Applying a Grammar-based Language Model to a Simplified Broadcast-News Transcription Task

Tobias Kaufmann
Speech Processing Group
ETH Zürich
Zürich, Switzerland
kaufmann@tik.ee.ethz.ch

Beat Pfister
Speech Processing Group
ETH Zürich
Zürich, Switzerland
pfister@tik.ee.ethz.ch

Abstract

We propose a language model based on a precise, linguistically motivated grammar (a hand-crafted Head-driven Phrase Structure Grammar) and a statistical model estimating the probability of a parse tree. The language model is applied by means of an N-best rescoring step, which allows to directly measure the performance gains relative to the baseline system without rescoring. To demonstrate that our approach is feasible and beneficial for non-trivial broad-domain speech recognition tasks, we applied it to a simplified German broadcast-news transcription task. We report a significant reduction in word error rate compared to a state-of-the-art baseline system.

1 Introduction

It has repeatedly been pointed out that N-grams model natural language only superficially: an Nth-order Markov chain is a very crude model of the complex dependencies between words in an utterance. More accurate statistical models of natural language have mainly been developed in the field of statistical parsing, e.g. Collins (2003), Charniak (2000) and Ratnaparkhi (1999). Other linguistically inspired language models like Chelba and Jelinek (2000) and Roark (2001) have been applied to continuous speech recognition.

These models have in common that they explicitly or implicitly use a context-free grammar induced from a treebank, with the exception of Chelba and Jelinek (2000). The probability of a rule expansion or parser operation is conditioned on various contextual information and the derivation history. An important reason for the success of these models is the fact that they are lexicalized: the probability distributions are also conditioned on the actual words occuring in the utterance, and not only on their parts of speech. Most statistical parsers achieve a high robustness with respect to out-of-grammar sentences by allowing for arbitrary derivations and rule expansions. On the other hand, they are not suited to reliably decide on the grammaticality of a given phrase, as they do not accurately model the linguistic constraints inherent in natural language.

We take a completely different position. In the first place, we want our language model to reliably distinguish between grammatical and ungrammatical phrases. To this end, we have developed a precise, linguistically motivated grammar. To distinguish between common and uncommon phrases, we use a statistical model that estimates the probability of a phrase based on the syntactic dependencies established by the parser. We achieve some degree of robustness by letting the grammar accept arbitrary sequences of words and phrases. To keep the grammar restrictive, such sequences are penalized by the statistical model.

Accurate hand-crafted grammars have been applied to speech recognition before, e.g. Kiefer et al. (2000) and van Noord et al. (1999). However, they primarily served as a basis for a speech understanding component and were applied to narrow-domain tasks such as appointment scheduling or public transport information. We are mainly concerned with speech recognition performance on broad-domain recognition tasks.

Beutler et al. (2005) pursued a similar approach.

Proceedings of ACL-08: HLT, pages 106–113,
Columbus, Ohio, USA, June 2008. ©2008 Association for Computational Linguistics

However, their grammar-based language model did not make use of a probabilistic component, and it was applied to a rather simple recognition task (dictation texts for pupils read and recorded under good acoustic conditions, no out-of-vocabulary words). Besides proposing an improved language model, this paper presents experimental results for a much more difficult and realistic task and compares them to the performance of a state-of-the-art baseline system.

In the following Section, we will first describe our grammar-based language model. Next, we will turn to the linguistic components of the model, namely the grammar, the lexicon and the parser. We will point out some of the challenges arising from the broad-domain speech recognition application and propose ways to deal with them. Finally, we will describe our experiments on broadcast news data and discuss the results.

2 Language Model

2.1 The General Approach

Speech recognizers choose the word sequence $\hat{W}$ which maximizes the posterior probability $P(W|O)$, where O is the acoustic observation. This is achieved by optimizing

$$\hat{W} = \underset{W}{\operatorname{argmax}}\, P(O|W) \cdot P(W)^{\lambda} \cdot ip^{|W|} \qquad (1)$$

The language model weight λ and the word insertion penalty ip lead to a better performance in practice, but they have no theoretical justification. Our grammar-based language model is incorporated into the above expression as an additional probability $P_{gram}(W)$, weighted by a parameter μ:

$$\hat{W} = \underset{W}{\operatorname{argmax}}\, P(O|W) \cdot P(W)^{\lambda} \cdot P_{gram}(W)^{\mu} \cdot ip^{|W|} \qquad (2)$$

$P_{gram}(W)$ is defined as the probability of the most likely parse tree of a word sequence W:

$$P_{gram}(W) = \max_{T \in parses(W)} P(T) \qquad (3)$$

To determine $P_{gram}(W)$ is an expensive operation as it involves parsing. For this reason, we pursue an N-best rescoring approach. We first produce the N best hypotheses according to the criterion in equation (1). From these hypotheses we then choose the final recognition result according to equation (2).

2.2 The Probability of a Parse Tree

The parse trees produced by our parser are binary-branching and rather deep. In order to compute the probability of a parse tree, it is transformed to a flat dependency tree similar to the syntax graph representation used in the TIGER treebank Brants et al (2002). An inner node of such a dependency tree represents a constituent or phrase. Typically, it directly connects to a leaf node representing the most important word of the phrase, the *head child*. The other children represent phrases or words directly depending on the head child. To give an example, the immediate children of a sentence node are the finite verb (the head child), the adverbials, the subject and the all other (verbal and non-verbal) complements.

This flat structure has the advantage that the information which is most relevant for the head child is represented within the locality of an inner node. Assuming statistical independence between the internal structures of the inner nodes n_i, we can factor $P(T)$ much like it is done for probabilistic context-free grammars:

$$P(T) \approx \prod_{n_i} P(\, childtags(n_i) \,|\, tag(n_i)\,) \qquad (4)$$

In the above equation, $tag(n_i)$ is simply the label assigned to the tree node n_i, and $childtags(n_i)$ denotes the tags assigned to the child nodes of n_i.

Our statistical model for German sentences distinguishes between eight different tags. Three tags are used for different types of noun phrases: pronominal NPs, non-pronominal NPs and prenominal genitives. Prenominal genitives were given a dedicated tag because they are much more restricted than ordinary NPs. Another two tags were used to distinguish between clauses with sentence-initial finite verbs (main clauses) and clauses with sentence-final finite verbs (subordinate clauses). Finally, there are specific tags for infinitive verb phrases, adjective phrases and prepositional phrases.

P was modeled by means of a dedicated probability distribution for each conditioning tag. The probability of the internal structure of a sentence was modeled as the trigram probability of the corresponding tag sequence (the sequence of the sentence node's child tags). The probability of an adjective phrase was decomposed into the probability

of the adjective type (participle or non-participle and attributive, adverbial or predicative) and the probability of its length in words given the adjective type. This allows the model to directly penalize long adjective phrases, which are very rare. The model for noun phrases is based on the joint probability of the head type (either noun, adjective or proper name), the presence of a determiner and the presence of pre- and postnominal modifiers. The probabilities of various other events are conditioned on those four variables, namely the number of prepositional phrases, relative clauses and adjectives, as well as the presence of appositions and prenominal or postnominal genitives.

The resulting probability distributions were trained on the German TIGER treebank which consists of about 50000 sentences of newspaper text.

2.3 Robustness Issues

A major problem of grammar-based approaches to language modeling is how to deal with out-of-grammar utterances. Obviously, the utterance to be recognized may be ungrammatical, or it could be grammatical but not covered by the given grammar. But even if the utterance is both grammatical and covered by the grammar, the correct word sequence may not be among the N best hypotheses due to out-of-vocabulary words or bad acoustic conditions. In all these cases, the best hypothesis available is likely to be out-of-grammar, but the language model should nevertheless prefer it to competing hypotheses. To make things worse, it is not unlikely that some of the competing hypotheses are grammatical.

It is therefore important that our language model is robust with respect to out-of-grammar sentences. In particular this means that it should provide a reasonable parse tree for any possible word sequence W. However, our approach is to use an accurate, linguistically motivated grammar, and it is undesirable to weaken the constraints encoded in the grammar. Instead, we allow the parser to attach any sequence of words or correct phrases to the root node, where each attachment is penalized by the probabilistic model $P(T)$. This can be thought of as adding two probabilistic context-free rules:

$$S \longrightarrow S' \;\; S \quad \text{with probability } q$$
$$S \longrightarrow S' \quad \text{with probability } 1-q$$

In order to guarantee that all possible word sequences are parseable, S' can produce both saturated phrases and arbitrary words. To include such a productive set of rules into the grammar would lead to serious efficiency problems. For this reason, these rules were actually implemented as a dynamic programming pass: after the parser has identified all correct phrases, the most probable sequence of phrases or words is computed.

2.4 Model Parameters

Besides the distributions required to specify $P(T)$, our language model has three parameters: the language model weight μ, the attachment probability q and the number of hypotheses N. The parameters μ and q are considered to be task-dependent. For instance, if the utterances are well-covered by the grammar and the acoustic conditions are good, it can be expected that μ is relatively large and that q is relatively small. The choice of N is restricted by the available computing power. For our experiments, we chose $N = 100$. The influence of N on the word error rate is discussed in the results section.

3 Linguistic Resources

3.1 Particularities of the Recognizer Output

The linguistic resources presented in this Section are partly influenced by the form of the recognizer output. In particular, the speech recognizer does not always transcribe numbers, compounds and acronyms as single words. For instance, the word *"einundzwanzig"* (twenty-one) is transcribed as *"ein und zwanzig"*, *"Kriegspläne"* (war plans) as *"Kriegs Pläne"* and *"BMW"* as *"B. M. W."* These transcription variants are considered to be correct by our evaluation scheme. Therefore, the grammar should accept them as well.

3.2 Grammar and Parser

We used the Head-driven Phrase Structure Grammar (HPSG, see Pollard and Sag (1994)) formalism to develop a precise large-coverage grammar for German. HPSG is an unrestricted grammar (Chomsky type 0) which is based on a context-free skeleton and the unification of complex feature structures. There are several variants of HPSG which mainly differ in the formal tools they provide for stating lin-

guistic constraints. Our particular variant requires that constituents (phrases) be continuous, but it provides a mechanism for dealing with discontinuities as present e.g. in the German main clause, see Kaufmann and Pfister (2007). HPSG typically distinguishes between *immediate dominance schemata* (rough equivalents of phrase structure rules, but making no assumptions about constituent order) and *linear precedence rules* (constraints on constituent order). We do not make this distinction but rather let immediate dominance schemata specify constituent order. Further, the formalism allows to express complex linguistic constraints by means of *predicates* or *relational constraints*. At parse time, predicates are backed by program code that can perform arbitrary computations to check or specify feature structures.

We have implemented an efficient Java parser for our variant of the HPSG formalism. The parser supports ambiguity packing, which is a technique for merging constituents with different derivational histories but identical syntactic properties. This is essential for parsing long and ambiguous sentences.

Our grammar incorporates many ideas from existing linguistic work, e.g. Müller (2007), Müller (1999), Crysmann (2005), Crysmann (2003). In addition, we have modeled a few constructions which occur frequently but are often neglected in formal syntactic theories. Among them are prenominal and postnominal genitives, expressions of quantity and expressions of date and time. Further, we have implemented dedicated subgrammars for analyzing written numbers, compounds and acronyms that are written as separate words. To reduce ambiguity, only noun-noun compounds are covered by the grammar. Noun-noun compounds are by far the most productive compound type.

The grammar consists of 17 rules for general linguistic phenomena (e.g. subcategorization, modification and extraction), 12 rules for modeling the German verbal complex and another 13 construction-specific rules (relative clauses, genitive attributes, optional determiners, nominalized adjectives, etc.). The various subgrammars (expressions of date and time, written numbers, noun-noun compounds and acronyms) amount to a total of 43 rules.

The grammar allows the derivation of "intermediate products" which cannot be regarded as complete phrases. We consider complete phrases to be sentences, subordinate clauses, relative and interrogative clauses, noun phrases, prepositional phrases, adjective phrases and expressions of date and time.

3.3 Lexicon

The lexicon was created manually based on a list of more than 5000 words appearing in the N-best lists of our experiment. As the domain of our recognition task is very broad, we attempted to include any possible reading of a given word. Our main source of dictionary information was Duden (1999).

Each word was annotated with precise morphological and syntactic information. For example, the roughly 2700 verbs were annotated with over 7000 valency frames. We distinguish 86 basic valency frames, for most of which the complement types can be further specified.

A major difficulty was the acquisition of *multiword lexemes*. Slightly deviating from the common notion, we use the following definition: A syntactic unit consisting of two or more words is a multiword lexeme, if the grammar cannot derive it from its parts. English examples are idioms like *"by and large"* and phrasal verbs such as *"to call sth off"*. Such multi-word lexemes have to be entered into the lexicon, but they cannot directly be identified in the word list. Therefore, they have to be extracted from supplementary resources. For our work, we used a newspaper text corpus of 230M words (Frankfurter Rundschau and Neue Zürcher Zeitung). This corpus included only articles which are dated before the first broadcast news show used in the experiment. In the next few paragraphs we will discuss some types of multiword lexemes and our methods of extracting them.

There is a large and very productive class of German prefix verbs whose prefixes can appear separated from the verb, similar to English phrasal verbs. For example, the prefix of the verb *"untergehen"* (to sink) is separated in *"das Schiff geht unter"* (the ship sinks) and attached in *"weil das Schiff untergeht"* (because the ship sinks). The set of possible valency frames of a prefix verb has to be looked up in a dictionary as it cannot be derived systematically from its parts. Exploiting the fact that prefixes are attached to their verb under certain circumstances, we extracted a list of prefix verbs from the above newspaper text corpus. As the number of prefix verbs is

very large, a candidate prefix verb was included into the lexicon only if there is a recognizer hypothesis in which both parts are present. Note that this procedure does not amount to optimizing on test data: when parsing a hypothesis, the parser chart contains only those multiword lexemes for which all parts are present in the hypothesis.

Other multi-word lexemes are fixed word clusters of various types. For instance, some prepositional phrases appearing in support verb constructions lack an otherwise mandatory determiner, e.g. *"unter Beschuss"* (under fire). Many multi-word lexemes are adverbials, e.g. *"nach wie vor"* (still), *"auf die Dauer"* (in the long run). To extract such word clusters we used suffix arrays proposed in Yamamoto and Church (2001) and the pointwise mutual information measure, see Church and Hanks (1990). Again, it is feasible to consider only those clusters appearing in some recognizer hypothesis. The list of candidate clusters was reduced using different filter heuristics and finally checked manually.

For our task, split compounds are to be considered as multi-word lexemes as well. As our grammar only models noun-noun compounds, other compounds such as *"unionsgeführt"* (led by the union) have to be entered into the lexicon. We applied the decompounding algorithm proposed in Adda-Decker (2003) to our corpus to extract such compounds. The resulting candidate list was again filtered manually.

We observed that many proper nouns (e.g. personal names and geographic names) are identical to some noun, adjective or verb form. For example, about 40% of the nouns in our lexicon share inflected forms with personal names. Proper nouns considerably contribute to ambiguity, as most of them do not require a determiner. Therefore, a proper noun which is a homograph of an open-class word was entered only if it is "relevant" for our task. The "relevant" proper nouns were extracted automatically from our text corpus. We used small databases of unambiguous given names and forms of address to spot personal names in significant bigrams. Relevant geographic names were extracted by considering capitalized words which significantly often follow certain local prepositions.

The final lexicon contains about 2700 verbs (including 1900 verbs with separable prefixes), 3500 nouns, 450 adjectives, 570 closed-class words and 220 multiword lexemes. All lexicon entries amount to a total of 137500 full forms. Noun-noun compounds are not included in these numbers, as they are handled in a morphological analysis component.

4 Experiments

4.1 Experimental Setup

The experiment was designed to measure how much a given speech recognition system can benefit from our grammar-based language model. To this end, we used a baseline speech recognition system which provided the N best hypotheses of an utterance along with their respective scores. The grammar-based language model was then applied to the N best hypotheses as described in Section 2.1, yielding a new best hypothesis. For a given test set we could then compare the word error rate of the baseline system with that of the extended system employing the grammar-based language model.

4.2 Data and Preprocessing

Our experiments are based on word lattice output from the LIMSI German broadcast news transcription system (McTait and Adda-Decker, 2003), which employs 4-gram backoff language models. From the experiment reported in McTait and Adda-Decker (2003), we used the first three broadcast news shows[1] which corresponds to a signal length of roughly 50 minutes.

Rather than applying our model to the original broadcast-news transcription task, we used the above data to create an artificial recognition task with manageable complexity. Our primary aim was to design a task which allows us to investigate the properties of our grammar-based approach and to compare its performance with that of a competitive baseline system.

As a first simplification, we assumed perfect sentence segmentation. We manually split the original word lattices at the sentence boundaries and merged them where a sentence crossed a lattice boundary. This resulted in a set of 636 lattices (sentences). Second, we classified the sentences with respect to content type and removed those classes with an excep-

[1] The 8 o'clock broadcasts of the "Tagesschau" from the 14^{th} of April, 21^{st} of April and 7^{th} of Mai 2002.

tionally high baseline word error rate. These classes are interviews (a word error rate of 36.1%), sports reports (28.4%) and press conferences (25.7%). The baseline word error rate of the remaining 447 lattices (sentences) is 11.8%.

From each of these 447 lattices, the 100 best hypotheses were extracted. We next compiled a list containing all words present in the recognizer hypotheses. These words were entered into the lexicon as described in Section 3.3. Finally, all extracted recognizer hypotheses were parsed. Only 25 of the 44000 hypotheses[2] caused an early termination of the parser due to the imposed memory limits. However, the inversion of ambiguity packing (see Section 3.2) turned out to be a bottleneck. As $P(T)$ does not directly apply to parse trees, all possible readings have to be unpacked. For 24 of the 447 lattices, some of the N best hypotheses contained phrases with more than 1000 readings. For these lattices the grammar-based language model was simply switched off in the experiment, as no parse trees were produced for efficiency reasons.

To assess the difficulty of our task, we inspected the reference transcriptions, the word lattices and the N-best lists for the 447 selected utterances. We found that for only 59% of the utterances the correct transcription is among the 100-best hypotheses. The first-best hypothesis is completely correct for 34% of the utterances. The out-of-vocabulary rate (estimated from the number of reference transcription words which do not appear in any of the lattices) is 1.7%. The first-best word error rate is 11.79%, and the 100-best oracle word error rate is 4.8%.

We further attempted to judge the grammaticality of the reference transcriptions. We considered only 1% of the sentences to be clearly ungrammatical. 19% of the remaining sentences were found to contain general grammatical constructions which are not handled by our grammar. Some of these constructions (most notably ellipses, which are omnipresent in broadcast-news reports) are notoriously difficult as they would dramatically increase ambiguity when implemented in a grammar. About 45% of the reference sentences were correctly analyzed by the grammar.

[2]Some of the word lattices contain less than 100 different hypotheses.

4.3 Training and Testing

The parameter N, the maximum number of hypotheses to be considered, was set to 100 (the effect of choosing different values of N will be discussed in section 4.4). The remaining parameters μ and q were trained using the leave-one-out cross-validation method: each of the 447 utterances served as the single test item once, whereas the remaining 446 utterances were used for training. As the error landscape is complex and discrete, we could not use gradient-based optimization methods. Instead, we chose μ and q from 500 equidistant points within the intervals $[0, 20]$ and $[0, 0.25]$, respectively. The word error rate was evaluated for each possible pair of parameter values.

The evaluation scheme was taken from McTait and Adda-Decker (2003). It ignores capitalization, and written numbers, compounds and acronyms need not be written as single words.

4.4 Results

As shown in Table 1, the grammar-based language model reduced the word error rate by 9.2% relative over the baseline system. This improvement is statistically significant on a level of $< 0.1\%$ for both the Matched Pairs Sentence-Segment Word Error test (MAPSSWE) and McNemar's test (Gillick and Cox, 1989). If the parameters are optimized on all 447 sentences (i.e. on the test data), the word error rate is reduced by 10.7% relative.

For comparison, we redefined the probabilistic model as $P(T) = (1 - q)q^{k-1}$, where k is the number of phrases attached to the root node. This reduced model only considers the grammaticality of a phrase, completely ignoring the probability of its internal structure. It achieved a relative word error reduction of 5.9%, which is statistically significant on a level of $< 0.1\%$ for both tests. The improvement of the full model compared to the reduced model is weakly significant on a level of 2.6% for the MAPSSWE test.

For both models, the optimal value of q was 0.001 for almost all training runs. The language model weight μ of the reduced model was about 60% smaller than the respective value for the full model, which confirms that the full model provides more reliable information.

experiment	word error rate
baseline	**11.79%**
grammar, no statistics	11.09% (-5.9% rel.)
grammar	**10.70%** (-9.2% rel.)
grammar, cheating	10.60% (-10.7% rel.)
100-best oracle	4.80%

Table 1: The impact of the grammar-based language model on the word error rate. For comparison, the results for alternative experiments are shown. In the experiment "*grammar, cheating*", the parameters were optimized on test data.

Figure 1 shows the effect of varying N (the maximum number of hypotheses) on the word error rate both for leave-one-out training and for optimizing the parameters on test data. The similar shapes of the two curves suggest that the observed variations are partly due to the problem structure. In fact, if N is increased and new hypotheses with a high value of $P_{gram}(W)$ appear, the benefit of the grammar-based language model can increase (if the hypotheses are predominantly good with respect to word error rate) or decrease (if they are bad). This horizon effect tends to be reduced with increasing N (with the exception of $89 \leq N \leq 93$) because hypotheses with high ranks need a much higher $P_{gram}(W)$ in order to compensate for their lower value of $P(O|W) \cdot P(W)^{\lambda}$. For small N, the parameter estimation is more severely affected by the rather accidental horizon effects and therefore is prone to overfitting.

5 Conclusions and Outlook

We have presented a language model based on a precise, linguistically motivated grammar, and we have successfully applied it to a difficult broad-domain task.

It is a well-known fact that natural language is highly ambiguous: a correct and seemingly unambiguous sentence may have an enormous number of readings. A related – and for our approach even more relevant – phenomenon is that many weird-looking and seemingly incorrect word sequences are in fact grammatical. This obviously reduces the benefit of pure grammaticality information. A solution is to use additional information to asses how "natural" a reading of a word sequence is. We have done a

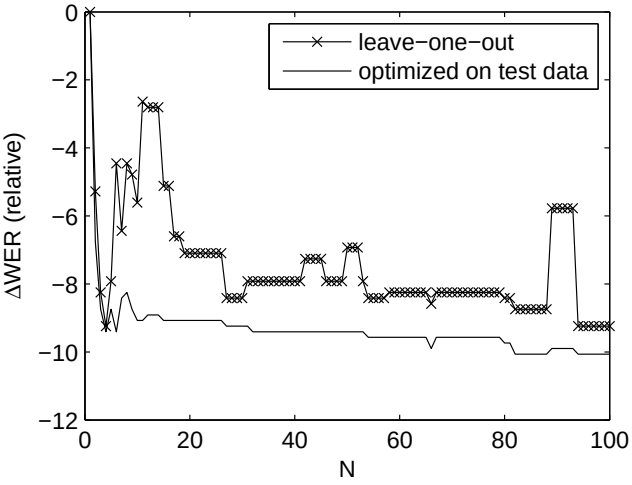

Figure 1: The word error rate as a function of the maximum number of best hypotheses N.

first step in this direction by estimating the probability of a parse tree. However, our model only looks at the structure of a parse tree and does not take the actual words into account. As N-grams and statistical parsers demonstrate, word information can be very valuable. It would therefore be interesting to investigate ways of introducing word information into our grammar-based model.

Acknowledgements

This work was supported by the Swiss National Science Foundation. We cordially thank Jean-Luc Gauvain of LIMSI for providing us with word lattices from their German broadcast news transcription system.

References

M. Adda-Decker. 2003. A corpus-based decompounding algorithm for German lexical modeling in LVCSR. In *Proceedings of Eurospeech*, pages 257–260, Geneva, Switzerland.

R. Beutler, T. Kaufmann, and B. Pfister. 2005. Integrating a non-probabilistic grammar into large vocabulary continuous speech recognition. In *Proceedings of the IEEE ASRU 2005 Workshop*, pages 104–109, San Juan (Puerto Rico).

S. Brants, S. Dipper, S. Hansen, W. Lezius, and G. Smith. 2002. The TIGER treebank. In *Proceedings of the Workshop on Treebanks and Linguistic Theories*, Sozopol, Bulgaria.

E. Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the NAACL*, pages 132–139, San Francisco, USA.

C. Chelba and F. Jelinek. 2000. Structured language modeling. *Computer Speech & Language*, 14(4):283–332.

K. W. Church and P. Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

M. Collins. 2003. Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29(4):589–637.

B. Crysmann. 2003. On the efficient implementation of German verb placement in HPSG. In *Proceedings of RANLP*.

B. Crysmann. 2005. Relative clause extraposition in German: An efficient and portable implementation. *Research on Language and Computation*, 3(1):61–82.

Duden. 1999. *– Das große Wörterbuch der deutschen Sprache in zehn Bänden*. Dudenverlag, dritte Auflage.

L. Gillick and S. Cox. 1989. Some statistical issues in the comparison of speech recognition algorithms. In *Proceedings of the ICASSP*, pages 532–535.

T. Kaufmann and B. Pfister. 2007. Applying licenser rules to a grammar with continuous constituents. In Stefan Müller, editor, *The Proceedings of the 14th International Conference on Head-Driven Phrase Structure Grammar*, pages 150–162, Stanford, USA. CSLI Publications.

B. Kiefer, H.-U. Krieger, and M.-J. Nederhof. 2000. Efficient and robust parsing of word hypotheses graphs. In Wolfgang Wahlster, editor, *Verbmobil. Foundations of Speech-to-Speech Translation*, pages 280–295. Springer, Berlin, Germany, artificial intelligence edition.

K. McTait and M. Adda-Decker. 2003. The 300k LIMSI German broadcast news transcription system. In *Proceedings of Eurospeech*, Geneva, Switzerland.

S. Müller. 1999. *Deutsche Syntax deklarativ. Head-Driven Phrase Structure Grammar für das Deutsche*. Number 394 in Linguistische Arbeiten. Max Niemeyer Verlag, Tübingen.

S. Müller. 2007. *Head-Driven Phrase Structure Grammar: Eine Einführung*. Stauffenburg Einführungen, Nr. 17. Stauffenburg Verlag, Tübingen.

G. Van Noord, G. Bouma, R. Koeling, and M.-J. Nederhof. 1999. Robust grammatical analysis for spoken dialogue systems. *Natural Language Engineering*, 5(1):45–93.

C. J. Pollard and I. A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago.

A. Ratnaparkhi. 1999. Learning to parse natural language with maximum entropy models. *Machine Learning*, 34(1-3):151–175.

B. Roark. 2001. Probabilistic top-down parsing and language modeling. *Computational Linguistics*, 27(2):249–276.

M. Yamamoto and K. W. Church. 2001. Using suffix arrays to compute term frequency and document frequency for all substrings in a corpus. *Computational Linguistics*, 27(1):1–30.