A PCFG for Prosodic Structure. Experiments on German

*Michaela Atterer** & *Sabine Schulte im Walde*+

*Institute for Natural Language Processing, University of Stuttgart, Germany
+Brockhaus Duden Neue Medien GmbH, Dudenstraße 6, 68167 Mannheim, Germany
*atterer@ims.uni-stuttgart.de; +sabine.schulte-im-walde@bifab.de

Abstract

In this paper we investigate the usefulness of a probabilistic context-free grammar (PCFG) for assigning prosodic structure to unlabelled text. We develop and train a grammar for experiments on German, utilising prosodic non-terminal categories such as *phi-phrases*. The PCFG is evaluated on test data and by human blind labelling. The statistical prosodic rules can be used in a text-to-speech synthesis system for determining the location of prosodic breaks.

1. Introduction

Text-to-speech synthesis systems sound monotonous and unnatural if the flow of words is not interrupted by prosodic breaks (manifested as boundary tones or pauses). They divide an utterance up into smaller units, thus imposing prosodic structure on it. Among others, [1] and [2] investigate the nature of this prosodic structure, and how it relates to the syntactic structure of sentences. They define rules on how to map syntactic structure onto prosodic structure.

[3] compute prosodic structure for speech synthesis by using the syntactic output of a parser. The raw text is assigned a syntactic structure according to a grammar (used by the parser), with the grammar being developed by humans. But instead of developing a grammar for syntactic structure and then mapping syntactic trees onto prosodic trees, it seems more natural from a computational point of view to directly develop a grammar for *prosodic* structure, and parse the sentence according to this prosodic grammar.

In this paper we investigate how to develop and use a probabilistic context-free grammar (PCFG) for the task of assigning prosodic structure to unlabelled text. To our knowledge this has not been tried yet, and we believe it is worth finding out how feasible this approach is. We start out with explaining PCFGs and how they are used for parsing. Then we review some linguistic theories on prosodic structure that inspired our grammar. Next we describe our grammar rules, and how we trained and evaluated the grammar.

2. PCFGs

Definition 2.1 *A probabilistic context-free grammar $PCFG$ is a quintuple $\langle N, T, R, p, S\rangle$ with*

- N *finite set of non-terminal symbols*
- T *finite set of terminal symbols,* $T \cap N = \emptyset$
- R *finite set of rules* $C \rightarrow \gamma$, $C \in N$ *and* $\gamma \in (N \cup T)^*$
- p *corresponding finite set of probabilities on rules,* $(\forall r \in R) : 0 \leq p(r) \leq 1$ *and* $(\forall C \in N) : \sum_\gamma p(C \rightarrow \gamma) = 1$
- S *distinguished start symbol,* $S \in N$

Probabilistic context-free grammars combine context-free grammar rules with statistical ratings. Therefore, they can model complex syntactic structures of sentences and utilise statistical grammar parameters for structural plausibility judgements. The ranking of multiple syntactic tree analyses is based on parse tree probabilities for sentences or parts of sentences, cf. equation (1). The probability of a syntactic tree analysis $p(t)$ for a sentence is defined as the product of probabilities for the rules r applied in the tree. The frequency of a rule r in the respective tree is given by $f_t(r)$.

$$p(t) = \prod_{r \ in \ R} p(r)^{f_t(r)} \quad (1)$$

The PCFG parameters are learned by a parser. For our purposes we use the parser `LoPar` [4] which executes the parameter training of PCFGs by the *Inside-Outside Algorithm* [5], an instance of the *Expectation-Maximisation (EM) Algorithm* [6]. The EM-algorithm is an unsupervised iterative technique for maximum likelihood approximation of training data. It is guaranteed to find a local optimum in the search space. For the *Inside-Outside Algorithm*, the EM-parameters refer to grammar-specific training data, i.e. how to determine the probabilities of sentences with respect to a grammar.

3. The prosodic hierarchy

The PCFG constructed here is largely influenced by the idea of a prosodic hierarchy illustrated by the example in Figure 1 which was adapted from [2]. They follow an account by [1] which states that even though the prosodic tree structure of a sentence is not identical to the tree structure imposed by theories of syntax, a non-trivial mapping between the two structures does exist. A key element in this mapping is the phonological (ϕ) phrase which bundles prosodic words ω. As opposed to intonational (I) phrases which are usually associated with a break, ϕ-phrases can more easily be defined in syntactic terms. Nespor and Vogel define a ϕ-phrase as "a lexical head X and all the material on its non-recursive side up to the next head outside of the maximal projection of X". For example, the lexical head of a noun phrase like "her attic" is the noun "attic", the lexical head of a verb phrase is the verb, etc. The non-recursive side in English is always the left-hand side. So a lexical head forms a ϕ-phrase with all the words to its left up to the next head unless this head is inside the maximal projection of the former head. So ϕ-phrases are basically lower-level syntactic chunks where function words are attached. This idea and the idea of bundling ϕ-phrases into intonational phrases is used in the PCFG described in this paper.

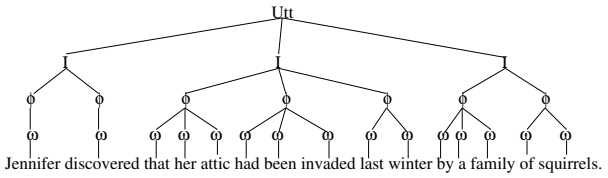

Figure 1: Example of the prosodic hierarchy.

4. Topological positions of German verbs

In German, modal and auxiliary verbs are often separated from the main verb by other material, cf. example 1. This peculiarity of the German topological structure is the reason why we need to distinguish two kinds of verbal ϕ-phrases in our grammar in section 5: (i) rules for verbs which occur in the left slot 'linke Satzklammer' of a sentence (such as *ließ* in example 1), and (ii) rules for verbs which occur in the right slot 'rechte Satzklammer' of a sentence (such as *schneiden* in example 1). We assume that the left part of the verb is never followed by a break option, and thus cannot form a ϕ-phrase on its own, but builds a ϕ-phrase with the following chunk (here: a noun chunk). The right part of the verb can be a ϕ-phrase on its own, even though this is not very likely.

(1) Er *ließ* seine Haare gestern von dem Friseur, der seit kurzem den Laden in der X-strasse gemietet hat, *schneiden*.

lit: He *had* his hair yesterday by the hair dresser, who since recently the salon in the X-street rented has, *cut*.

5. A German PCFG for prosodic phrases

We created a relatively simple PCFG that took little time to develop. The grammar has four kinds of rules, *Hierarchy Rules*, *Attaching Rules*, *Chunking Rules*, and *Simple Rules*.

Hierarchy Rules are the top-level rules and bundle syntactic chunks found by lower-level rules of the grammar. They group intonational phrases (IPs) into utterances (UTTs), and ϕ-phrases into IPs. As opposed to previous approaches concerning the prosodic hierarchy we distinguish several categories to identify the syntactic chunk on which ϕ-phrases are based. We have noun-ϕ-phrases (PHI.N), adjective-ϕ-phrases (PHI.A), etc. An utterance `UTT` consists of up to 6 IPs.

```
UTT → IP
UTT → IP IP
UTT → ...
UTT → IP IP IP IP IP IP
```

Examples of the rules for bundling ϕ-phrases are as follows. An IP can consist of a noun phrase, or of an adjective phrase. Or it can consist of a proper-name ϕ-phrase `PHI.E`, a verb-right ϕ-phrase `PHI.VR` and a punctuation mark, or of a verb-left ϕ-phrase, a noun ϕ-phrase and a verb-right ϕ-phrase followed by a punctuation mark:

```
IP → PHI.N
IP → PHI.A

IP → PHI.E PHI.VR PM
IP → PHI.VL PHI.N PHI.VR PM
```

Chunking Rules construct syntactic chunks, such as noun chunks. The rules are flat; no recursion is used, with one exception: ϕs are allowed to occur on the right-hand side of a rule, so they can be extended by prepositions, function words etc. We define 6 different kinds of ϕ-phrases: noun-ϕ-phrases (PHI.N), prepositional ϕ-phrases (PHI.P), adjective and adverb ϕ-phrases (PHI.A), proper name ϕ-phrases (PHI.E), verb-left ϕ-phrases (PHI.VL), and verb-right ϕ-phrases (PHI.VR). Some examples are given below[1].

```
PHI.N → ADJ ADJ NN
PHI.P → PREP PHI.N
PHI.VR → PHI.N VERB.R
```

A noun ϕ-phrase can consist of two adjectives followed by a noun, a prepositional ϕ-phrase can consist of a preposition and a noun ϕ-phrase. A verb-right ϕ-phrase can consist of a noun ϕ-phrase followed by a verb form in the right verb slot of a sentence. The grammar contains about 80 of these chunking rules.

Attaching Rules extend ϕ-phrases by attaching function words to them. Most function words attach to the left of ϕ-phrases in German, but there are also function words such as post-positions which attach to the right. The following rules exemplify how function words are attached to ϕ-phrases. Function words which qualify as left-attaching function words `FKT.L` attach to the left. Right-attaching function words `FKT.R` attach to the right.

```
PHI.N → FKT.L PHI.N
PHI.N → PHI.N FKT.R
```

Simple Rules define what is a left-attaching function word FKT.L or a right-attaching function word FKT.R:

```
FKT.L → KON                    (conjunction)
FKT.L → PTKNEG                 (negation particle)

FKT.R → APPO                   (postposition)
FKT.R → APZR                   (circumposition, right part)
```

They also define the verbal complexes at the end of a sentence which can take part in a verb-right -ϕ-phrase PHI.VR:

```
VERB.R → VMFIN                 (finite modal verb)
VERB.R → VVPP VAPP VAFIN       (2 participles + auxiliary)
```

Or they are used for simplification, e.g. in the case where we want to treat various kinds of POS-tags the same way:

```
DET → PDAT                     (attributive pronoun)
DET → ART                      (article)

BREAK → ⟨H%⟩                   (high tone)
BREAK → ⟨L%⟩                   (low tone)
BREAK → ⟨%⟩                    (undistinguishable tone)

PM → $(                        (parenthesis)
PM → $.                        (period)
```

6. Training and testing the PCFG

For the training of the PCFG, we use a training corpus which is marked up with breaks. In order to utilise the information on the breaks for the training effect, we extend each ϕ-rule in the grammar with a `BREAK` category, for example:

```
IP → PHI.E PHI.VR PHI.VL PHI.N PHI.A PM BREAK
IP → PHI.E PHI.VR PHI.VL PHI.N PHI.A BREAK
```

The training of the grammar was then performed in 4 training iterations on 6,000 words (380 sentences) of the IMS Radio News Corpus [8], a corpus of German news broadcast read by various speakers. Some of the sentences are repeated, by different speakers with different prosodic renditions. The corpus is manually labelled with prosodic breaks and accents. During the training step 90 sentences of the training corpus could not be parsed. But these failures are not a major problem for two

[1] The part-of-speech tags in our grammar refer to the STTS tag set [7].

reasons: (i) Some of the unparsed sentences suggest breaks we do not want, e.g. breaks between a noun and its modifying adjective:

(2.1) Bei der traditionellen | Revolutionsfeier [...]
(2.2) Lit: At the traditional | revolution celebration ...

(ii) Others which could not be parsed in the training step can well be parsed *in a correct way* (!) in the testing step, because there is usually more than one correct way to assign prosodic breaks to a sentence. After the grammar training, the artificial `BREAK` categories were deleted, while each of the rules is, of course, still marked with the probability learned from its `BREAK`-version. The trained grammar was run on the training corpus, but with the breaks deleted from the corpus annotation. Now there were only 6 sentences which could not be parsed.

The grammar was tested on 91 unseen sentences (1,407 words) of a separate test corpus from the IMS Radio News Corpus. The rule probabilities learnt in the training phase are used to build prosodic tree structures as shown in Figure 2. The most probable parse tree for each sentence was selected by the Viterbi algorithm. 8 sentences could not be parsed, with 2 of them occuring twice, so in fact there were only 6 failures. We measured *recall* (the percentage of breaks in the test corpus that were found by the model) and *precision* (the percentage of the breaks assigned by the model which were correct according to the test corpus). The F-score is calculated out of these two measures as $F = (2 \times \text{recall} \times \text{precision})/(\text{recall} + \text{precision})$. For the evaluation, those sentences that failed are assigned breaks only at punctuation marks. The evaluation results can be seen in Table 1. The baseline refers to prosodic structures where breaks are only inserted at punctuation marks in all sentences. The PCFG performs considerably better than the baseline.

Table 1: Results on IMS Radio News Corpus. (For the PCFG, unparsed sentences were evaluated with breaks at punctuation marks only.)

	Recall	Precision	F-score
PCFG	69.3	79.6	74.1
Baseline	49.1	95.2	64.8

A greater challenge and more interesting experiment than comparing the PCFG to the baseline is comparing it to one of the best methods for phrase-break assignment used so far, namely a Hidden-Markov Model (HMM). We employed an HMM-based approach similar to [9] (work in progress) where we used a window of POS-bigrams and a context length of 6 (which was the best context length for Taylor and Black). The HMM experiment achieves a precision of 88.7%, a recall of 80.3%, and an F-score of 84.3%.

7. Human evaluation

One peculiarity of prosodic structure is that there is usually more than one way to break up an utterance. The following examples show the prosodic structure that three different subjects assigned to the same sentence [10].[2]

(3.1) And Nelson Mandela has of course | been willing to pay that price.
(3.2) And Nelson Mandela has | of course | been willing to pay that price.
(3.3) And Nelson Mandela has of course been willing | to pay that price.

This peculiarity might have unwanted effects on an automatic evaluation as performed in section 6, where the prosodic structure assigned by a model is compared to only one gold standard. When we examine the prosodic phrasing that was given by the PCFG we find that most of the sentences are actually assigned an acceptable prosodic structure according to our intuition, and that the PCFG seems thus to be better than the automatic evaluation shows. To test whether our intuition is right we had two human evaluators judge the sentences of the test corpus. Both can be considered experts in prosodic phrasing, prosody and speech synthesis in general. Of the 91 sentences which were in the test corpus we discarded those where the PCFG and the HMM model agreed. We also discarded any sentences that occurred twice. This left us with 66 sentences. The subjects were presented with pairs of sentences where one sentence was marked up with the prosodic breaks the HMM assigned, and the other was marked up with the structure assigned by the PCFG. For about half of the pairs the sentence output by the HMM was presented first; for the other half the sentence with the structure assigned by the PCFG was presented first. One of the subjects worked through the sentence pairs in reverse order compared to the other. The subjects were asked to mark for each pair which of the phrasing alternatives they preferred. They were also allowed to mark "both" in cases where they did not prefer one alternative over the other, but they were asked to use the "both" lable sparingly.

One of the subjects marked 5 sentences with "both", 31 sentences as the HMM being better, and 30 sentences as the PCFG being better. The other subject marked 12 sentences as both being equal, 29 as the HMM being better, and 25 as the PCFG being preferred. The two subjects agreed on 48 of the 66 sentences (i.e. they were of the same opinion in terms of which model was better). 2 of those 48 sentences were marked with "both", 23 were marked as the HMM being preferred, and 23 were marked as the PCFG being preferred.

8. Discussion

In this paper we investigated how feasible it would be to use a PCFG for the task of assigning prosodic structure to text. We created a relatively simple PCFG that did not take much time to develop. The automatic evaluation results are clearly above the baseline. They are, however, below the results a Hidden-Markov Model can achieve. A human evaluation that we carried out, however, suggests that the PCFG produces results almost as acceptable as an HMM. We take these results as encouraging, and think that, if some of the problems and difficulties we encountered during our development of the PCFG can be solved, a PCFG can render excellent phrasing results. The main problems that need to be solved are the following.

1. Balancing of prosodic constituents: The PCFG does not yet contain any mechanism which assures that the prosodic phrases are balanced, i.e. of approximately equal length. One way to go about this would be to combine the Viterbi search with a balancing mechanism.
2. Higher-level syntactic structure: One of the strengths of a PCFG is that it can learn relationships between constituents. In our case, the PCFG can learn which is the most likely phrasing for a row of 3 PP-ϕ-phrases when the sentence ends with a VR-ϕ-phrase. However, it cannot learn that a certain kind of I-phrase is more likely to

[2] The sentence is part of the Spoken English Corpus [11].

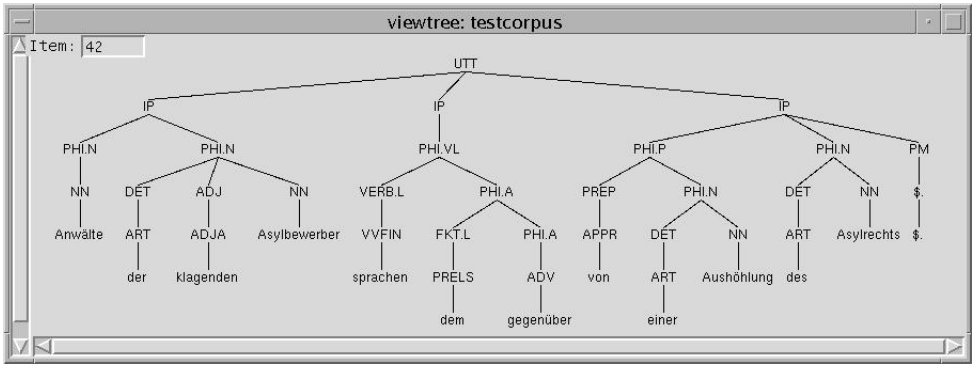

Figure 2: Example of prosodic structure assigned by the PCFG.

form a phrase with its preceding kind of I-phrase than with the following, because we do not have different types of I-phrases. It would be necessary to assign I-phrase categories to improve the grammar. Besides the problem of finding categories that make sense there is also the problem of combinatorical explosion, if rules like UTT $\rightarrow$ IP IP IP IP IP have to be rewritten. The number of categories would thus have to be small.

PCFGs have some more advantages over purely statistical techniques that make them worth being investigated. As they incorporate some human knowledge (in the grammar rules) they can overcome problems related to the training corpus. Prosodically labelled corpora often contain prosodic breaks at locations that are not optimal for the following reasons:

1. The speaker had an idiosyncratic way of structuring sentences. Human speakers are able to compensate for this by using other prosodic means, e.g. special, artful intonation patterns. A TTS system is usually not able to do this.
2. The annotation of the test corpus might be too strongly influenced by the labelling system. In `ToBI` [12], for instance, a prosodic phrase is required to contain an accent. So a labeller might not mark a break in absence of an accent, even in the presence of a pause.

When developing a PCFG we have influence on where we want to allow breaks, and can thus prevent the learning algorithm from learning idiosyncracies of the corpus.

Another advantage of using a PCFG is that some rare events that are not covered by the training corpus can still be modelled. For example, our training corpus might not contain all the tags that our tagger can assign. By looking at the tag-set however, we can find rare events such as post-positions and treat them adequately in the grammar. The PCFG will then be able to treat those tags when it is tested on a corpus which contains them.

9. Conclusion

In this work we have used some ideas from prosodic phonology to develop a PCFG for prosodic structure. We have shown how to adapt these ideas to both the technical requirements of the PCFG and the language (German) which is being modelled. In a machine evaluation the PCFG performed clearly better than the baseline, but it was worse than an HMM. A human evaluation, however, showed that on those sentences where both evaluators were of the same opinion, the PCFG is just as good as the HMM. We discussed some of the problems that need to be overcome for PCFGs to successfully compete with HMMs, and we pointed out some advantages of PCFGs which do not necessarily show up in the evaluation results.

10. References

[1] Selkirk, E., 1981. On prosodic structure and its relation to syntactic structure. In *Nordic Prosody II: Papers from a Symposium* (T. Fretheim, ed.), Trondheim: Tapir.

[2] Nespor, M.; Vogel, I., 1986. *Prosodic Phonology*. Foris publications.

[3] Schweitzer, A.;Haase, M., 2000. Zwei Ansätze zur syntaxgesteuerten Prosodiegenerierung. *KONVENS 2000 - Sprachkommunikation*, (Berlin), VDE-Verlag.

[4] Schmid, H., 2000. LoPar: Design and implementation. Arbeitspapiere des Sonderforschungsbereichs 340, No. 149, tech. rep., IMS Stuttgart.

[5] Lari, K.; Young, S.J., 1990. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*,4, 35–56.

[6] Baum, L.E., 1972. An inequality and associated maximization technique in statistical estimation for probabilistic functions of markov processes. *Inequalities*, III, 1–8.

[7] Schiller, A; Teufel, S; Stöckert, C.,1995. Vorläufige Guidelines für das Tagging deutscher Textcorpora mit STTS. IMS, Universität Stuttgart. `http://www.ims.uni-stuttgart.de/ftp/pub/corpora/stts_guide.ps.gz`.

[8] Rapp, S., 1998. *Automatische Erstellung von Korpora für die Prosodieforschung*. PhD thesis, IMS, Universität Stuttgart.

[9] Taylor, P.; Black, A., 1998. Assigning phrase breaks from part-of-speech sequences, *Computer Speech and Language*, 12, 99–117.

[10] Atterer, M., 2000. Assigning prosodic structure for speech synthesis via syntax-prosody mapping. Master's thesis, Division of Informatics, University of Edinburgh.

[11] Knowles, G.; Williams, B.; Taylor, L.,eds., 1996. *A Corpus of Formal British English Speech: The Lancaster/IBM Spoken English Corpus*. London: Longman.

[12] Beckman, M.E.; Ayers, G.M., 1993. Guidelines for ToBI labelling. `http://www.ling.ohio-state.edu/~tobi/ame_tobi/`.