Treebank Conversion - Establishing a testsuite for a broad-coverage LFG from the TIGER treebank

Martin Forst
Institute for Natural Language Processing
University of Stuttgart
forst@ims.uni-stuttgart.de

Abstract

This paper reports on the conversion of the TIGER treebank, a syntactically interpreted corpus of German newspaper texts, into a testsuite for a broad-coverage Lexical-Functional Grammar (LFG) for German. This is done by first converting the TIGER XML representation into a relational Prolog-like representation and then applying term-rewriting rules as they are used in certain MT transfer components to that representation. The output consists of (partly underspecified) f-structures, which can then be mapped against the grammar's output for evaluation purposes.

1 Introduction

In grammar development, one is often confronted with the dilemma that coverage, efficiency, linguistic adequacy and the reduction of ambiguities are conflicting goals. At present, a linguist involved in the development of a broad-coverage grammar such as the German ParGram[1] LFG is additionally limited by the absence of large annotated testsuites, when wanting to evaluate the grammar thoroughly or when trying to find out in how far grammar modifications affect one of the above features due to inexpected interactions. Of course, it is possible to run the grammar on large corpora and to state afterwards what percentage of the sentences in a given corpus got at least one analysis (E.g., the German ParGram LFG parses 54.8% of the NEGRA corpus.), how many sentences timed out or failed because of storage overflow (1.2%) and what percentage was not parsed (44%). It is virtually impossible, however, to determine whether the intended analysis is among the often extremely numerous analyses proposed by the grammar, nor can we really test what the exact consequences of grammar modifications are. In order to evaluate systematically whether and how one of the features mentioned above can be improved without decreasing its performance with respect to another one, it is necessary to make use of large testsuites that be annotated according to the formalism used. Since the manual annotation of such testsuites would be extremely time-consuming, it seems reasonable to use an existing treebank, the TIGER corpus in our case, and to convert it to the format we need, which is the one of LFG f-structures.

Similar efforts of f-structure annotation of treebanks have been reported on in Van Genabith et al. (1999), Sadler et al. (2000), Frank (2000), Frank et al. (2001), Van Genabith et al. (2001), and Cahill et al. (2002). As in all that work the source format (AP corpus, Susanne corpus, Penn treebank) differs considerably from the TIGER format in that it encodes mainly phrase-structural information, our approach is quite different, however, from the ones mentioned. Dependency information being expressed explicitly in the edge labels,

[1] ParGram is a joint research project on the parallel development of large-scale grammars for several languages involving PARC, Fuji Xerox, the University of Bergen, the University of Manchester and the University of Stuttgart.

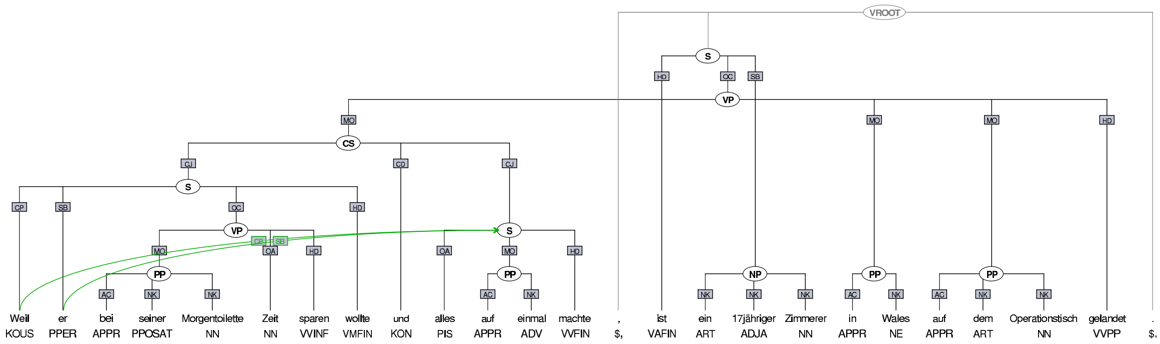

Figure 1: TIGER tree representation of corpus sentence no. 23747

```
  <t id="s23747_21" word="auf" pos="APPR" morph="--" />
  <t id="s23747_22" word="dem" pos="ART" morph="--" />
  <t id="s23747_23" word="Operationstisch" pos="NN" morph="--" />
  <t id="s23747_24" word="gelandet" pos="VVPP" morph="--" />
  <t id="s23747_25" word="." pos="$." morph="--" />
</terminals>
<nonterminals>
  <nt id="s23747_500" cat="PP">
    <edge label="AC" idref="s23747_3" />
    <edge label="NK" idref="s23747_4" />
    <edge label="NK" idref="s23747_5" />
  </nt>
  <nt id="s23747_501" cat="PP">
    <edge label="AC" idref="s23747_11" />
    <edge label="NK" idref="s23747_12" />
  </nt>
  <nt id="s23747_502" cat="NP">
    <edge label="NK" idref="s23747_16" />
```

Figure 2: excerpt of the TIGER XML representation of corpus sentence no. 23747

we do not need to f-annotate the treebank (or a context-free grammar extracted from it), but we can directly convert the hybrid TIGER representation into f-structures.

Another related work is Frank (2001), which consists of the extraction of an LTAG from the NEGRA corpus. Here, the source format is comparable to ours, the TIGER format being an extension of the NEGRA format, and the main differences with respect to our work are due to the different target format. For the conversion of the corpus to a collection of f-structures, constituency information is almost irrelevant, whereas it is crucial for the extraction of an LTAG.

Finally, our conversion is, of course, in many ways similar to the inverse conversion from LFG analyses to TIGER trees (Zinsmeister et al., 2002). E.g. we use the same term-rewriting system. However, since the relation between TIGER trees and f-structures is far from being a one-to-one mapping, it raises new questions. Moreover, we aim at converting the entire TIGER treebank into an 'f-structure bank' without any human intervention, an objective that is quite different from grammar-based treebank annotation.

Our presentation of the conversion process is organized as follows. Section 2 describes the first step in the process, which is the transformation of TIGER trees into feature structures. In section 3 we present the transfer system we use for the transformation of TIGER-like feature structures into f-structures, as well as a number of transfer phenomena and their treatment in that formalism. Finally, section 4 gives an outlook on the possibilities for future work offered by the resulting f-structure bank.

2 The TIGER treebank and the relational TIGER representation

The TIGER treebank is a corpus of currently 36,000 syntactically annotated German newspaper sentences. The annotation consists of generalized graphs, i.e. trees which may contain crossing and secondary edges. Edges are labeled, so that a TIGER tree encodes both phrase-structural information and information on dependency relations.[2]

The TIGER trees are represented in a specific XML format, the so-called TIGER XML.[3] Figure 2 illustrates what the TIGER XML representation of an annotated sentence like the one in figure 1 looks like.

In order to be able to use an existing term-rewriting system for the conversion of the TIGER trees into f-structures, we first need to have the TIGER corpus available in a relational Prolog-like representation. Instead of being a generalized graph, a TIGER tree then has to take the form of a feature structure.

This conversion raises a first problem: In a TIGER tree, there can be several identically labelled edges that go from one single node to various of its daughter nodes. In feature structures, on the contrary, a given attribute can only have one unique value. It is thus not possible to convert a TIGER tree into a feature structure by a one-to-one mapping. Fortunately, there is quite a straightforward solution to this problem: As attributes in a feature structure can be set-valued, all identically labeled daughter nodes of a given node can be put into a set. The resulting representation differs somewhat from the initial tree, but it contains basically the same information.

Another problem that we need to deal with when converting TIGER trees into feature structures is the fact that, generally, the latter do not encode any information about precedence relations. This kind of information can be crucial, however, for subsequent steps in the conversion from one format to the other. Genitive attributes, for example, are labeled AG in the TIGER treebank, whether they are on the left or on the right of their head noun. The broad-coverage LFG for German, on the contrary, analyzes them in two different ways, either as a SPEC POSS, when they are in prenominal position, or as a member of the set-valued feature ADJUNCT, when they appear postnominally. This means that a minimum of information about precedence needs to be encoded in the relational TIGER representation.

This can be done with the help of a special predicate that allows us to state that a certain node A precedes another node B. By means of this predicate, called 'scopes' in the system we use, we express precedence relations between daughters of the same mother node. This kind of information is sufficient to disambiguate all TIGER-LFG mismatches which can be disambiguated on the basis of precedence information.

The first step of the conversion of TIGER trees into f-structures thus consists of transforming the trees into feature structures. As this task does not require any major structural changes, it can be carried out quite comfortably by means of an XSL style sheet[4]. Figure 3 shows an excerpt of the relational Prolog-like representation of the corpus sentence displayed in 1 that results from the XSL conversion of the TIGER XML representation illustrated in figure 2. Figure 4 displays the corresponding feature structure.

```
cf(1, eq(attr(var(504), 'AC' ), var(1001504))),
cf(1, in_set(var(21), var(1001504))),
cf(1, eq(attr(var(21), 'TI-FORM' ), 'auf')),
cf(1, eq(attr(var(21), 'TI-POS' ), 'APPR')),
cf(1, scopes(var(22), var(23))),
cf(1, eq(attr(var(504), 'NK' ), var(1012504))),
cf(1, in_set(var(22), var(1012504))),
cf(1, eq(attr(var(22), 'TI-FORM' ), 'dem')),
cf(1, eq(attr(var(22), 'TI-POS' ), 'ART')),
cf(1, eq(attr(var(504), 'NK' ), var(1012504))),
cf(1, in_set(var(23), var(1012504))),
cf(1, eq(attr(var(23), 'TI-FORM' ), 'operationstisch')),
cf(1, eq(attr(var(23), 'TI-POS' ), 'NN')),
cf(1, eq(attr(var(509), 'HD' ), var(24))),
cf(1, eq(attr(var(24), 'TI-FORM' ), 'gelandet')),
cf(1, eq(attr(var(24), 'TI-POS' ), 'VVPP')),
```

Figure 3: excerpt of the relational Prolog-like representation of corpus sentence no. 23474

3 Treebank conversion by (MT) transfer rules

Although the f-structures we obtain from our broad-coverage LFG and the TIGER treebank representations coincide in core aspects, e.g. the encoding of grammatical functions, there are mismatches in analysis details that are comparable to translation mismatches in natural language trans-

[2]For more details on the annotation scheme see Skut et al. (1997), Brants & Hansen (2002), and Brants et al. (2002).

[3]See Mengel & Lezius (2000).

[4]Thanks to Hannes Biesinger for a first version of the XSL style sheet and to Stefanie Dipper for her contribution to its final adaption.

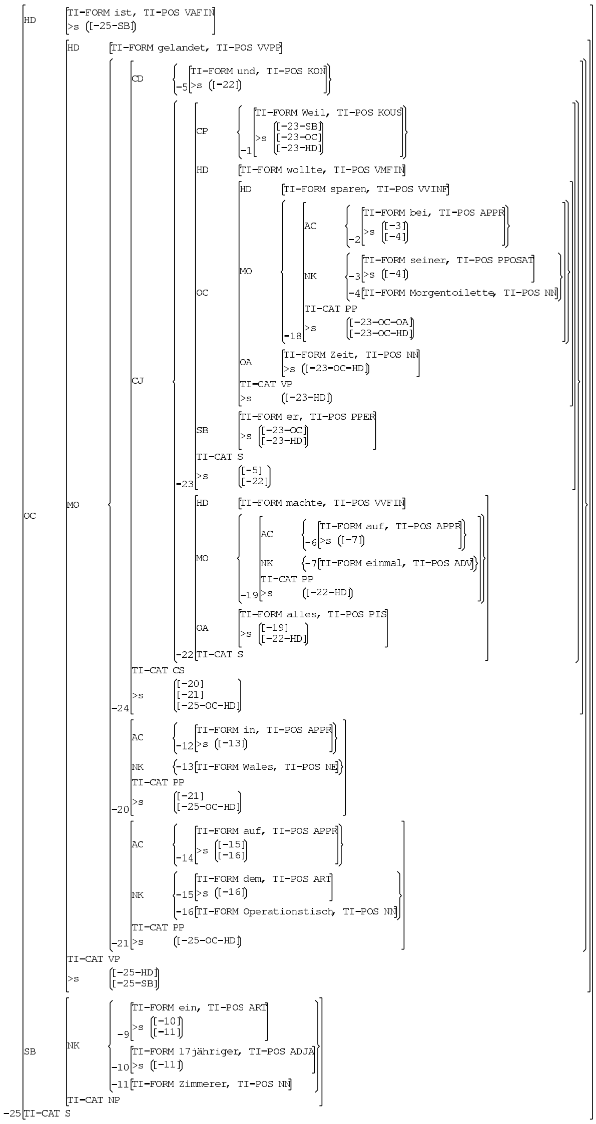

Figure 4: representation of corpus sentence no. 23747 as a TIGER-annotated feature structure

lation. One such phenomenon is the flat analysis of auxiliary constructions generally adopted in LFG versus the intricate analysis that has been chosen for the TIGER treebank. This kind of mismatches motivates the use of transfer technology originally developed for machine translation.

3.1 The transfer system

The transfer system we use is a term rewriting system based on Prolog. It has originally been developed by Martin Kay and is now part of the XLE grammar development platform. The rules it processes are ordered, which means that the output of a given rule r_i is input to rule r_{i+1}. Each rule replaces a certain set of predicates (those on the left-hand side of the rule) by another set of predicates (those on its right-hand side). Input and output predicates are separated by a rewriting symbol, the operator '==>'. The most basic rules simply rewrite the name of the predicate and pass on the values of the arguments unchanged. For example, the rule given in (1) maps the TIGER edge label OA (accusative object) to the LFG function OBJ.

(1) `oa(X,Y) ==> obj(X,Y).`

In addition, it is possible to specify predicates on the left-hand side that have to be matched, but are not replaced (marked with a '+'), as well as predicates that must not be matched for the rule to be applied (marked with a '-'). These mechanisms are used in the following rule, which takes a partial feature structure whose attribute TI-POS has the value PIAT (for 'attributive indefinite pronoun') out of the set that is the value of the feature NK (for 'noun kernel') and attributes it to a new feature SPEC QUANT, if that feature does not yet exist.

(2)
```
+nk(A,SET), in_set(B,SET),
+ti_pos(B,'PIAT'), -spec(A,_)
==>
spec(A,SPEC), quant(SPEC,B).
```

It is also possible to delete features by writing a zero on the right-hand side of a rule, which stands for the empty set. In this case, all predicates on the left-hand side of the rule are deleted from the set of terms without replacement.

(3) `ti_form(_,_) ==> 0.`

Finally, the possibility of defining rules as optional needs to be mentioned as well. Optional rules are characterized by the use of the operator '?=>' instead of '==>'. They allow us to transfer a given input feature structure to two alternative output structures - or more, if several optional rules are applied. We can thus handle cases where we cannot clearly decide on the sole basis of the input what the output must look like. The TIGER label MO (for 'modifier'), for example, is such a phenomenon, because the context is not always sufficient to determine whether it is to be transferred to an element of the set-valued feature ADJUNCT, to an OBL-DIR (directional oblique), an OBL-LOC (locative oblique) or still another grammatical function. The following rule optionnally transfers a MO-PP with an AC (the edge label used for pre- and postpositions in TIGER) that has the form 'nach' into an OBL-DIR.

(4)
```
+ti_cat(S,'S'),
+mo(S,MO), in_set(PP,MO),
+ti_cat(PP,'PP'), +ac(PP,APPR),
+ti_form(APPR,'nach') ?=>
obl_dir(S,PP).
```

For reasons of userfriendliness and maintainability, the XLE transfer system also allows the use of templates and macros. They are short-hand notations for sets of rules and predicates respectively. As they are not directly relevant for our presentation, however, we do not present them here in more detail.

3.2 Transfer phenomena

Unlike transfer in machine translation, the transfer from TIGER trees to LFG f-structures does not aim at changing the surface string. The task is rather to map a limited set of grammatical features into another limited set of grammatical features. Nevertheless, the format conversion is far more complex than a simple mapping from one feature set to another, because (i) there is no one-to-one correspondence between features and (ii) the different analyses chosen for certain grammatical phenomena can have relatively heavy repercussions on the structure of the representations involved.

3.2.1 Ambiguous edge labels

In section 2, we mentioned the case of the TIGER edge label AG, which depending on the position of the AG constituent with respect to its head noun corresponds to either a SPEC POSS feature or an ADJUNCT feature in a German LFG analysis. Still, this kind of ambiguity can easily be resolved on the basis of precedence information, so that we simply need two obligatory rules for the transfer of AGs, one for prenominal ones and a 'default rule' for postnominal ones. As rules are ordered, the 'default rule' is only applied, if the more specific rule was not.

(5) a.
```
+ti_cat(NP,'NP'),
+nk(NP,NKSET),
+in_set(HEAD,NKSET),
+ti_pos(HEAD,'NN'),
+scopes(AG,HEAD), ag(NP,AG)
==>
spec(NP,SPEC), poss(SPEC,AG).
```
b.
```
ag(NP,AG) ==>
adjunct(NP,ADJUNCT),
in_set(AG,ADJUNCT).
```

A somewhat more complex case is the transfer of the predicate MO. It can correspond to the predicates ADJUNCT, OBL-DIR and OBL-LOC. This is due to the fact that PPs such as *auf dem Operationstisch* in corpus sentence no. 23747 are analysed as subcategorized arguments in the German LFG (cf. figure 7), and not as ADJUNCTs or MOs respectively, as it is the case in the TIGER treebank (cf. figure 1).

We deal with this case by first using the optional rule in (6a), which similarly to the one in (4) converts a MO into an OBL-LOC, and then applying the default rule given in (6b), which transfers all MOs to ADJUNCTs. In order not to obtain too many output f-structures, we try to limit the application of the optional rules to as few contexts as is reasonably possible, while keeping them general enough to cover all cases that we need for a justifiable comparison of the output of the German LFG and the TIGER annotation.

(6) a.
```
+ti_cat(S,'S'),
+mo(S,MO), in_set(PP,MO),
+ti_cat(PP,'PP'),
+ac(PP,APPR),
+ti_form(APPR,'auf') ?=>
obl_loc(S,PP).
```
b.
```
mo(S,MO) ==> adjunct(S,MO).
```

3.2.2 Structural changes

Given that TIGER trees on the one hand encode information about both phrase structure and dependency relations and that f-structures on the other hand only represent the latter type of information, it is not surprising that the analysis of a few grammatical phenomena differs considerably between the TIGER corpus and the German LFG analyses. This is the case of analytic tenses, for example, which generally get a flat analysis in LFG, the auxiliary and the main verb being treated as f-structure co-heads, whereas in TIGER the VP containing the non-finite main verb form is analysed as a clausal object (OC) of the auxiliary. Figure 5 shows a TIGER tree containing an analytic verb form and figure 6, the corresponding f-structure.

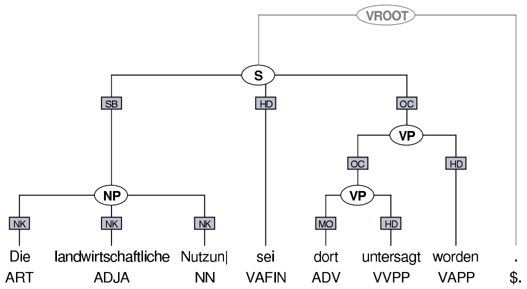

Figure 5: TIGER tree representation of corpus sentence no. 2456

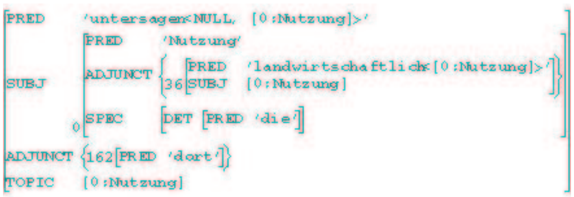

Figure 6: f-structure associated to corpus sentence no. 2456 by the German broad-coverage LFG

This kind of structural change is known as head-switching in the field of machine translation. As studies about the treatment of head-switching phenomena have shown, they can be dealt with without major difficulty by a term-rewriting system.

Another phenomenon for which structural changes have to be made on the way from a TIGER tree representation to an f-structure is the attachment of ADJUNCTs or MOs respectively that modify a verb which is embedded under a modal verb. For the German LFG it has been decided that this kind of ADJUNCT is a feature of the outer f-structure and not of the partial XCOMP f-structure within it. This analysis helps to avoid a systematic ambiguity, which would arise if the attachment to both the outer and the embedded verb were allowed, but which generally is not of importance from a semantic point of view. In TIGER, according to its annotation principles, a MO is always attached where it belongs semantically, which means that in most cases it is embedded in the TIGER counterpart of XCOMP, namely OC (for 'clausal object'); in unclear cases, it is attached as low as possible. This difference in adjunct attachment can be observed in in figures 1 and 7 with respect to the PP *bei seiner Morgentoilette*.

Again, this and all other kinds of structural changes needed for the conversion of TIGER trees into f-structures can be handled quite comfortably with a term-rewriting system.

4 Outlook to the use of the resulting 'f-structure bank' in grammar development

Preliminary experiments in view of the conversion of the whole TIGER treebank are very encouraging. A thorough evaluation of the transfer quality, in which we will manually check the transfer result for 200 randomly selected sentences from the TIGER corpus that can be parsed by the German ParGram LFG, is still to be carried out. Open questions are mainly the exact treatment of secondary edges, which for the moment are converted just like normal edges, as well as the conversion of parenthetical and elliptical constructions. This kind of constructions being rather marginal, we hope to have converted a large part of the TIGER treebank into an f-structure bank by the time of the workshop.

Having this large German f-structure bank available will make it possible to evaluate the German ParGram LFG in a much more informative way than this can be done at the moment. No longer will we be restricted to observing what percentage of a given corpus can be parsed by the grammar, what proportion fails due to timeout or storage overflow and what percentage is rejected, but we will have the means to determine whether the desired analysis is among the analyses proposed by the grammar and whether it is among

Figure 7: f-structure associated to corpus sentence no. 23747 by the German broad-coverage LFG

the preferred solutions.[5] For the first time, we will have a detailed picture of how good (or bad) the analyses are which the German ParGram LFG proposes, and it will no longer be a problem to control the repercussions of grammar modifications intended to increase coverage, for example, on parsing quality, efficiency etc.

Last but not least, the resulting f-structure bank will be indispensable for the supervised training of statistical disambiguation modules. Actually, we plan to employ exponential models on ambiguous LFG parses along the lines of Crouch et al. (2002).

In conclusion, treebanking along with treebank conversion opens up a whole series of new possibilities for the development of fine-grained syntactic analyzers. Most importantly, it will permit the use of probabilistic disambiguation based on supervised training and facilitate detailed grammar evaluation. And as it should be relatively easy to adapt our treebank conversion approach to the representations resulting from related syntactic analyses (e.g. HPSG), it might even open the way for the comparison of different parsers for German.

[5]XLE provides a non-statistical OT-inspired disambiguation method, which prefers or disprefers certain solutions with respect to other ones. See Frank et al. (1998).

References

Bouma, G., G. van Noord, R. Malouf (2000). Alpino: Wide-coverage computational analysis of Dutch. In *Proceedings of Computational Linguistics in the Netherlands*, Amsterdam, Netherlands.

Butt, M., T. H. King, M.-E. Niño, and F. Segond (1999). *A Grammar Writer's Cookbook*. CSLI Publications, Stanford, CA.

Brants, S., S. Dipper, S. Hansen, W. Lezius, and G. Smith (2002). The TIGER Treebank. In Hinrichs, E. & K. Simov (eds.), *Proceedings of the First Workshop on Treebanks and Linguistic Theories (TLT 2002)*, Sozopol, Bulgaria.

Brants S. & S. Hansen (2002). Developments in the TIGER annotation scheme and their realization in the corpus. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC '02)*. Las Palmas, Spain.

Cahill, A., M. McCarthy, J. van Genabith, A. Way (2002). Evaluating Automatic F-Structure Annotation for the Penn-II Treebank. In Hinrichs, E. & K. Simov (eds.), *Proceedings of the First Workshop on Treebanks and Linguistic Theories (TLT 2002)*, Sozopol, Bulgaria.

Cahill, A., M. McCarthy, J. van Genabith, A. Way (2002). Automatic Annotation of the Penn-Treebank with LFG F-Structure Information In Lenci, A., S. Montemagni, V. Pirelli (eds.), *Proceedings of the LREC Workshop on Linguistic Knowledge Acquisition and Representation - Bootstrapping Annotated Language Data*, ELRA - European Language Resources Association, Paris, France.

Carroll, J., G. Minnen, Ted Briscoe (1999). Corpus annotation for parser evaluation. In *Proceedings of the EACL workshop on Linguistically Interpreted Corpora (LINC)*, Bergen, Norway.

Crouch, R., R. M. Kaplan, T. H. King, and S. Riezler (2002). A comparison of evaluation metrics for a broad-coverage stochastic parser. In *Proceedings of the "Beyond PARSEVAL" Workshop at the 3rd International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Spain.

Crouch, R., M. Johnson, R. M. Kaplan, T. H. King, J. T. Maxwell III, S. Riezler (2002). Parsing the Wall Street Journal using a Lexical-Functional Grammar and Discriminative Estimation Techniques. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 2002*, Philadelphia.

Dipper, S. (2000). Grammar-based corpus annotation. In *Proceedings of the Workshop on Linguistically Interpreted Corpora*, Luxembourg.

Dipper, S. (to appear). *Implementing and Documenting Large-scale Grammars - German LFG*. Ph.D. thesis, University of Stuttgart, Germany.

Emele, M., M. Dorna, A. Lüdeling, H. Zinsmeister, C. Rohrer (2000). Semantic-based Transfer. In: Wahlster, W. (ed.): *Verbmobil: Foundations of Speech-to-Speech Translation*, Springer Verlag.

Frank, A., T. H. King, J. Kuhn, J. Maxwell (1998). Optimality theory style constraint ranking in large-scale lfg grammars. In Butt, M. & T. H. King (eds.), *Proceedings of the LFG98 Conference*, University of Queensland, Brisbane, CSLI Online Publications, Stanford, CA. `http://www-csli.stanford.edu/publications/`.

Frank, A. (2000). Automatic F-structure Annotation of Treebank Trees. In Butt, M. & T. H. King, (eds.), *Proceedings of the LFG00 Conference*, University of California at Berkeley, CSLI Online Publications, Stanford, CA. `http://www-csli.stanford.edu/publications/`.

Frank, A. (2001). Treebank Conversion - Converting the NEGRA treebank to an LTAG grammar. In *Proceedings of the Workshop on Multi-layer Corpus-based Analysis*, Iasi, Romania.

Frank, A., L. Sadler, J. van Genabith, A. Way (2001). From Treebank Resources to LFG F-Structures. Automatic F-Structure Annotation of Treebank Trees and CFGs extracted from Treebanks. To appear in: Abeillé, A. (ed.): *Treebanks. Building and using syntactically annotated corpora*, Kluwer Academic Publishers.

Mengel, A. & W. Lezius (2000). An XML-based encoding format for syntactically annotated corpora. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC '00)*, Athens, Greece.

Sadler, L., J. van Genabith, A. Way (2000). Automatic F-Structure Annotation from the AP Treebank. In Butt, M. & T. H. King, *Proceedings of the LFG00 Conference*, University of California at Berkeley, CSLI Online Publications, Stanford, CA. `http://www-csli.stanford.edu/publications/`.

Skut, W., B. Krenn, T. Brants, and H. Uszkoreit (1997). An Annotation Scheme for Free Word Order Languages. In *Proceedings of ANLP-97*.

Van Genabith, J., A. Way, L. Sadler (1999). Semi-Automatic Generation of F-Structures from Treebanks. In Butt, M. & T. H. King, *Proceedings of the LFG99 Conference*, University of Manchester, Great Britain, CSLI Online Publications, Stanford, CA. `http://www-csli.stanford.edu/publications/`.

Van Genabith, J., A. Frank, A. Way (2001). Treebank vs. Xbar-based Automatic Feature-Structure Annotation. In Butt, M. & T. H. King, *Proceedings of the LFG01 Conference*, University of Hong Kong, Hong Kong, CSLI Online Publications, Stanford, CA. `http://www-csli.stanford.edu/publications/`.

Zinsmeister, H., J. Kuhn, S. Dipper (2002). TIGER TRANSFER - Utilizing LFG Parses for Treebank Annotation. In Butt, M. & T. H. King, *Proceedings of the LFG02 Conference*, National Technical University of Athens, Greece, CSLI Online Publications, Stanford, CA. `http://www-csli.stanford.edu/publications/`.