Handling Disfluencies in Spontaneous Language Models

Jacques Duchateau, Tom Laureys, Kris Demuynck, and Patrick Wambacq

ESAT–PSI Speech Group, K.U. Leuven

Abstract

In automatic speech recognition, a stochastic language model (LM) predicts the probability of the next word on the basis of previously recognized words. For the recognition of dictated speech this method works reasonably well since sentences are typically well-formed and reliable estimation of the probabilities is possible on the basis of large amounts of written text material. However, for spontaneous speech the situation is quite different: disfluencies distort the normal flow of sentences and written transcripts of spontaneous speech are too scarce to train good stochastic LMs. Both factors contribute to the poor performance of automatic speech recognizers on spontaneous input. In this paper we investigate how one specific approach to disfluencies in spontaneous language modeling influences recognition performance.

1 Introduction

The automatic recognition of spontaneous speech is currently one of the main topics in speech research. Practical applications include voice operated telephone services, automatic closed captioning for TV programmes, control of handheld devices, automatic transcription of meetings, etc. Yet, the recognition accuracy of freely spoken language is quite poor when compared to that of dictated speech: while the word error rate (WER) for large vocabulary speaker-independent dictation is about 5%, the WER for spontaneous speech recognition ranges from 15% for broadcast news (Beyerlein et al. 1999, Gauvain et al. 1999) to 40% for meeting and telephone conversation transcription (Yu et al. 2000).

Several factors contribute to an explanation of the low accuracy of spontaneous speech recognition. First, the acoustical environment for spontaneous speech recognition is often 'corrupted' by background noise, echo, music, bandwidth limitations, etc. As opposed to dictation, which is typically set in a rather quiet office environment, applications with spontaneous speech input require added environmental robustness (Gadde et al. 2002). Second, when comparing casual to dictated speech, the articulation quality of the former is very often lower while the speaking rate is higher. In addition, spontaneous speech reflects more emotions than read-aloud speech. All these elements put heavier demands on the recognizer's acoustic models, for which only limited matching training material is available. Finally, the same lack of stylistically matching training data holds for the spontaneous language models. Written transcripts of casual language use are rather scarce, while typical large vocabulary stochastic language models rely on vast amounts of training material (Adda et al. 1999). The occurrence of disfluencies in casual speech makes the problem even worse.

This paper focuses on the latter problem as it describes some experiments in spontaneous language modeling for automatic speech recognition. In the literat-

ure different approaches have already been pursued. Ma et al. (2000) try to deal with spontaneous language by incorporating knowledge of discourse theory: sentences typically start with given information whereas new information comes at the end. Correspondingly, two 'expert' language models were trained on the relevant sentence parts, yielding a slight 0.3% absolute improvement in WER for recognition of spontaneous telephone conversations (Switchboard). Disfluencies almost always occurred in the sentence's given information part. Zechner and Waibel (1998) explore N-best list rescoring on the basis of chunking information. The underlying motivation is that the coverage of the chunker bears information in order to discriminate between syntactically acceptable and syntactically anomalous recognition hypotheses. The technique reduced the WER by 0.3% absolute on Switchboard. Finally, Stolcke and Shriberg (1996) report on dealing with disfluencies in language modeling by editing the prediction context. More specifically, the prediction context for a newly hypothesized word is 'cleaned up' by removing the disfluencies in it. On the Switchboard task, no change in WER was found (so parallel to the other approaches, differences in WER are not significant). The research described in this paper extends the work by Stolcke and Shriberg by implementing a more flexible manipulation of the prediction context: sentence restarts are allowed at any point in the sentence (not only after the first and second word), and repetitions and hesitations are only removed from the context when they do not contain informational value.

The paper is organized as follows. First, we give a brief overview of the standard architecture of a large vocabulary continuous speech recognizer, particularly emphasizing the role of the language model. In section 3 we explain in more detail the problems of spontaneous language modeling and present our research on the topic. Section 4 describes the experimental set-up and gives results on a recognition task. Finally, we conclude and discuss future research on the topic.

2 Automatic Speech Recognition Architecture

A typical architecture for large vocabulary automatic speech recognition systems is shown in figure 1. We will briefly describe the search engine and the acoustic models. The language model will be presented in more detail.

The core module in recognition systems for large vocabulary is the *search engine*. The task of this module consists of efficiently searching for the most likely sequence of words, given an input speech signal. In practice, the engine goes through the signal frame by frame (a *frame* is a 10 millisecond time slice of the signal), hypothesizes known words from a phonemic dictionary, and stochastically evaluates how probable they are, using the acoustic model to determine the match with the speech signal and the language model to assess how likely the string of words is in the language.

As a full evaluation of any possible sentence with words that can start and end at any point in time is not feasible, the search engine will select during the search the most promising words and partial sentences, and will only explore those (a technique called *beam search*). Fortunately, in practice such a selection works

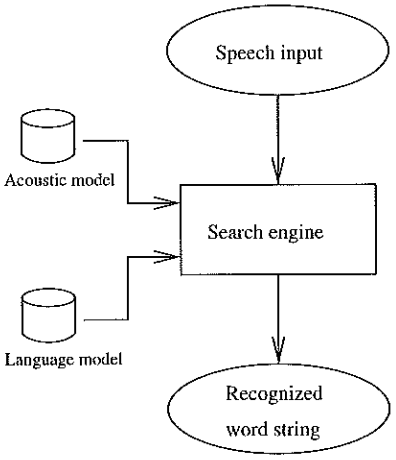

Figure 1: Architecture of a standard speech recognition engine

pretty well, and feasible, real time recognition is mostly possible without pruning away the sentence with the highest probability.

The first information source, the *acoustic model*, produces for each phoneme the probability that that specific phoneme was pronounced in a given speech frame. It first calculates some features that reflect the spectral content of the frame. Suppose for instance that—for vowels—the first and second formant are calculated. In practice about 30 features are used which describe the energy and energy change (time derivatives) in different frequency bands.

Next, based on the calculated features, the probability of a phoneme is found by evaluating the acoustic model for that phoneme. In the simple case with two formants as features, this acoustic model can be the average and variance of both formants (this is a two-dimensional *gaussian* distribution), which are estimated on a large acoustic training database with many samples of the phoneme. In typical acoustic models, a mixture of 30-dimensional gaussians describes a phoneme.

The role of the *language model* (LM) is defined as discriminating between likely and unlikely word sequences in a language.[1] While in small-scale constrained applications this is a feasible task for formal grammars, for large vocabulary unconstrained speech recognition we have to apply stochastic language models in order to ensure coverage. Stochastic LMs define a probability distribution $P(W)$ over word strings W, reflecting the probability of W occurring in the specific language. So a stochastic LM ideally assigns a higher probability to an ac-

[1] Note that this implies more than discriminating between linguistically acceptable and unacceptable word strings. The language model should also be able to pick the most likely string from two grammatically well-formed strings. For this reason, training and testing material should stylistically be as close as possible.

ceptable sentence than to a nonsense expression. By applying the chain rule we can decompose $P(W)$ as follows:

$$\begin{aligned} P(W) &= P(w_1, w_2, \ldots, w_n) \\ &= P(w_1)P(w_2 \mid w_1) \ldots P(w_n \mid w_1, w_2, \ldots, w_{n-1}) \\ &= \prod_{i=1}^{n} P(w_i \mid w_1, w_2, \ldots, w_{i-1}), \end{aligned}$$

where $P(w_i \mid w_1, w_2, \ldots, w_{i-1})$ is the probability that word w_i will follow given the preceding sequence of words $w_1, w_2, \ldots, w_{i-1}$. This preceding sequence of words is called the *prediction context* for w_i.

As the LM's probability distribution is estimated on large text corpora, it is in practice infeasible to get reliable estimates for values of i higher than say 4 or 5; longer prediction contexts will virtually never appear in the training corpus and thus mostly be assigned probability 0. As a result, stochastic LMs adopt the Markov assumption, stating that the probability of a word depends only on a fixed number of previous words. This assumption turns the stochastic LM into an N-gram, or more specifically into a unigram $P(w_i)$ (prediction context of 0 words), a bigram $P(w_i \mid w_{i-1})$ (prediction context of 1 word), etc. Taking into account the available amounts of textual training data, most large vocabulary speech recognition work with trigrams up to five-grams, combined with a back-off to lower order N-grams if necessary.

3 Language Models for Spontaneous Speech

First we sum up the challenges involved in spontaneous language modeling in general. Then we describe the model we developed for dealing with disfluencies in the prediction context.

3.1 Challenges

The challenges of statistically modeling spontaneous language are numerous. We discuss the grammatical and stylistic differences between written and spoken language, and the different types of disfluencies covered in the described research.

In section 1 we already touched upon the issue of lack of stylistically matching training data for stochastic spontaneous language modeling. In practice, stochastic LMs for large vocabulary recognition tasks are trained on minimally 10M words of (stylistically matching) text.[2] Whereas large collections of written electronic texts allow for the training of accurate dictation LMs, such text corpora are not available for casual spoken language. As a result, stochastic LMs for spontaneous speech recognition are trained on stylistically different written material and/or on too small an amount of available literal transcripts of conversations, meetings, etc. The grammatical and stylistic differences between written and spoken language have

[2] Nowadays training on hundreds of millions of words is not unusual.

been described in detail by Biber (1988). Biber presents a quantitative analysis of 67 linguistic features in 23 spoken and written registers. By examining consistent co-occurrence patterns between these features he was able to identify linguistic dimensions (narrative/non-narrative concerns, explicit/situation-dependent reference, etc.) and to interpret them in functional terms. Along most of the six dimensions Biber discerns, spontaneous conversations and written material (press releases, prose, etc.) are far apart. A simplified example showing the difference in linguistic feature frequency between genres can be found in figure 2, adapted from Huang et al. (2001).

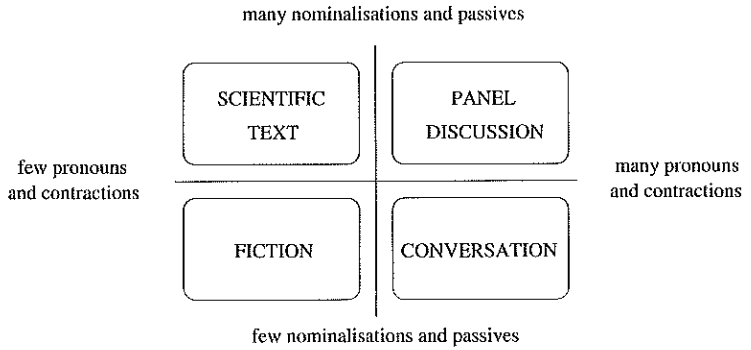

Figure 2: Register continuum with typical features

One solution that has recently been explored to collect more spontaneous language data, incorporates data from the internet and more in particular from newsgroups (Vaufreydaz et al. 1999). Although newsgroup language use is definitely more casual and interactive than, for example, newspaper text, it is still far from optimal for training stochastic spontaneous language models.

Among the features distinguishing spontaneous from read speech, we want to highlight the occurrence of speech disfluencies. The disfluency types we focus on in this work are listed below:

repetitions: *That is what **what** I think.*

hesitations: *That is what **um** I think.*

sentence restarts: *That is what... Yeah I think so.*

Note that we did not include all types of disfluencies (e.g. speech errors, substitutions, ...). This decision is motivated by the fact that about 85% of the disfluencies in our train/test corpus Switchboard (described in more detail below) are of the three types listed above (Shriberg 1996). The selected types therefore adequately reflect the modeling problem we want to cope with here.

One of the hypotheses explaining the difficulty of spontaneous language modeling by means of N-grams points explicitly to disfluencies: as N-grams base their word prediction on a local context of N-1 previous words, intervening disfluencies render this context less uniform. Or put differently, the prediction of a next word would be more accurate if based on a context from which disfluencies are removed and which is extended to the left with regular words to make up for the removed disfluencies. So when using a trigram LM in the case of the hesitation mentioned above, we hypothesize that 'I' would be better predicted by the context 'is what' than by 'what um'. The disfluencies themselves are predicted the same way as regular words.

Yet, as shown by Siu and Ostendorf (1996) and Shriberg and Stolcke (1996), in some cases disfluencies *are* good predictors for following words. Hesitations, for example, sometimes tend to precede less frequently used words (depending, among other factors, on the sentence position of the hesitation). In addition, repetitions are not always grammatically incorrect (e.g. *I hope that that work is done.*) So simply removing disfluencies from the prediction context seems too crude. In our model we tried to incorporate this observation by allowing the system to pick the most probable option when both a context with disfluency and a cleaned-up context were available.

3.2 The Proposed Model

We explain in detail how the proposed model works by taking the case for repetitions as an example. The model for repetitions is sketched in figure 3. In this figure, the LM is presented as a Markov Model: in the circles the prediction context for the next word is given, on the arcs the next word is shown. As can be seen on the figure, we assume that a trigram language model is used. The upper path illustrates the normal LM procedure. Suppose that word 'B' is repeated, then the prediction of the next word 'C' is based on the context 'B B'. The removal of the repetition is demonstrated by the lower path. The prediction of 'C' is made on the basis of the modified context 'A B'; the repeated word 'B' is removed. As mentioned above, we also investigated selecting the most probable prediction context, this is the context with the most probable transition to the newly hypothesized word according to the stochastic LM. So in that case the prediction of 'C' is based on the most probable of both contexts mentioned.

The analogous models for hesitations (symbol *uh*) and sentence restarts (the symbol $< S >$ is used to represent the context at the beginning of a sentence) are depicted in figures 4 and 5 respectively. The top arc gives the normal LM procedure, the bottom arc handles the disfluency particularly. The figures show that in these cases, it takes one word more for both options to join again. It should be noted that in the model for sentence restarts, a sentence restart is allowed following any word, even though this generates many hypotheses.

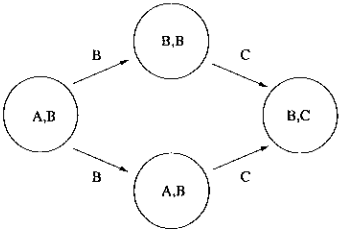

Figure 3: The model for repetitions

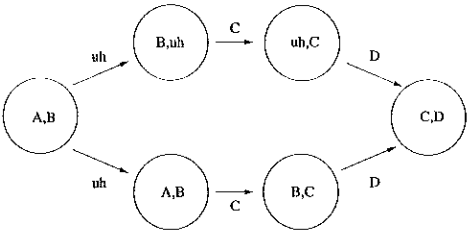

Figure 4: The model for hesitations

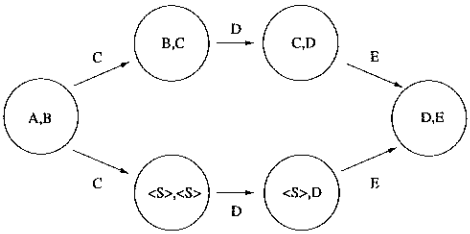

Figure 5: The model for sentence restarts

4 Experiments and Results

4.1 Experimental Setup

The adapted LM was evaluated by means of recognition experiments with the ESAT speech recognizer. The Switchboard corpus, a collection of informal telephone conversations in American English (Godfrey et al. 1992), was used for training and testing. For the evaluation, gender independent acoustic models were trained on 65 hours of Switchboard data. A global phonetic decision tree defines the 8351 tied states in the cross-word context dependent and position dependent models.[3] A Good-Turing smoothed trigram language model was built on the basis of 2M words taken from Switchboard literal conversation transcripts.[4] The recognition lexicon, which consisted of 23K words, was closed (no out-of-vocabulary words).

For the test set 5 phone calls (10 different speakers) were selected which were not part of the training material. They formed 22.9 minutes of speech and contained 4977 words in 531 sentences. About 6% of these words were disfluencies. Note that due to the rather small percentage of disfluencies we should not expect enormous changes in WER when applying specific disfluency modeling techniques.

Before turning to the real recognition experiments, we set up a small-scale experiment to investigate whether the probabilities in the trigram language model, estimated on a rather small 2M word text database, were reliable enough to distinguish between the different prediction contexts compared in the experiments. We did this by analyzing sentence restarts after the hesitation *uh* in a Switchboard test set. The test set contained 72 occurrences of *uh* in the middle of the sentence. From a manual examination we learned that in 18% of these cases the sentence restarted following the hesitation, and in the remaining 82% of the cases the sentence just went on.

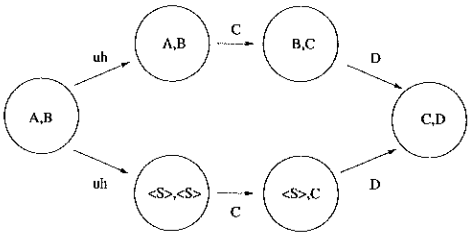

Figure 6: The model for the language model evaluation

[3] For more details on acoustic modeling, see Huang et al. (2001).

[4] Note that we stuck to the text material which was stylistically (matching topic, word usage, ...) closest to the test material, even though this consists of only 2M words. Including material from a (stylistically) different source would be a separate research topic by itself, as explained in section 3.1.

Table 1: WERs for the different disfluency types with varying context manipulation

	unchanged	**changed**	**choice**
repetition	39.2%	39.0%	39.2%
hesitation	39.2%	39.4%	39.2%
restart	39.2%	n/a	39.5%

Next, we made the language model choose between both contexts with the model depicted in figure 6. We found that both for the sentence restarts and for the continued sentences, the LM was able to select the correct transition in the model in 84% of the cases. This clearly indicates that most information on the optimal LM prediction context can be found in the trigram language model.

4.2 Results and Discussion

The resulting WERs for the recognition experiments are summarized in table 1. For each of the three disfluency types, three types of context manipulation were investigated: leaving the context unchanged (the baseline experiment), changing the context according to the model, and choosing the most probable of the two former options. Note that a forced context change according to the model is not applicable to the case with sentence restarts, as this would mean that all words are predicted as if they were the first word in the sentence.

As can be seen from the table, only a forced removal of repetitions from the context yields an improvement in WER (0.2% absolute). This result is consistent with other research (Stolcke and Shriberg 1996). On the other hand, cleaning up hesitations leads to slightly worse results. As already discussed in section 3.1, this higher WER can be attributed to the fact that, depending on their sentence position, hesitations can offer reliable cues for lexical choice.

As for the models that leave the recognizer the choice between two prediction contexts, the presented results do not show a consistent behaviour. It seems that there is a trade-off between an improvement in WER because the information in the language model is used more optimally (as shown in section 4.1), and a deterioration because introducing options may tend to *overgenerate* possible hypotheses.

We will illustrate this idea of overgeneration with an example. Suppose that the speech signal for a well-formed and well-pronounced sentence matches acoustically to a string of words, except for a few *superfluous* phonemes (they are in fact not superfluous: the string of words is wrong). Then the acoustic match can be improved (often drastically) by adding a short word to the string of words. However this extra word will typically not fit the language model, and hence the sentence with the extra word will be rejected. But when the recognizer is allowed to restart the sentence at any point, for instance following the extra word, chances are much higher that at least one of the generated possibilities fits the language model.

In that case, the recognizer doesn't have any (statistical) ground left to reject the sentence.

The danger of overgeneration is especially high for the model with sentence restarts as it allows to reset the context to a new sentence following any word. This explains the slightly higher WER when offering options to this type of disfluency. It may be a good idea to restrict offering choices here, for instance to cases where the previous word indicates a probable sentence restart (like *uh*), or to cases in which there are acoustic-prosodic cues pointing to the presence of a disfluency.

5 Conclusions and Future Research

In this paper we investigated whether spontaneous language modeling could benefit from a specific approach to disfluencies. We tried to improve on a plain trigram LM by manipulating prediction contexts containing repetitions, hesitations or restarts. First, disfluencies were automatically removed from the LM prediction context. This turned out to be beneficial for repetitions, while having a bad effect on contexts containing hesitations. In a second experiment we offered both the manipulated and non-manipulated prediction context and let the search engine pick the one with the most probable transition to the newly hypothesized word. This way we tried to anticipate the fact that in some cases disfluencies are strongly correlated with lexical choice. The results for the latter experiment were rather disappointing. We think this is largely due to leaving too many options open to the system. Building in some context-dependent restrictions might be beneficial here.

A first step in our future research will be the combination of our restart model with acoustic-prosodic information. The current option of starting a new sentence at each point overgenerates. This overgeneration could be reduced by including acoustic information on sentence and/or phrase breaks. Further, we will set up some experiments on the inclusion of additional LM training material. Experimenting with 'casual', more or less interactive text data from the internet might be an option here. A stochastic LM trained on additional carefully selected matching data could not only improve recognition as such, but might also lead to a more accurate automatic context selection. Finally, we will build language models for speech recognition of spontaneous language use in Dutch. The CGN corpus (Corpus Gesproken Nederlands/Spoken Dutch Corpus)[5] offers some new possibilities in this respect.

Acknowledgements

This research was funded by IWT in the STWW programme, project ATraNoS (`http://atranos.esat.kuleuven.ac.be`).

[5] `http://lands.let.kun.nl/cgn`

References

Adda, G., Jardino, M. and Gauvain, J. (1999), Language modeling for broadcast news transcription, *Proceedings of the European Conference on Speech Communication and Technology*, Vol. IV, Budapest, pp. 1759–1762.

Beyerlein, P., Aubert, X., Haeb-Umbach, R., Harris, M., Klakow, D., Wendemuth, A., Molau, S., Pitz, M. and Sixtus, A. (1999), The Philips/RWTH system for transcription of broadcast news, *Proceedings of the European Conference on Speech Communication and Technology*, Vol. II, Budapest, pp. 647–650.

Biber, D. (1988), *Variation across Speech and Writing*, Cambridge University Press, Cambridge.

Gadde, V., Stolcke, A., Vergyri, D., Zheng, J., Sonmez, K. and Venkataraman, A. (2002), Building an ASR system for noisy environments: SRI's 2001 SPINE evaluation system, *Proceedings of the International Conference on Spoken Language Processing*, Vol. III, Denver, pp. 1577–1580.

Gauvain, J., Lamel, L., Adda, G. and Jardino, M. (1999), Recent advances in transcribing television and radio broadcasts, *Proceedings of the European Conference on Speech Communication and Technology*, Vol. II, Budapest, pp. 655–658.

Godfrey, J., Holliman, E. and McDaniel, J. (1992), Switchboard: Telephone speech corpus for research and development, *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, Vol. I, San Francisco, pp. 517–520.

Huang, X., Acero, A. and Hon, H. (2001), *Spoken Language Processing*, Prentice Hall, Englewood Cliffs.

Ma, K., Zavaliagkos, G. and Meteer, M. (2000), Bi-modal sentence structure for language modeling, *Speech Communication* **31** (1), 51–67.

Shriberg, E. (1996), Disfluencies in Switchboard, *Proceedings of the International Conference on Spoken Language Processing*, Vol. Addendum, Philadelphia, pp. 11–14.

Shriberg, E. and Stolcke, A. (1996), Word predictability after hesitations: a corpus-based study, *Proceedings of the International Conference on Spoken Language Processing*, Vol. III, Philadelphia, pp. 1868–1871.

Siu, M. and Ostendorf, M. (1996), Modeling disfluencies in conversational speech, *Proceedings of the International Conference on Spoken Language Processing*, Vol. I, Atlanta, pp. 386–389.

Stolcke, A. and Shriberg, E. (1996), Statistical language modeling for speech disfluencies, *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, Vol. I, Atlanta, pp. 405–408.

Vaufreydaz, D., Akbar, M. and Rouillard, J. (1999), Internet documents: A rich source for spoken language modeling, *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, Keystone, pp. 277–281.

Yu, H., Tomokiyo, T., Wang, Z. and Waibel, A. (2000), New developments in automatic meeting transcription, *Proceedings of the International Confer-*

ence on Spoken Language Processing, Vol. IV, Beijing, pp. 310–313.

Zechner, K. and Waibel, A. (1998), Using chunk based partial parsing of spontaneous speech in unrestricted domains for reducing word error rate in speech recognition, *Proceedings of the 17th Conference on Computational Linguistics (COLING/ACL'98)*, Montreal, pp. 1453–1459.