Experiments in German Treebank Parsing

Sisay Fissaha[1], Daniel Olejnik[2], Ralf Kornberger[2], Karin Müller[3], and Detlef Prescher[3]

[1] IAI, Saarland University, Martin-Luther Str. 14, 66111 Saarbrücken
sisay@iai.uni-sb.de
[2] Computer Science Dep., Saarland University, PO Box 151150, 66041 Saarbrücken
{kaligula, kornberg}@coli.uni-sb.de
[3] ILLC, University of Amsterdam, Nieuwe Achtergracht 166, 1018 WV Amsterdam
{kmueller, prescher}@science.uva.nl

Abstract. State-of-the-art parsers for English are based on probabilistic grammars induced from treebanks. German, a language with a free-er word order and a richer morphology, is likely to pose new problems to techniques developed especially for English. This paper describes one of the first parsing experiments using probabilistic context-free grammars extracted from the German treebank NEGRA. Particularly, we introduce "partial-parent encoding" for NEGRA, a variant of the parent-encoding technique. Evaluation shows that grammatical functions, and partial-parent encoding improve the performance of a parser for German. Moreover, we find evidence that extending the size of the treebank as well as adding more linguistic information will improve the results.

1 Introduction

In the field of probabilistic treebank parsing, a lot of research has been invested in the search of well performing training and parsing strategies (e.g. [5], [6]). Less work has been dedicated to the study and the production of the required grammars, although it has been recognized that the issue of automatically extracting efficient grammars from treebanks is of central importance for further progress in the design of high performance statistical parsers [8]. Many of the empirical studies have been performed on English using the Penn treebank [10]. German, however, a language with a free-er word order and a richer morphology, is likely to pose new problems for techniques that have been developed on English.

In this paper, we report on a series of parsing experiments carried out on the NEGRA treebank [17]. NEGRA differs in important respects from the Penn treebank: the trees are relatively flat, but are annotated by a rich system of grammatical functions.

So, the main interest of the current paper is threefold: First, we investigate whether techniques of automatically extracting efficient grammars from treebanks - successfully applied to English by using Penn treebank - can be tailored to a new language and to a treebank with a different structure. Second, we apply a well-established evaluation measure, PARSEVAL [2], in order to provide a

common ground of comparison for future experiments on parsing German. Third, we study the influence on parsing performance of various choices concerning the automated production of the grammar from NEGRA, e.g., presence/absence of grammatical functions. Specifically, we investigate the potential of using the rich linguistic information offered by NEGRA for developing probabilistic parsers for German with different levels of complexity. The complexity of the parsers relates to the type of linguistic information encoded in the treebank, which is systematically added to the grammars.

Our work uses similar ideas like [5] and [8]. It combines Charniak's probability model with a context-encoding technique proposed by Johnson. Both experiments were carried out on English and the Penn Treebank [10]. However, the different structure of German language expressed by a flatter structure of NEGRA, and a richer syntactic information requires that the approaches of Charniak and Johnson has to be tailored to German. Particularly, we investigate the effect of incorporating grammatical functions, and a variant of Johnson's context encoding technique, which we call "partial parent-encoding".
Most of the grammars previously developed for German are either restricted to specific language constructs such as verb final or relative clauses, topological fields, chunks, etc.([1], [9], [4], [15], [3]) or are confined to specific grammar formalisms ([13]) whereas others describe combinations of deep and shallow parsing ([7], [11]).

We believe that little attention was paid in the past to the use of an existing treebank in order to develop a robust and broad-coverage probabilistic PCFG for German. Compared to earlier work on developing a parser for German, the advantages of our approach are: First, it does not require manual grammar development since it makes use of an already existing resource, the NEGRA treebank, enabling short development time. Second, the results are evaluated using PARSEVAL values providing detailed information about the different aspects of our parser. This has the further advantage of enabling comparison of our parser with other state-of-the-art parsers. So far, German grammars were either evaluated with respect to partial analyzes (chunks, noun phrases, verb phrases, frames etc), or could be hardly evaluated with PARSEVAL measures (like LFG or HPSG parsers). Finally, NEGRA grammars provide fine-grained analyzes without jeopardizing coverage significantly.

2 The NEGRA Treebank

Our experiments are carried out on the NEGRA treebank, a corpus of syntactically annotated sentences taken from a German newspaper. Currently, NEGRA contains about 20,000 sentences. The annotations are theory independent, but at the same time flexible enough to allow for the generation of theory specific representations.

In NEGRA, different types of linguistic information such as grammatical functions, syntactic and morphological categories are encoded. Argument structure is expressed using trees with crossing branches resulting in rather flat trees.

More information is expressed by a rich system of function labels describing, for example, complements and modifiers, or headed and non-headed structures.

For our experiments, we use the so-called "Penn Format" of the NEGRA treebank, which gives us a common ground to compare our results with the very first grammars trained on the Penn treebank. The Penn format of NEGRA makes use of a context-free backbone, where crossing-branches are substituted by traces in order to express such relations as local and non-local dependencies or discontinuous constituents.

3 Experiments with Probabilistic NEGRA Grammars

The simplest way of developing a grammar is to read off the grammar from a treebank, a corpus of manually annotated sentences. We [1] use NEGRA as our treebank, which we randomly divide into two separate corpora: 2 000 sentences for testing and 18 000 for training. In contrast to Charniak and others, we do not delete the sentences of length greater than 40 in the test data.

As in [5], traces are ignored, and a new start symbol is added, since all trees in the NEGRA treebank bear an empty top node. Furthermore, we replace all lexical items in the trees with their respective part-of-speech tags. Here, we use STTS tags [18] taken from the preterminal nodes of the lexical items. No further changes are made.

To create a PCFG, we apply the "relative frequency estimation" technique of [5]. If $f(r)$ is the number of times a rule r occurs in the training corpus, and $lhs(r)$ is the non-terminal that r expands, then the probability assigned to r is:

$$p(r) = \frac{f(r)}{\sum_{lhs(r')=lhs(r)} f(r')} \ .$$

Using the LoPar parser [14] (without using its smoothing option), we obtain the maximum-probability parse of each sentence, and compare it to the one given in the testing corpus. Using the scoring tool evalb [16] (without specifying any options in its parameter file), we evaluate our PCFGs by the following PARSEVAL [2] measures: **Labeled precision**, the percentage of labeled non-terminal brackets in the maximum-probability parse that also appeared in the treebank parse; **Labeled recall**, the percentage of labeled non-terminal brackets from the treebank that also appeared in the maximum-probability parse; **Exact-match rate**, the percentage of maximum-probability parses exactly matching their counterpart in the treebank. To simplify our figures, we report exact-match rate and **labeled f-measure** which is defined as the harmonic mean of labeled precision and labeled recall.

As mentioned in Section 2, the non-terminals of the NEGRA grammars consist of a combination of constituent and function labels. Thus, the PARSEVAL measures evaluate the performance of the NEGRA grammars with respect to both constituents and functions.

[1] The experiments were conducted by the participants (advanced students and supervisors) of the seminar *Training and Evaluation of Probabilistic Context-Free Grammars* given at the Computational Linguistics Department of Saarland University.

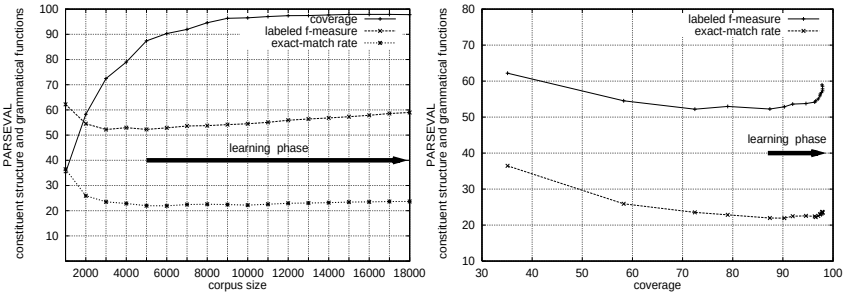


Fig. 1. Learning curves for NEGRA grammars providing constituent structure and grammatical functions: PARSEVAL measures and grammar coverage in the number of sentences used for training (left-hand side), and PARSEVAL measures in grammar coverage (right-hand side).

3.1 Main Result

The grammar obtained consists of 28,773 rules of which 7,426 occur more than once. The following fragment of the NEGRA grammar shows, for example, that noun phrases consisting of a determiner and a noun occur more often as subjects than as direct or indirect objects:

$$\begin{aligned} \text{NP-SB} &\longrightarrow \text{ART-NK NN-NK} \quad (3{,}498) \\ \text{NP-OA} &\longrightarrow \text{ART-NK NN-NK} \quad (1{,}385) \\ \text{NP-DA} &\longrightarrow \text{ART-NK NN-NK} \quad (338) \end{aligned}$$

The left-hand side of Figure 1 displays the grammar coverage, the labeled f-measure, and the exact-match rate versus the number of sentences used for training. Absolute values for grammar coverage range from 35% (1,000 sentences) to 98% (18,000 sentences). The coverage monotonically increases, and training on 6,000 sentences is enough to achieve a grammar coverage of more than 90%. Absolute values for labeled f-measure and exact-match range from 52% to 62%, and from 22% to 37% respectively. A clear learning effect occurs: The curves for both measures monotonically increase, when training is conducted on 5,000 sentences or more. An over-training effect was not observed.

The right-hand side of Figure 1 displays both curves again, but this time as a function of the grammar coverage. So, absolute values for the labeled f-measure and the exact-match remain the same. The figure reveals, however, that the learning effect is directly related to the coverage of the grammars: Labeled f-measure and exact-match rate monotonically increase, when the grammars achieve at least 87% coverage, i.e., when the symbolic grammars are broad enough.

3.2 Effect of Grammatical Functions

In this paragraph, we concentrate on predicting the constituent structure of German sentences using grammar rules with and without grammatical functions.

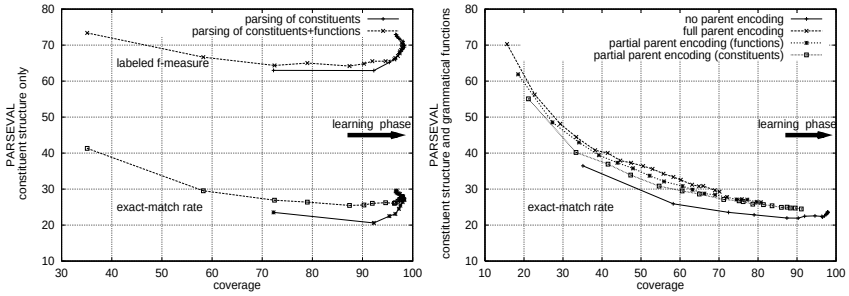


Fig. 2. Left-hand side: The effect of grammatical functions to parsing: PARSEVAL measures for parsing with and without grammatical functions. Right-hand side: the effect of three variants of parent encoding on parsing performance for NEGRA grammars providing constituent structure and grammatical functions.

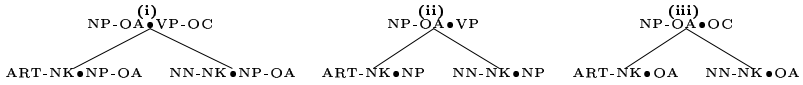
More precisely, we investigate the influence of using grammatical functions to parsing of constituents.

The left-hand side of figure 2 displays the results of our experiments in terms of labeled f-measure and exact-match rate subjected to coverage. On comparable levels of coverage, the grammars trained on functions and constituents perform consistently better than grammars trained on constituents only. On average, labeled f-measure is improved by 2%, and exact-match rate by 5%. Moreover, a clear learning effect occurs for grammars having a high coverage ($\geq$87%).

3.3 Effect of Parent Encoding

[8] showed that the tree representations used in a treebank corpus can have an effect on the performance of the PCFG estimated from that corpus. Johnson demonstrated especially that the so-called parent encoding technique, when applied to the Penn treebank, results in a parser with a significantly improved performance.

Parent encoding is a simple tree-transformation technique, where each preterminal node's parent's category is appended onto its own label. In this paragraph, we apply variants of this technique to NEGRA to encode different context information: (i) full parent encoding using both constituent and function labels, (ii) partial parent encoding using the constituent labels only, and (iii) partial parent encoding using the function labels only. The following (partial) trees display the results of these three variants for a direct-object noun phrase (NP-OA) embedded in a clausal verb-phrase (VP-OC) in the original tree:



The right-hand side of figure 2 displays the performance of the resulting grammars, when trained with these parent-encoding techniques. The evaluation

is carried out on the original trees. For comparison, the result of the experiment without parent encoding is also displayed.

On comparable levels of coverage, the grammars trained with a full or partial parent-encoding technique clearly outperform the grammars obtained without parent-encoding (av. 4-7% higher exact-match). A closer look at the exact-match rates further reveals that full parent encoding yields consistently better results (70-29%) than partial parent encoding with functions (62-26%), which in turn achieves consistently better results than partial parent encoding with constituents (55-25%). From the perspective of the achieved coverage, we observe a reversed order of performance where the experiment without parent encoding achieves the highest score (98%), followed by partial parent encoding with constituents (92%) and grammatical functions (82%), and full parent encoding (70%). Moreover, the curves do not show any learning effect for parent encoding.

3.4 Error Analysis

By analyzing the output of the parser, we observe that labeled recall is consistently higher than labeled precision. This points out that the parser tends to assign more structure to the input than it is actually annotated. We also observe that it is very hard for the parser to predict the internal structure of the sentence node S. The high number of S-rules (27% of the grammar) indicates that NEGRA provides a relatively flat structure. A further problem for the parser is the relatively high ambiguity of the NEGRA grammars which makes it difficult for the parser's probability model to choose the correct analysis.

4 Discussion

We demonstrated that the use of an existing resource like NEGRA is an efficient way of developing a robust (98% coverage) and fine-grained grammar for German. Our experiments substantiate the underlying hypothesis that grammatical functions and parent encoding improve the parsing performance. Although this information seems to be common knowledge in the field of parsing, this work provides a concrete empirical evidence for German: Parsing with constituents and functions recognizes constituent structures significantly better than parsing with constituents only (5% av. in terms of the exact-match rate on comparable levels of grammar coverage). Moreover, the investigated variants of parent encoding improve parsing performance (4-7% av.).

We can also make some concrete comparison of the results in this paper to those in parsing English. First, [5] achieved better labeled precision and recall values for recognition of constituents, however, using a larger training corpus than ours. Second, [8] demonstrated that parent encoding, when applied to the Penn treebank, results in a parser with an improved performance (7% av.). In the present paper, we also observe an improved performance for parent encoding (7% av.). However, this improvement comes at a cost of a dramatically lower coverage (70%). Compared to this, we observe higher coverages for partial-parent

encoding with functions (82% av.) and categories (92% av.), but lower improvements in performance (3-5%). For all findings, however, the different size of the Penn and the NEGRA treebank must be taken into account: Since we observe monotonically increasing learning curves in our experiments, we expect that the difference in performance (between the mentioned Penn and NEGRA grammars) can be reduced by increasing the size of the NEGRA corpus.

To the best of our knowledge, this is the first paper presenting a German grammar based on a PCFG, which focuses on both constituent structure and grammatical functions, and which is evaluated by the PARSEVAL measures. This states a basis for future comparisons of German parsing systems using the widely accepted PARSEVAL evaluation measure for treebank grammars. Although our main results are not directly comparable, we discuss the coverage values of existing parsing systems for German. The most robust NEGRA grammar achieved a coverage of 98% on free text. The German grammars developed manually in declarative formalisms achieve a coverage less than 50% ([13],[7]). The system of [11] provides as ours constituent structure and grammatical functions, and achieves a coverage of 94% on free text. However, the difference of 4% might result from a lower tagging accuracy. There are a few shallow grammars yielding a perfect coverage of 100%, but these grammars are restricted to specific language constructs, whereas our grammar provides relatively fine-grained grammatical analyzes for almost all sentences of German.

Another point worth mentioning relates to the learning effect observed for the different probabilistic parsers. Training of constituents and grammatical functions on the basis of STTS tags shows a clear learning effect. However, all other training methods (e.g. incorporating function tags, or parent encoding) show no learning effect, which indicates a serious sparse data problem. Over-training was not observed in any case, pointing out that it would be worth to extend the NEGRA treebank in terms of both corpus size and linguistic information.

5 Conclusion

Using the NEGRA treebank [17], we developed a robust parser (98% coverage) for German, which still delivers relatively fine-grained analyzes. Our work built on simple PCFGs [5] and parent encoding [8]. We showed that these techniques developed for English can be successfully tailored to German: Partial-parent encoding, if compared to full-parent encoding, resulted in grammars with a dramatically higher coverage and a slightly lower performance. We believe that our findings are useful for future exploration of techniques for languages other than English. We tested our German parsers using a well-established evaluation measure, PARSEVAL [2], in order to provide a common ground of comparison for future experiments on parsing German. We further plan to evaluate dependencies [12]. We showed in our experiments that there is a direct relation between the quality of a parser and the linguistic knowledge incorporated. Grammatical functions resulted in the most significant improvements, which agrees with the design principles of the NEGRA treebank. The learning curves of the probabilis-

tic grammars suggested that further improvements can be achieved by extending the NEGRA treebank. Unfortunately, extending NEGRA is off our hands. Alternatively, we intend to use NEGRA grammars as a starting point for manual grammar development. One way of extending the NEGRA grammars is to incorporate more information at different linguistic levels, e.g., more detailed morphological rules, and lexical semantics for head words.

References

1. Markus Becker and Anette Frank. 2002. A stochastic topological parser of German. In *Proc. of COLING 2002*, Taipei, Taiwan.
2. Ezra Black. 1992. Meeting of interest group on evaluation of broad-coverage grammars of English. LINGUIST List 3.587, http://www.linguistlist.org/issues/3/3-587.html.
3. Thorsten Brants, Wojciech Skut, and Brigitte Krenn. 1997. Tagging Grammtical Functions. In *Proc. of EMNLP 1997*, Providence, RI, USA.
4. Thorsten Brants. 1999. Cascaded Markov Models. In *Proc. of EACL-99*, Bergen.
5. Eugene Charniak. 1996. Tree-bank grammars. Technical Report CS-96-02, Brown University.
6. Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of ACL-1997*, Madrid.
7. B. Crysmann, A. Frank, B. Kiefer, U. Krieger, S. Müller, G. Neumann, J. Piskorski, U. Schäfer, M. Siegel, Uszkoreit H., and F. Xu. 2002. An integrated architecture for shallow and deep processing. In *Proc. of ACL 2002*, UPenn, PA.
8. Mark Johnson. 1998. PCFG models of linguistic tree representations. *Computational Linguistics*, 24(4).
9. Sandra Kübler and Erhard W. Hinrichs. 2001. From chunks to function-argument structure: A similarity-based approach. In *Proc. of ACL 2001*, Toulouse, France.
10. Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. In *Proc. of the ARPA Human Language Technology Workshop*.
11. G. Neumann and J. Piskorski. 2002. A shallow text processing core engine. *Journal of Computational Intelligence*.
12. PARSEVAL. 2002. Beyond PARSEVAL - Towards Improved Evaluation Measures for Parsing Systems. Workshop of the Third LREC Conference 2002, 2nd June.
13. S. Riezler, D. Prescher, J. Kuhn, and M. Johnson. 2000. Lexicalized stochastic modeling of constraint-based grammars using log-linear measures and EM training. In *Proc. of ACL-2000*.
14. Helmut Schmid, 1999. *LoPar. Design and Implementation*. Insitut für Maschinelle Sprachverarbeitung, Universität Stuttgart.
15. Sabine Schulte im Walde and Helmut Schmid. 2000. Robust German noun chunking with a probabilistic context-free grammar. In *Proc. of COLING 2000*.
16. Satoshi Sekine. 1999. Scoring code for the Wall Street Journal Penn treebank. http://www.research.att.com/~mcollins/.
17. Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proc. of ANLP-1997*, Washington, DC.
18. Christine Thielen and Anne Schiller. 1995. Ein kleines und erweitertes Tagset fürs Deutsche. In *Tagungsberichte des Arbeitstreffens Lexikon und Text*, Lexicographica Series Maior, Tübingen. Niemeier.