h_{i+1})$ and $p(FP|h_i)$;

Ngram-based features have been chosen based on the work of [12], where they claim that filled pauses can be used by the listener in order to identify whether what is going to be said afterwards will be hard to understand or not. Since a filled pause mainly points out speaker problems on finding the desired words, it suggests to the listener that following words will be hard to understand because they have been hard to produce. This reasoning has lead us to add language model probabilities as features. Lower probabilities of the actual and next word will indicate an uncommon expression and therefore it might have been difficult to generate.

POS values have been used here, in order to give the machine learning technique (i.e. decision tree) the possibility to evaluated syntactic structures. Different context lengths have been tested too. Shorter one in Set number 1 and larger one in Set number 2.

Filled pauses can also be in the editing phase of a disfluency [13]. This is the reason for including the feature *repeat* in feature set number 3.

The inclusion of the word itself makes the algorithm very slow since all questions about each possible word has to be tested. In order to avoid this practical problem, we have added to our system the concept of *candidate*. A candidate is a word that allows a filled pause to be placed after it. Therefore, no filled pauses will be placed after words that are not considered candidates. The set of candidates is chosen by sorting words in the training data set. The number of times they preceded a filled pause is used as sorting criteria. This list is then truncated and top most words are considered candidates. Moreover, few candidates are enough to consider a high number of filled pauses. For example, ten candidates can cover 53% of filled pauses in the corpus used in this paper.

In order to test whether the candidates-based approach leads to an improvement, the last data-set has been added to the experiments. *Set4* is equal to *Set1* but without feature w_i, only pos_i is known from words preceding a filled pause.

2.3 Corpora

The algorithm presented in Section 2.1 is data-driven. Therefore, corpora is needed to train it. The corpus used in the present work has been collected within the LC-STAR project. It consists on lab recorded "spontaneous" conversations about the tourist information topic. Several volunteers were asked to perform a conversation over a telephone playing several roles such as a costumer of a hotel. No guidance on the specific subject was given. It is highly spontaneous, it contains 317,000 words. In addition to filled pauses, it contains disfluencies such as restarts or repetitions. It has been manually transcribed at word level mainly for recognition purposes.

It is a Spanish corpus of 64 speakers. It contains 317,000 words and 5,700 filled pauses (i.e. 1.8%). The vocaulary size is 11,000 words and 1,173 out of them preceded a filled pause at leas once.

3 Rhythm and Synthesis of Filled Pauses

In previous works [8] we implemented a set of rules to predict duration and F0 contours of filled pauses and repetitions. Those rules where expected to be useful for the unit selection TTS system in order to look up the corresponding units in the inventory to be concatenated to compose the final sequence. As far as it concerns to filled pauses, the main result of the study was to show that filled pauses are very stable in duration (we decided to use a constant value) and in frequency (the lowered mean value of the preceding and the following word). But results where not satisfactory in terms of the quality of the synthetic output of the TTS system because the rules implemented did not consider several important aspects concerning to the rhythm imposed by the disfluencies that must be taken into account.

In this section we present a study based on a corpus of 65 sentences recorded from a male speaker specially for disfluent speech synthesis. We differentiate here between the rhythm of the whole sentence (rhythTot), the rhythm preceding the filler pause(rhythPre), the one following the filled pause (rhythPost) and the duration of the filled pause itself. The mean syllable duration is used as a measure of the rhythm. Table 1 shows values for the 48 filled pauses that appear in the corpus.

As can be observed in Table 1 there are no significant differences ($P > 0.05$) between the rhythm of the whole sentence, the preceding and following to the filled pause. However, the filled pause duration is significantly higher than the rest ($P < 0.05$). Therefore, we can conclude that the insertions of filled pauses does not modify the rhythm of the utterance. Also the duration of the syllable just before the filled pause has been studied. As can be observed again in Table 1 there are significant differences between the sentence rhythm and this syllable length. This result lead us to conclude that it exists a lengthening of the syllable immediately before the filled pause with respect to the global rhythm of the sentence.

	FPdur	*rhythmTot*	*rhythmPre*	*rhythmPost*	*lastSyl*
Average	**320**	190	180	190	**360**
Standard Deviation	160	30	50	40	50

Table 1. Study of filled pauses rhythm, calculated over the 48 realisations that appear in the corpus. Units are *ms*.

As the TTS system does not include these considerations, at the moment we decided to enter disfluencies manually in a pool of sentences to test the benefits of the inclusion of filled paused on generating more expressive synthetic speech. The procedure followed was first to generate free of filled pauses synthetic speech and then to insert the filled pause in the place indicated by the algorithm explained in previous sections. We select the disfluency to be inserted (emm, uhh) from a list of them taking also into account its F0 values and the rules devised in [8]. By using the PSOLA features of the `praat` program [14] the duration of the syllables and the duration of filled pauses are adapted to the new situation. These modifications are done following conclusions presented in this section.

4 Evaluation

4.1 Prediction.

In order to evaluate the prediction algorithm, the corpus (see Section 2.3) has been split into three sets: training, development and test sets. Which consisted on 80%, 10% and 10% of the whole database respectively. A set of language models and decision trees where trained for several amounts of candidates. Moreover, the best set of models in the development set where chosen to be applied to the test set. F-measure was chosen as the optimisation function.

	Set1	*Set2*	*Set3*	*Set4*
Motivation	small context	large context	*repeat* feature	no-candidates
F-measure	**86**	**86**	84	85
Language model order	2	2	2	4
Number of candidates	40	40	30	0

Table 2. Summary of experiments for each set of features proposed. Best systems on development set are shown. F-measure corresponds to test-set.

Table 2 shows F-measure values over the test for best systems on the development set and for each set of features proposed. There are not big differences between feature sets. The use of the *repeat* feature slightly decreases the F-measure. In contrast, the use of candidates slightly improves results, moreover, its computational cost is much lower.

Table 3 shows detailed results for the winner system based on Set1. The algorithm classifies each word as preceding (FP) or not ($\overline{FP}$) a filled pause. It can be observed how the algorithm presented here inserts filled pauses with 96% precision. However, it only adds 57% of filled pauses existing in the test set.

Confusion Matrix	$\overline{FP}$	FP	*Recall*	*Precision*	*F-measure*
$\overline{FP}$	**30,991**	12	99.9%	99.2%	99
FP	254	**347**	**57.7%**	**96.7%**	**82.3**

Table 3. Classification Results of best system. Precision and Recall are shown as well as F-measure for each class. Second order language model and 20 candidates.

Presented results lead us to claim that the algorithm presented can be used to learn from conversational corpus in order to generate them afterwards.

4.2 Perceptual Evaluation

Disfluent speech synthesis is a relatively recent research line. The evaluation of such systems is thus an unsolved issue. While speech synthesis evaluation is still a hot research topic; emotional, disfluent, . . . speech synthesis even add more difficulties to the evaluation process. However, qualitative as well as quantitative perceptual evaluations are necessary to allow us to extract conclusions that can lead our future research work.

In order to evaluate the quality of the inserted filled pauses as well as of its synthesis, some sentences extracted from the test set described in Section 4.1 were synthesised with and without disfluencies. The unit-selection synthesiser from Universitat Politècnica de Catalunya was used [15], and filled pauses were inserted in the audio using the methodology described in Section 3.

The test consisted on a set of 6 audio pairs. Each pair consisted on a sentence synthesised with and without filled pauses. Three of them where randomly chosen from the set of sentences the algorithm matched the reference (i.e. inserted a filled pause in a place where the reference contained one). The rest where chosen from the set the algorithm inserted a filled pauses in a place where the reference did not contain any (i.e. the algorithm did not match the reference). Each evaluator had to answer to 5 questions for each pair. Three of them related to naturalness and adequacy of the voice for a dialogue system:

- **Q1** Do you think that filled pauses make the voice *(more|equal|less)* natural?
- **Q2** Do you think that filled pauses make the voice *(more|equal|less)* suitable for a dialogue?
- **Q3** Do you think that filled pauses make the voice *(more|equal|less)* human-like?

and two questions related to the position of filled pauses and quality of their synthesis:

- **Q4** Do you think that filled pauses are (*correctly*|*incorrectly*) pronounced?
- **Q5** Do you think that filled pauses are (*correctly*|*incorrectly*) placed?

Answers to first three questions where given values [$more$ = 1, $equal$ = 0, $less$ = −1] respectively. The questionnaire was answered by 21 evaluators which had few or no relation with speech synthesis plus 4 evaluators that are speech experts. Results are shown in Table 4.

question	**Q1**		**Q2**		**Q3**		**Q4**		**Q5**	
	NoExp.	*Exp.*	*NoExp.*	*Exp.*	*NoExp.*	*Exp.*	*NoExp.*	*Exp.*	*NoExp.*	*Exp.*
match	0.4(0.6)	0.6(0.2)	0.08(0.7)	0.3(0.5)	0.4(0.5)	0.7(0.4)	71%	91%	71%	83%
no-match	0.5(0.6)	0.2(0.7)	0.6(0.4)	0.6(0.6)	0.3(0.5)	0.8(0.1)	78%	100%	82%	83%

Table 4. Perceptual results. Mean values are given and variance in brackets for Q1, Q2 and Q3. For Q4 and Q5 the percentage of evaluator that answer *"correct"* is given.

In Table 4 it can be observed how mean values show that the sentences presented in the test have been considered more natural (0.4 to 0.6) and more human-like (0.4 to 0.8). They have been considered slightly more suitable for a dialogue.

On the other hand, experts and non-experts agree that filled pauses where mainly correctly placed and correctly pronounced. As can be seen in Table 4, experts evaluated synthesis with higher values than non-experts, it might be due to the fact that they are more used to speech synthesis quality.

Opinion	**Evaluators**	**Reason**
`In favour`	22	*human-like speech is easier to understand*
`Does not matter`	2	*Not necessary but can be understood.*
`Against`	1	*Communication with machines has to be simple.*

Table 5. Qualitative summary of answers to the open question: *Do you think that it is interesting that in human-machine interactions speech synthesisers produce a more human-like voice?.*

In addition to these five questions an open question has been included in the test. It was: *Do you think that it is interesting that in human-machine interactions speech synthesisers produce a more human-like voice?* and answers to this question have been summarised in Table 5. These comments, thus, support our claim that talking speech synthesis is worth further research.

5 Conclusion

In the present paper we have described a system that is able to insert filled pauses in a text. It has been evaluated objectively against a reference corpus. Results have shown that filled pauses can be inserted with a precision of up to 96%. These results, lead us to the conclusion that filled pauses can be correctly inserted in a text by means of simple machine learning techniques. Furthermore, perceptual results support the conclusion coming up from objective results. Even sentences that do no correspond with the reference have been evaluate as correct. It has also been shown that the use of filled pauses in speech synthesis can increase the naturalness of the speech and that conversational speech is somehow desired by users. This encourages future work on disfluent speech synthesis including repetitions, restarts, etc.

6 Acknowledgements

This work has been partially founded by the Spanish government, under grant TEC2006-13694-C03 and by Consejería de Educación under project JCYL (VA053A05).

References

1. Mostefa, D., Garcia, M.N., Hamon, O., Moreau, N.: Deliverable 16: Evaluation report. Technical report, ELDA (2006)
2. Bennett, C.L., Black, A.W.: The blizzard challenge 2006. In: Proceedings of Blizzard Challenge 2006 Workshop. (2006) Pittsburgh, PA.
3. Shröder, M.: Emotional Speech Synthesis: A Review. In: Proceedings of Eurospeech. Volume 1. (2001) 561–564 Aalborg, Denmark.
4. Gobl, C., Bennet, E., Chasaide, A.N.: Expressive synthesis: How crucial is voice quality. In: Proceedings of IEEE Workshop on Speech Synthesis. (2002) 91–94 Santa Monica, California.
5. Werner, S., Hoffman, R.: Pronunciation variant selection for spontaneous speech synthesis - A summary of experimental results. In: Proc. of International Conference on Speech Prosody. (2006) Dresden, Germany.
6. Tree, J.E.F.: The effects on of false starts and repetitions on the processing of subsequent words in spontaneous speech. Journal of Memory and Language **34** (1995) 709–738
7. Tseng, S.C.: Grammar, Prosody and Speech Disfluencies in Spoken Dialogues. PhD thesis, Department of Linguistics and Literature, University of Bielefeld (1999)
8. Adell, J., Bonafonte, A., Escudero, D.: Disfluent speech analysis and synthesis: a preliminary approach. In: Proc. of 3th International Conference on Speech Prosody. (2006) Dresden, Germany.
9. Agüero, P.D., Bonafonte, A.: Phrase break prediction: a comparative study. In: Proc. of XIX Congreso de la Sociedad Española para el procesamiento del Lenguaje Natura. (2003) Alcala de Henares, Spain.
10. Stolcke, A.: SRILM - an extensible language modeling toolkit. In: Proc. Intl. Conf. Spoken Language Processing. (2002) Denver, Colorado.
11. Carreras, X., Chao, I., Padró, L., Padró, M.: Freeling: An open-source suite of language analyzers. In: Proc. of the 4th International Conference on Language Resources and Evaluation (LREC'04). (2004) Lisbon, Portugal.
12. Tree, J.E.F.: Listeners' uses of *um* and *uh* in speech comprehension. Memory & Cognition **29**(2) (2001) 320–326
13. Shriberg, E.E.: Preliminaries to a Theory of Speech Disfluencies. PhD thesis, Berkeley's University of California (1994)
14. Boersma, P., Weenink, D.: Praat: doing phonetics by computer (version 4.3.04) (2005) http://www.praat.org/.
15. Bonafonte, A., Agüero, P.D., Adell, J., Pérez, J., Moreno, A.: Ogmios: The UPC text-to-speech synthesis system for spoken translation. In: TC-STAR Workshop on Speech-to-Speech Translation, Barcelona, Spain (2006) 199–204