Grammar-based Corpus Annotation

Stefanie Dipper
Institut für maschinelle Sprachverarbeitung
Universität Stuttgart

1 Introduction

There is an increasing number of linguists interested in large syntactically annotated corpora (treebanks).[1] Such corpora can serve as a base for statistical applications and, at the same time, may be used in theoretical linguistics as a source for investigations about language use.

The most important treebank nowadays is the Penn Treebank (Marcus et al., 1993; Marcus et al., 1994). Many statistical taggers and parsers have been trained on this treebank, e.g. (Ramshaw and Marcus, 1995; Srinivas, 1997; Alshawi and Carter, 1994). Furthermore, context-free and unification-based grammars have been derived from the Penn Treebank (Charniak, 1996; van Genabith et al., 1999a; van Genabith et al., 1999c; van Genabith et al., 1999b). These parsers, trained or created by means of the treebank, very successfully parse unseen text with respect to correct POS tagging and chunking, and hence can be applied for enlarging the treebank.

However, the situation is different for languages other than English. Ongoing projects are still in the process of building treebanks, e.g. for German (NEGRA corpus (Skut et al., 1997), now continued in the TIGER project; the German treebank in Verbmobil (Stegmann et al., 1998)), for Czech (The Prague Dependency Treebank (Hajič, 1998)); for French (Abeillé et al., 2000). In consequence, the base that parsers could be trained on is still more or less missing. Hence alternative ways of corpus annotation that are not based on statistical parsers may be investigated.

The NEGRA/TIGER corpus consists of German newspaper texts. Currently about 30.000 sentences are annotated with dependency structures. Large parts of the annotation are performed by human annotators supported by the tool `annotate` that integrates a partial parser and a part-of-speech tagger (Brants, 2000b).

As one part of the TIGER project, it is investigated to what extent a symbolic grammar can be applied in annotation. In this approach an existing symbolic LFG grammar is used to parse the corpus. After parsing, disambiguation has to be supported manually. First results of this approach are the topic of this paper.

2 Annotation by Grammar

2.1 Scenario

In the approach presented in this paper, a broad coverage symbolic LFG grammar (Lexical Functional Grammar, (Bresnan, 1982)) is used to parse the corpus. Usually, the grammar output is ambiguous. Disambiguation is done partly manually, partly by a grammar internal ranking mechanism. Finally, the correct reading is saved in PROLOG format.

In our application, a transfer component will convert the PROLOG output into the NEGRA export format (Brants, 1997; Kuhn et al., 2000), or into other representation for-

[1] I would like to thank an anonymous referee for helpful comments on an earlier version of this paper.
The work reported here has been partially funded by the *Deutsche Forschungsgemeinschaft*, project TIGER.

mats such as an XML-based encoding format (Mengel and Lezius, 2000).

In the following sections, LFG parsing and disambiguation is presented, followed by some remarks on grammar coverage and robustness, and annotation accuracy. To illustrate these remarks, parsing results are presented in the final section.

2.2 Representations in LFG

The LFG grammar applied in parsing has been developed using the Xerox Linguistic Environment (XLE). The output of an LFG grammar basically consists of two representations, the constituent structure (c-structure) of the sentence being parsed, and its functional structure (f-structure), containing information about predicate-argument-structure, about attachment sites of adjuncts, and about tense, mood etc. In figure 1, c- and f-structure for *Maria sieht Hans* ('Maria sees Hans') are displayed.

In case of an ambiguous sentence, XLE allows for "packing" the different readings into one complex f-structure representation. All features are represented only once; feature constraints that only hold in one of the readings are marked by variables. The result is an f-structure that is annotated with variables to show where alternatives are possible.

In figure 2, the alternative c-structures for *Maria sieht Hans mit dem Fernglas* ('Maria sees Hans with the telescope') are displayed. The readings differ with respect to the attachment site of the PP *mit dem Fernglas*, either dominated by VP or by NP.

Figure 3 shows the corresponding f-structures, combined in a single f-structure. The PP's f-structure, embedded under the feature ADJUNCT, is displayed only once. In the example, variables a:1 and a:2 indicate the alternative attachments.

The correct reading is selected by a human annotator after parsing. Selection is done either by picking the correct c-structure tree or by clicking on the respective variables in the f-structure.[2]

[2]XLE provides various browsing tools applying to c-structure as well as to f-structure which can be used for manual disambiguation (cf. (King et al., 2000) where these tools are described extensively). This is similar to the syntactic and semantic sentence properties that are displayed by the disambiguation tool "TreeBanker" (Carter, 1997).

2.3 Semi-automatic Disambiguation

In the scenario sketched above, disambiguation is exclusively done by a human annotator. In fact, however, XLE provides a (non-statistical) mechanism for suppressing certain ambiguities automatically. The mechanism consists of a constraint ranking scheme inspired by Optimality Theory (OT) (Frank et al., 1998). Each rule and each lexicon entry can be marked by so-called OT marks. When a sentence is parsed, each analysis is annotated by a multi-set of OT marks. The OT marks keep a record of all rules and lexicon entries being used during the parse to arrive at the analysis in question. The grammar contains a ranked list of all OT marks. When an ambiguous sentence is parsed, the OT mark multi-sets of all readings compete with each other. A multi-set containing a higher ranked OT mark than another multi-set is filtered out.

An example is given in (1). In German, the subject as well as the object can occupy the first position (1a,b). If neither the subject nor the object is overtly case marked, both readings are possible in principle (1c). But in fact, the order subject – object is far more frequent. Hence the second reading can be suppressed by an OT mark. Note that this does not generally exclude objects in first position – as soon as objects are case-marked in an unambiguous way, they are not suppressed any more.

(1) a. der Hans sieht Maria.
the(nom.) H. sees M.
'Hans sees Mary.'

b. den Hans sieht Maria.
the(acc.) H. sees M.
'It is Hans that Mary sees.'

c. Hans sieht Maria.
H. sees M.
'Hans sees Mary.' (preferred)
'It is Hans that Mary sees.'

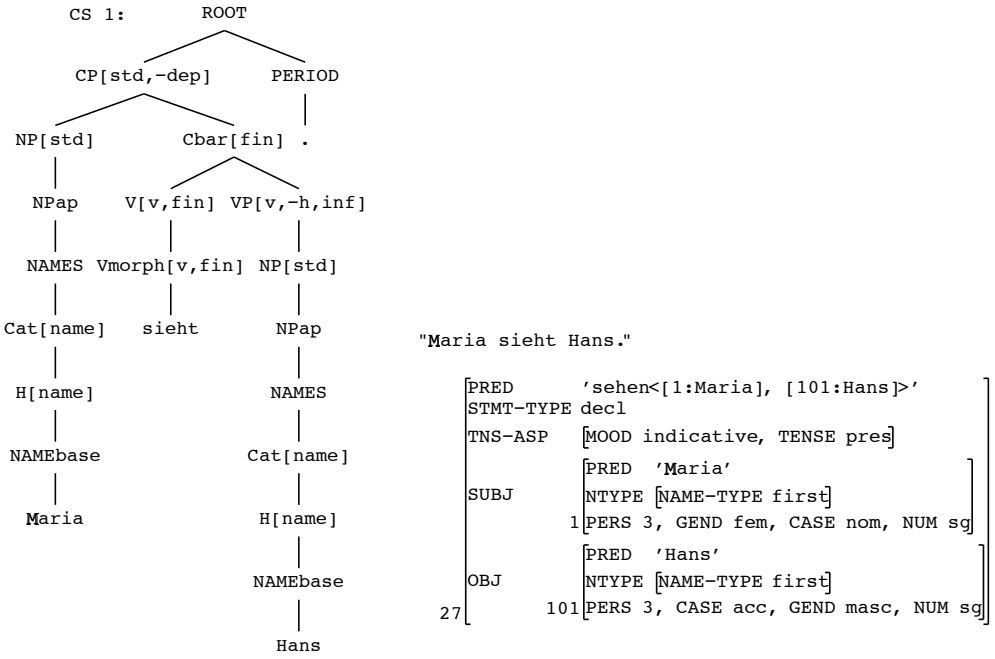

Figure 1: c- and f-structure for *Maria sieht Hans*

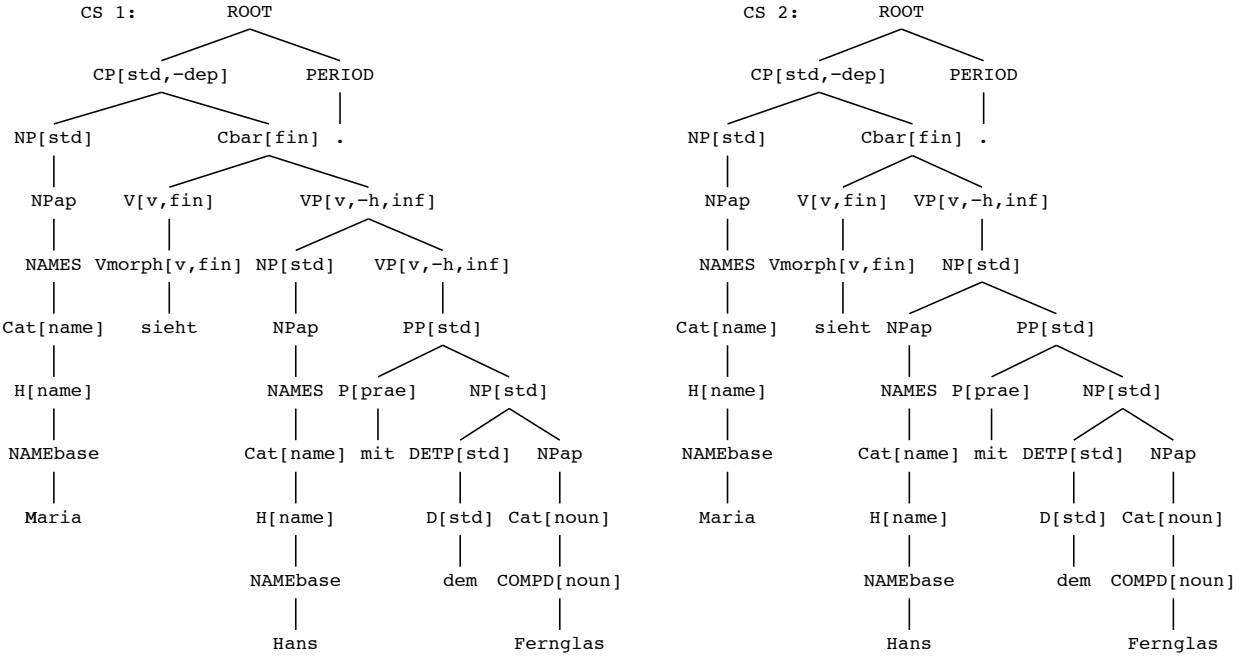

Figure 2: c-structures for *Maria sieht Hans mit dem Fernglas*

In those cases where the correct reading is erroneously suppressed (if, for example, the correct reading does have an object without case-marking in first position), the relevant OT mark can easily be deactivated by the human annotator.

In the ambiguous example presented in 2.2, two readings in fact have been sup-

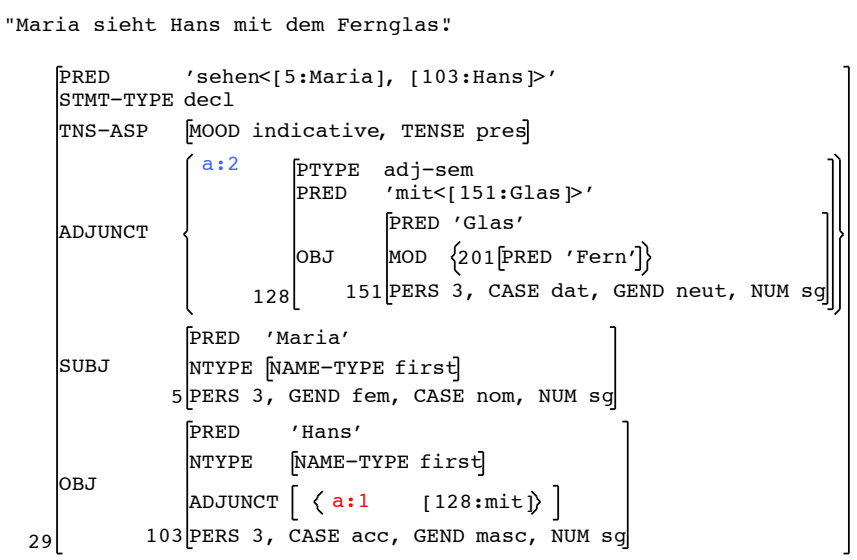

Figure 3: Packed f-structure for *Maria sieht Hans mit dem Fernglas*

pressed by this mechanism. Without OT marks, the f-structure for *Maria sieht Hans mit dem Fernglas* contains two additional analyses with *Hans* as subject, cf. figure 4.

Very often, however, the OT mechanism does not help to determine the correct reading, e.g., when adverb attachment is involved. In these cases, the parser outputs all remaining readings, and disambiguation has to be done manually.[3]

2.4 Coverage and Robustness

For building large annotated corpora, consecutive sentences have to be parsed. Thus, coverage and robustness of the grammar used for annotation is important.

Statistical approaches clearly cope better with free, random text than symbolic approaches. On the one hand, statistical taggers and parsers are able to analyze defective input such as sentences containing typing errors or even ungrammatical sentences. On the other hand, they can provide analyses for rare constructions without getting into ambiguity problems when parsing ordinary sentences – in these cases, rare construction rules are suppressed automatically.

In contrast, parsing by a pure (i.e. non-statistical) LFG grammar yields deep and detailed analyses but at the cost of lower coverage and robustness. Purely symbolic parsing therefore requires text preprocessing.[4] Typing errors and other shortcomings must be corrected, special constructions like newspaper headers have to be marked. For an optimal result, proper nouns such as names of people, organizations, etc. should be listed in a lexicon.

However, even after the best possible text preprocessing and lexicon completion, there will certainly still be constructions that are not parsed by the grammar, e.g., constructions like ellipses and non-constituent coor-

[3] In (Riezler et al., 2000), a statistical model applied to an LFG grammar for German is presented that may be used to support manual disambiguation.

[4] Especially in the domain of speech data processing, much research has been devoted to robust parsers. (Rosé and Lavie, To appear) show that even with a symbolic LFG-style grammar, the parser's flexibility can be increased to cope with word skipping, insertions, etc. However, since this increases the amount of ambiguity, a statistical disambiguation is a prerequisite – which we do not have currently.

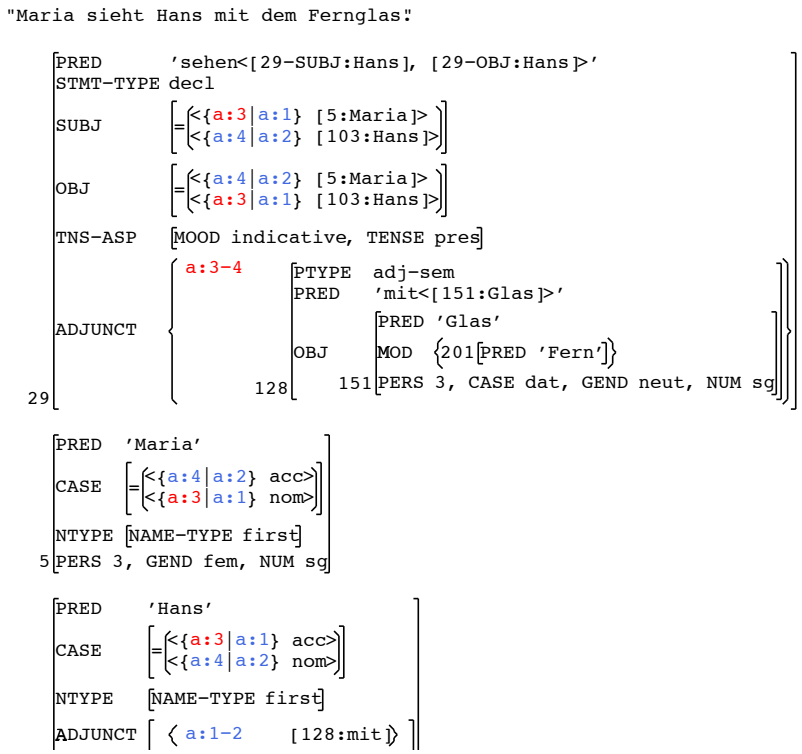

Figure 4: f-structure for *Maria sieht Hans mit dem Fernglas*, no OT filtering

dination. These constructions are problematic since adding the respective rules raises the number of (unwanted) ambiguities for nearly all sentences, and, in addition, it has a negative impact on parsing efficiency.[5]

Clearly, bad coverage and robustness is a problem for grammar-based corpora annotation. XLE provides a special mechanism to improve coverage and robustness. Certain rules or restrictions can be marked by special "STOPPOINT" OT marks. If a sentence is now parsed, these rules or restrictions are ignored. Only if the first parse fails are these rules or restrictions activated and a second parse is started. In this way, rules for rare constructions can be added and restrictions (for instance, on agreement) can be relaxed, without causing serious ambiguity problems for ordinary sentences. Currently we use STOPPOINT OT marks for verb participles used adverbially or in copula constructions. Many of them actually are lexicalized (like *dringend* 'urgent', *verrückt* 'crazy') but nevertheless may be missing from our lexicon. Hence we allow for these participles in general in a second parse, without getting additional readings for each sentence in analytic past tense, i.e. containing an auxiliary plus participle. Further research has to show how to apply this mechanism in an optimized way.

2.5 Accuracy

With respect to accuracy, a grammar-based annotation performs well. We mention three aspects of the approach presented here that support accuracy of annotation.

First, an analysis by an LFG grammar is syntactically consistent, otherwise the parse would have failed. For example, LFG analyses never contain inconsistencies such as the following: missing subject-verb agreement; words tagged as infinitive but functioning as the head of a finite clause; the head of a NP tagged as nominative but the NP function-

[5]Note that non-constituent coordination can be handled in LFG in an elegant way (Maxwell and Manning, 1996). So again, the problem is one of ambiguity management.

ing as an accusative object; etc.

Second, the grammar certainly is not error-free and grammar internal errors may carry over to the analyses but these errors are systematic. If, for example, a proper noun like *Kohl* is not listed as a name in the grammar's lexicon, all analyses of sentences about the person Helmut Kohl falsely contain the reading of *Kohl* as a common noun ('cabbage'). But once the error is detected in one analysis or in the grammar itself, it is often possible to automatically track down all other instances of the same error occurring previously in the annotation. Note, however, that such errors may be difficult to detect.

Third, manual disambiguation of LFG analyses usually does not impair accuracy of the annotated corpus, since in many cases, disambiguation is guided by prominent properties. When picking the correct reading, the human annotator can make use of clear, prominent properties of the analyses, namely constituent structure and predicate-argument-structure.

2.6 Some Performance Data

To illustrate the findings of the preceding sections, we present some figures indicating the grammar's performance. Note, however, that the grammar has not been tuned or trained with respect to the corpus.

In a first experiment, 2000 sentences from the TIGER corpus (German newspaper texts) were parsed. In a first pass, the text was parsed without any preprocessing (except for splitting the text into sentences). In a second pass, header markers were added and quotes were removed (since the grammar currently does not accept quoted text; the quotes can be easily recovered after parsing).[6] These text modifications were done automatically. The grammar performance improved considerably, cf. rows 1 and 2 in figure 5.

[6]Quotes are problematic for several reasons: They are ambiguous and either mark direct speech or quote material in the running text. Quotes do not always correspond to constituents boundaries and matching pairs of quotes may be distributed over distinct sentences.

Then the grammar was partly rewritten with two major modifications: first, the grammar was tuned for efficiency (without affecting coverage); second, PP and adverb attachment were allowed in a more general way than in the previous grammar version. This increased coverage as well as ambiguity, as can be seen in the third row, reporting about 6000 sentences (preprocessed in the same way as in the second pass).

The first column shows the number of sentences in the test corpus, the second column shows the number of sentences that got a parse (without checking for correctness). As can be seen, in the first pass only 28% of the sentences were parsed as opposed to 40% after some text preprocessing. After some general grammar modifications, 47% were parsed.[7]

The third column contains the number of analyses or readings per parsed sentence. Only readings that were not filtered out by the XLE internal disambiguation mechanism are taken into account (hence "optimals"). Both average as well as median are given. As can be seen from figure 5, in the third pass the average number of readings increased massively. But nevertheless the median is 2, so most of the sentences are still easy to disambiguate manually. Note that in this experiment, it was not checked whether the correct reading was among the analyses.

The forth column reports about the number of analyses that were suppressed by XLE disambiguation (hence "suboptimals").

Finally, average parsing time and number of tokens per sentence are given.

In a second experiment, 300 sentences were parsed and the analyses were evaluated.

[7]We are only aware of one sentence-based evaluation involving a grammar with comparably deep analyses: without tuning, the XTAG grammar parsed 39.09% of 6364 sentences ($\leq$ 15 words long) from the Wall Street Journal with an average of 7.53 analyses per sentence (Doran et al., 1994). Other evaluations usually measure performance below sentence level, such as chunking or (super)-tagging (Srinivas, 1997; Ramshaw and Marcus, 1995; Brants, 1999), and hence are not comparable with our grammar that does not yield partial analyses (yet).

	#sentences	parsed	optimals		suboptimals		time(sec)		#tokens
			Ø	Med	Ø	Med	Ø	Med	Ø
1.	2000	553 (= 28%)	7	2	1689	7	17	1.8	15.5
2.	2000	809 (= 40%)	6	2	3480	10	17	1.8	15.3
3.	6000	2833 (= 47%)	28	2	34331	18	14	1.9	16.0

Figure 5: LFG parsing results for German newspaper sentences

160 sentences were parsed by the grammar; among these, 120 parses contained the correct reading (the correct reading had to be part of the "optimal" analyses), cf. figure 6.

We also did some preliminary evaluation of the errors.

- 10% of the sentences were not parsed because of gaps in the morphological analyzer.[8]
- 4% of the sentences failed because of storage overflow or timeouts (with limits set to 100 MB storage and 100 seconds parsing time).
- More than 30% of the sentences failed because gaps in the lexicon, which are mostly due to missing subcategorization frames.[9]

We decided not to manually disambiguate sentences that get more than 20 analyses. This is the case for 5.8% of the sentences. With this restriction, a trained human annotator disambiguates about one sentence per minute on average.[10]

To sum up the findings of this section: in the short-term, these data suggest the necessity of the following: further text preprocessing such as correction of typing errors; completion of the grammar's lexicon by extracting unknown words from the corpus.

However, in the long-term, we will have to apply statistical disambiguation. This will allow us to include robustness mechanisms.

In the meantime, the remainder of the sentences that have not been correctly parsed by our grammar are annotated by means of the tool `annotate`.

3 Conclusion and Outlook

We have presented first results in syntactic annotation of a large German corpus by a symbolic LFG grammar. On average, the grammar parses 47% of the sentences. Among these, 75% contain the correct reading. Disambiguation is done partly by the XLE internal ranking mechanism. Remaining ambiguities (median: 2) are solved by a human annotator. This takes about one minute per sentence with an average length of 16.0 tokens.

By means of a transfer component, LFG representations can be converted into canon-

[8] We use a guesser mechanism for capitalized words that also handles genitive and plural inflection. All morphological failures are due to non-capitalized unknown words or else capitalized words containing strings other than characters or numbers.

[9] The base lexicon is mainly extracted automatically from corpora (Eckle-Kohler, 1999) and mostly consists of subcategorization frames (in the TSNLP format). There are 14.000 verb lemmata with 28.500 frames (115 different frames); 1100 adjective lemmata with 1650 frames (17 different); 780 noun lemmata with 970 frames (3 different). The TSNLP frames are converted automatically into an LFG format (Bröker and Dipper, 1999).

[10] This result is very similar to that reported in (Brants, 2000a), where a trained annotator needs on average 50 seconds to annotate a sentence with an average length of 17.5 tokens.

#sentences	parsed	correct reading among optimals
300	160 (= 53%)	120 (= 40%)

Figure 6: Evaluation of 300 sentences

ical treebank formats.

Coverage and robustness are weak points in grammar-based annotation. The performance data presented in 2.6 point to a need to further exploit text preprocessing and to complete the grammar's lexicon. In the longer term, however, statistical disambiguation and robustness mechanisms such as relaxation of certain restrictions have to be investigated.

References

Anne Abeillé, Lionel Clément, and Alexandra Kinyon. 2000. Building a treebank for French. In *Proceedings of the LREC 2000*, Athens, Greece.

Hiyan Alshawi and David Carter. 1994. Training and scaling preference functions for disambiguation. *Computational Linguistics*, 20(4).

Thorsten Brants. 1997. The NeGra export format for annotated corpora (version 3). Technical report, NEGRA Project, Universität des Saarlandes.

Thorsten Brants. 1999. Cascaded Markov Models. In *Proceedings of 9th Conference of the EACL 1999*, Bergen, Norway.

Thorsten Brants. 2000a. Inter-annotator agreement for a German newspaper corpus. In *Proceedings of the LREC 2000*, Athens, Greece.

Thorsten Brants. 2000b. Interactive corpus annotation. In *Proceedings of the LREC 2000, Athens, Greece.*

Joan Bresnan, editor. 1982. *The Mental Representation of Grammatical Relations.* MIT Press.

Norbert Bröker and Stefanie Dipper. 1999. Zur Konstruktion von Lexika für die maschinelle syntaktische Analyse. In J. Gippert and P. Olivier, editors, *Multilinguale Corpora - Codierung, Strukturierung, Analyse. 11. Jahrestagung der Gesellschaft fuer Linguistische DatenVerarbeitung.* Enigma corporation, Prag.

David Carter. 1997. The TreeBanker: a tool for supervised training of parsed corpora. In *Proceedings of the ACL Workshop on Computational Environments for Grammar Development and Linguistic Engineering*, Madrid, Spain.

Eugene Charniak. 1996. Tree-bank grammars. In *AAAI-96. Proceedings of the Thirteenth National Conference on Artificial Intelligence.* MIT Press.

Christine Doran, Dania Egedi, Beth Ann Hockey, B. Srinivas, and Martin Zaidel. 1994. XTAG system – a wide coverage grammar for English. In *Proceedings of International Conference on Computational Linguistics (COLING 94)*, Kyoto, Japan.

Judith Eckle-Kohler. 1999. *Linguistisches Wissen zur automatischen Lexikon-Akquisition aus deutschen Textkorpora.* Logos, Berlin.

Anette Frank, Tracy Holloway King, Jonas Kuhn, and John Maxwell. 1998. Optimality theory style constraint ranking in large-scale LFG grammars. In *Proceedings of the LFG98 Conference*, Brisbane, Australia. CSLI Online Publications, http://www-csli.stanford.edu/publications.

Jan Hajič. 1998. Building a syntactically annotated corpus: The Prague Dependency Treebank. In Eva Hajičova, editor, *Issues of Valency and Meaning. Studies in Honour of Jarmila Panevová.* Charles University Press, Prag.

Tracy Holloway King, Stefanie Dipper, Anette Frank, Jonas Kuhn, and John Maxwell. 2000. Ambiguity management in grammar writing. In *Proceedings of the ESSLLI 2000 Workshop on Linguistic Theory and Grammar Implementation*, Birmingham, Great Britain.

Jonas Kuhn, Heike Zinsmeister, and Martin Emele. 2000. From LFG structures to TIGER treebank annotations. Presented at the Workshop on Syntactic Annotation of Electronic Corpora, University of Tübingen.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2).

Mitchell P. Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the Human Language Technology Workshop*. Morgan Kaufmann.

John T. Maxwell and Christopher D. Manning. 1996. A theory of non-constituent coordination based on finite-state rules. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG96 Conference*, Grenoble, France. CSLI Online Publications, http://www-csli.stanford.edu/publications.

Andreas Mengel and Wolfgang Lezius. 2000. An XML-based encoding format for syntactically annotated corpora. In *Proceedings of the LREC 2000*, Athens, Greece.

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text chunking using transformation-based learning. In *Proceedings of the Third ACL Workshop on Very Large Corpora*, Dublin, Ireland.

Stefan Riezler, Detlef Prescher, Jonas Kuhn, and Mark Johnson. 2000. Lexicalized stochastic modeling of constraint-based grammars using log-linear measures and EM training. In *Proceedings of the ACL 2000*, Hong Kong, China.

Carolyn Penstein Rosé and Alon Lavie. To appear. Balancing robustness and efficiency in unification-augmented context-free parsers for large practical applications. In van Noord and Junqua, editors, *Robustness in Language and Speech Technology*. Kluwer Academic Press.

Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proceedings of ANLP-97*, Washington.

B. Srinivas. 1997. Performance evaluation of supertagging for partial parsing. In *Proceedings of Fifth International Workshop on Parsing Technology*, Boston, USA.

Rosmary Stegmann, Heike Schulz, and Erhard Hinrichs. 1998. Stylebook for the German treebank in Verbmobil. Verbmobil, Universität Tübingen.

Joseph van Genabith, Louisa Sadler, and Andy Way. 1999a. Data-driven compilation of LFG semantic forms. In *Proceedings of the EACL 1999 Workshop on Linguistically Interpreted Corpora (LINC-99)*, Bergen, Norway.

Joseph van Genabith, Louisa Sadler, and Andy Way. 1999b. Semi-automatic generation of f-structures from treebanks. In *Proceedings of the LFG99 Conference*, Manchester, Great Britain. CSLI Online Publications, http://www-csli.stanford.edu/publications.

Joseph van Genabith, Louisa Sadler, and Andy Way. 1999c. Structure preserving CF-PSG compaction, LFG and treebanks. In *Proceedings of the ATALA Treebank Workshop*, Paris, France.