A Stochastic Topological Parser for German

Markus Becker and **Anette Frank**
Language Technology Lab
DFKI GmbH
Stuhlsatzenhausweg 3
D-66123 Saarbrücken
{mbecker,frank}@dfki.de

Abstract

We present a new approach to topological parsing of German which is corpus-based and built on a simple model of probabilistic CFG parsing. The *topological field model* of German provides a linguistically motivated, flat *macro structure* for complex sentences. Besides the practical aspect of developing a robust and accurate topological parser for hybrid shallow and deep NLP, we investigate to what extent topological structures can be handled by context-free probabilistic models. We discuss experiments with systematic variants of a topological treebank grammar, which yield competitive results.[1]

1 Introduction

We present a new approach to topological parsing for German which is corpus-based and built on a simple model of probabilistic CFG parsing. Topological parsing is of special interest for shallow pre-processing of languages like German, which exhibit free word order and the so-called verb-second (V2) property. The *topological field model* (Höhle, 1983) is a theory-neutral model of clausal syntax that provides a linguistically well-motivated, but flat *macro structure* for complex sentences. As opposed to chunk-based partial parsing, the topological model is compatible with deep syntactic analysis, and thus perfectly suited for integrated shallow and deep NLP, by guiding deep syntactic analysis by partial, topological bracketing (Crysmann et al., 2002), or for pre-structuring of complex sentences for chunk-based processing (Neumann et al., 2000), as a *divide and conquer* strategy.

Previous approaches to topological parsing of German make use of hand-coded grammars (Wauschkuhn, 1996; Braun, 1999). In this paper we pursue a corpus-based, statistical approach, aiming at a robust parser with high accuracy. We make use of a treebank-induced probabilistic non-lexicalised CFG, following (Charniak, 1996). While this simple model is clearly outperformed by more refined stochastic models for full constituent-structure parsing,[2] our experiment is interesting in showing that for topological parsing a robust parser with high accuracy figures can be obtained with a standard stochastic model of non-lexicalised context-free treebank grammars.

Topological structures are partial or underspecified in that they do not encode internal structure and demarcation of subsentential constituents, i.e. NP, AP, PP or VP constituents. Topological base clauses[3] are characterised by morphological and categorial properties. Still, the topological parsing task is not trivial, in that the boundaries and relative embedding of base clauses and the demarcation of fields in general are not deterministic, and also lexically, or semantically determined. Thus, the complexity of topological parsing lies somewhere between chunk parsing and full constituent-structure parsing. The interesting question we are exploring in our approach is whether this type of syntactic structure can be successfully dealt with using a non-lexicalised PCFG model.

The aim of this paper is three-fold. Besides the practical aspect of (i) developing a robust

[1] The ideas that led to this paper grew from discussions with Feiyu Xu and Jakub Piskorski. The work was in part supported by a BMBF grant to the DFKI project WHITEBOARD (FKZ 01 IW 002). Special thanks go to Bernd Kiefer for providing us with a CFG parser and for his support in technical issues, and to Hubert Schlarb and Holger Neis for manual correction of our test corpus.

[2] E.g. (Collins, 1997) and later work, see (Belz, 2001).

[3] I.e. sentential clauses, see Section 2 for more detail.

and accurate topological parser, to be used for integration with deep syntactic analysis or for cascaded shallow analysis systems, we (ii) investigate how well topological structures can be modeled by context-free probabilistic grammars, while (iii) trying to detect specific phenomena that require more sophisticated models.

The paper is structured as follows. In Section 2 we present the field model for German and describe the creation of a topologically structured treebank, which we derive from the NEGRA corpus (Brants et al., 1997). Section 3 discusses previous work. Section 4 describes our corpus-based stochastic approach to topological parsing. In Section 5 we introduce formal variants of our treebank grammar, which illustrate problematic aspects in topological stochastic parsing, and possible strategies to their solution. Section 6 presents the testing setup and evaluation results for different grammar variants. The results are analysed in detail in Section 7. Section 8 concludes.

2 A Topological Corpus of German

German sentence structure is traditionally analysed in terms of its "field" or topological structure, which is determined by the position of the finite verb in left (LB) or right (RB) bracket position (1). In main clauses the finite verb typically occupies the second constituent position, following the so-called "Vorfeld" (VF) (V2 clauses). The Vorfeld can be missing in yes/no questions or embedded conditional clauses (V1 clauses), as well as in subordinate clauses with complementizer. In subordinate clauses the complementizer (or wh-/rel-phrase) demarcates the LB position, the finite verb is in RB position (VL clauses). Arguments and modifiers between LB and RB occupy the "middle field" (MF), extraposed material is found to the right of the right bracket, in the "Nachfeld" (NF).

(1)	Vorfeld (VF)	Left (LB) Bracket	Middle Field	Right (RB) Bracket	Nachfeld (NF)
V2	topic/ wh-phr.	finite verb	args/ adjs	(verbal complex)	extraposed constituents
V1	- -	finite verb	args/ adjs	(verbal complex)	extraposed constituents
VL	- wh-phr. rel-phr.	compl - -	args/ adjs	(verbal complex) +finite verb	extraposed constituents

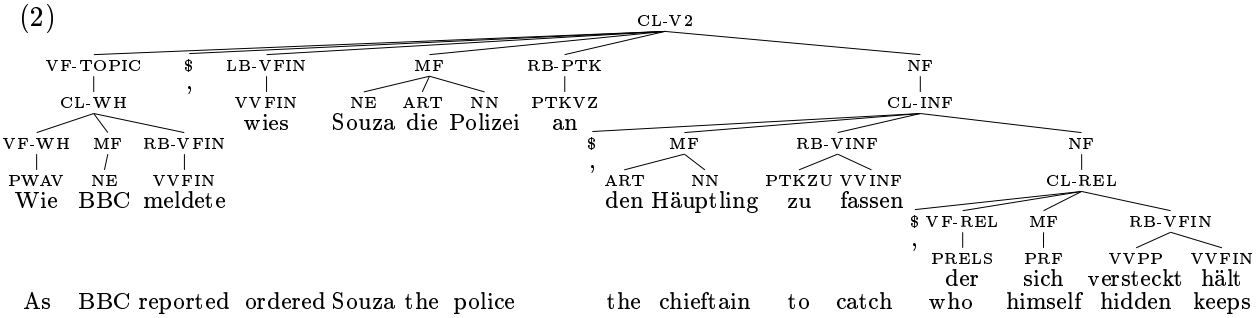
All modern theories of syntax rely – in one way or the other – on this descriptive model of German sentence structure. It is thus straightforward to define mappings from topological to deep syntactic structures of almost any syntactic framework. Its compatibility with deep syntactic analysis makes topological syntactic structure an ideal candidate for interleaving of shallow and deep NLP (Crysmann et al., 2002).

For our *corpus-based* approach, no topologically annotated corpus of German was available. The NEGRA treebank (Brants et al., 1997), a large annotated corpus of German newspaper text, follows an annotation scheme which combines structural and dependency annotations. However, the crucial topological clues, in particular the distinction between fronted or clause-final verb position, as well as the delimitation of pre-, middle- and post-fields are not encoded.

To derive a topological "treebank grammar" from the NEGRA corpus, we applied the *treebank conversion* method of (Frank, 2000). This method is built on a general tree description language, and allows the definition of fine-grained rules for structure conversion. Conversion rules specify partial structural *constraints* and *actions* for tree modifications, which are applied by removing or adding tree description predicates from the trees that satisfy the constraints.

We derived a topological corpus from the NEGRA treebank, by defining linguistically informed conversion rules which exploit additional annotations in the corpus, i.e. indirect linguistic evidence, to assign topological clues. In a second step we induced topological structures by flattening irrelevant *internal structure* within topological fields and introducing *topological category nodes* `DF`, `VF`, `MF`, and `NF` as well as `LB` and `RB` for left and right brackets.[4] Basic clauses are marked with labels `CL` which expand to various patterns of `DF`, `VF`, `LB`, `MF`, `LB`, and `NF` nodes. Basic clauses can be embedded within phrasal fields `VF`, `MF`, `NF`. The resulting structures give (i) an internal structure of basic clauses in terms of *fields* which are internally flattened to POS sequences, and (ii) an overall hierarchical structure of clausal embedding, including coordination. (2) gives an example of a complex topological struc-

[4] `DF` marks a special "discourse field" preceding `VF`, as in *Naja, er kommt halt später*–Well, he will come later.



ture. It illustrates the use of *parameterised* category nodes, which distinguish various types of clauses: `CL-V2,-V1,-INF,-REL,-WH`, pre-fields: `VF-TOPIC,-WH,-REL`, left: `LB-COMPL,-VFIN` and right brackets: `RB-VFIN,-VINF,-VPART,-PTK`.

The automatically derived topological corpus is used for extraction of a stochastic treebank grammar with reserved development and test sections. The test corpus was manually checked and corrected by two independent annotators. Manual correction of the test section yielded 93.0% labelled precision and 93.7% labelled recall of the automatic conversion procedure.

3 Topological Parsing of German

While partial parsers for detection of clausal structure are now available in many varieties and for many languages,[5] this type of parsing approach has always been considered difficult for languages like German. (Wauschkuhn, 1996) was among the first to present a partial parser for German. In a first step, the coarse syntactic clause structure is detected, using indicators like verbs, conjunctions, punctuation, etc. A fine grained analysis is carried out in the second step, by grouping the remaining fields into sequences of minimal "base" NPs or PPs. The analysis is still partial in that attachments of base NPs and PPs are not determined. The grammar is defined as a CFG with feature structures, where grammar rules are annotated with manually adjusted weights for parse ranking. Grammar rules, including the associated weights, are handcoded. (Wauschkuhn, 1996) reports coverage of 85.7% for clausal analysis. No figures are given for precision or recall.

(Braun, 1999; Neumann et al., 2000) report an approach to topological parsing of German, based on cascaded finite state automata. In a first pass, possible verb groups are identified. A second pass identifies subordinate clause structures, using similar cues as (Wauschkuhn, 1996). (Braun, 1999) carried out an evaluation over 400 sentences and reports coverage of 94.3%, precision of 89.7% and recall of 84.75%.

While these approaches are similar to our work in inducing topological structure from key linguistic indicators, they suffer from several problems. (i) Hand-coding of rules is laborious[6] and likely to miss out rare or exceptional phenomena, including ungrammatical constructions. (ii) Ambiguities are either resolved by manually assigned weights, or simply by using a greedy strategy (Braun, 1999). (iii) These approaches heavily exploit prescriptive punctuation rules. This is problematic for performance influenced deviations from standard punctuation or less standardised text sorts, leading to either a loss of coverage, or accuracy.

4 A Stochastic Topological Parser

In response to these problems we investigate a *corpus-based, stochastic* approach to topological parsing. It has been demonstrated[2] that stochastic parsing can achieve high figures of robustness and accuracy, while mostly restricted to purely constituent-based syntactic analysis.

For our task of topological parsing, we investigate the adequacy of the very simple, non-lexicalised model of (Charniak, 1996), if applied to rather flat, topological structures. Our working hypothesis was that the model should perform well, even if not lexicalised, since (i) there are less attachment decisions, due to the rather flat target structures. (ii) Topological structures as such, as well as attachment decisions for base clauses are less dependent on lexical information, than, e.g., attachment of PPs. Finally, (iii) a *corpus-based* stochastic grammar has a

[5] See for example (Ait-Mokhtar and Chanod, 1997; Gala-Pavia, 1999) for English, French, and Spanish.

[6] Wauschkuhn uses 366 rules for clausal analysis.

better chance to account for exceptional constructions and performance-influenced input.

Following the method of (Charniak, 1996) we extract a context free grammar from the corpus described in Section 2. From this grammar we derive formal grammar variants (see Section 5). Rule probabilities are estimated using maximum likelihood. We employ a flexible and efficient CFG chart parser (Kiefer and Scherf, 1996), which we extended to manage rule probabilities. Currently, we let the parser compute the full search space. N-best parse trees are efficiently determined by applying the Viterbi algorithm over packed tree structures.

5 Variations of Topological Grammars

As part of our experimental setup we induce formal variants of the topological treebank grammar. The aim is to explore different strategies, or 'models', and how well they perform in terms of coverage and accuracy.[7] These grammar variants illustrate problematic aspects in topological stochastic parsing, and strategies to their solution. In particular, we discuss (a) parameterisation of field categories, (b) alternative approaches to punctuation, (c) the use of binary field structures to address sparseness problems, and (d) the effects of grammar pruning.

(a) Parameterised categories Our topological corpus defines maximally informative structures where topological categories are associated with more fine-grained syntactic labels. For instance, relative clauses, which dominate a finite right bracket daughter `RB-VFIN`, are marked `CL-REL`, as opposed to verb-second clauses `CL-V2` with finite left bracket (`LB-VFIN`) (see (2)). A `VF` category that contains a relative pronoun will be marked `VF-REL`. Such fine-grained labels implicitly encode a larger syntactic context (cf. (Belz, 2001)): for example, a relative pronoun in `VF-REL` predicts (through coocurrence data in the corpus) that it is dominated by a grandfather category `CL-REL`, which takes a right bracket daughter `RB-VFIN`, as opposed to a left bracket daughter.

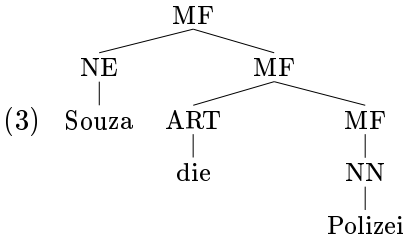
We extract grammar variants with and without parameterised categories, to investigate to which extent a more fine-grained and implicitly contextualised grammar helps to increase accuracy in a topological model of syntax.

(b) Punctuation The maximal decoration of a tree contains punctuation marks like commas, quotes, colons, etc.[8] While the correct attachment of punctuation marks is not part of our evaluation, the guiding intuition was that punctuation should help to identify clause boundaries. On the other hand, irregularities in punctuation setting cause noise in the data, increases grammar size, and could cause coverage problems. We compare the performance of grammar variants with and without punctuation.

(c) Binarisation Phrasal topological fields `VF`, `MF`, `NF` are underspecified for constituent boundaries of NPs, PPs, etc. The fields are radically flattened, directly expanding to sequences of POS categories. We expect a great variety of POS sequences as expansions of field categories, but at the same time reckon with considerable sparseness problems, due to unseen POS sequences.

To address this problem, we introduce (right-branching) binary field structures. The flat structure for the two constituents *Souza die Polizei* in (2) is transformed to the tree (3). Learning rules from binary subtrees effectively induces a unigram language model where the number of "cells" corresponds to the rather small number of POS categories. Again, we experiment with flat vs. binary grammar versions, to test their respective coverage and accuracy.

(3)



(d) Pruning Due to automatic transformation, the topological corpus contains some ill-formed structures. We test whether noise in the grammar can be reduced by pruning single occurrences of rules. We compare the performance of pruned and unpruned grammars.

[7] Henceforth we use accuracy as a measure for both precision and recall – often referred to as f-measure.

[8] Full stops, brackets, and hyphens were deleted.

6 Experiments and Results

Experimental setup The NEGRA corpus was split into randomised sections for training (16476), development (1000) and testing (1058), plus further held-out data for later experiments. For training and development we used the automatically derived topological corpus, while the test data was manually corrected (Section 2).

To test the performance of the grammar independently from a tagger, the input to the parser consists of the manually disambiguated POS sequences of the test corpus.[9]

Evaluation Measures For evaluation we employ the PARSEVAL measures of labeled recall and precision and crossing brackets, as well as complete match, i.e. full structure identity.[10] To accommodate for the differences between grammar versions, evaluation was conducted as follows. The evaluation measures in Tables 1 and 2 disregard punctuation and are based on simple node labels, i.e. category parameters are stripped. Finally, to allow clear comparison between binarised and flat grammar versions binarised parse trees are compiled to flat trees before evaluation against flat target trees.[11]

Results We conducted systematic tests for all combinations of grammar variants: ±para (parameterised categories), ±bin (binarised), ±pnct (punctuation), ±prun (pruning single rule occurrences), see results in Table 1.

Tables 2 and 3 give more detailed evaluation figures for the best performing model (v1) para+.bin+.pnct+.prun+. Table 2 lists labeled recall and precision results for individual topological categories. Field categories `VF...NF` receive high figures above 90%, to the exception of `NF`, yet with lower overall proportion (quota).

Table 3 reports alternative evaluation figures, namely evaluation by disregarding category parameters (param −), or by evaluating on complex category labels (param +); and by taking or not punctuation into account (punct +/−).

Finally, Fig. 4 displays a learning curve for stepwise extension of the training corpus.

[9] 8 sentences were set apart due to wrong POS tags.

[10] We verified our results using the evaluation tool `evalb` by Satoshi Sekine `http://www.cs.nyu.edu/cs/projects/proteus/evalb/`.

[11] Evaluating labeled recall and precision on binarised trees would yield unduly high figures, due to a high number of field-internal trivial assignments.

7 Discussion of Results

Table 1 shows better performance of grammars v1-8 using *parameterised categories*, as opposed to the complementary versions v9-16. Parameterised grammars make use of a richer structure, which is mapped to coarser topological categories for evaluation.[12] The implicit contextualisation in category labels clearly improves parsing results. While the rule set grows, a relative loss of coverage is only visible for non-binarised versions v5-8 as opposed to v13-16.

Binarisation shows dramatic effects in coverage and accuracy. Binarised grammars are smaller than their flat counterparts, but far less constrained, allowing the derivation of virtually any POS sequence. Flat grammars suffer from lack of coverage, especially those using rich category labels and/or punctuation. We see dramatic differences of about 100% complete match improvement between v6/v2, v8/v4, v16/v12, and significant contrasts in LP/LR and CB measures. Thus, binarisation solves the sparseness problem for flat topological CFGs without jeopardising accuracy.

Using *punctuation* in parsing leads to improved accuracy measures, yet only in binarised grammars, where sparseness problems are circumvented. Flat grammars with punctuation show lower coverage than their counterparts – higher accuracy measures are probably due to lower coverage. Use of punctuation is similar to parameterisation of labels, in that grammar-internally it helps to discriminate fields, while for evaluation it is filtered from the parse trees.

Pruning of single rule occurrences leads to significant reduction in grammar size, in particular for non-binarised grammars. Here, pruning incurs significant loss in coverage. This is expected, since extremely flat rules are likely not to re-occur several times. For binarised grammars pruning yields rule sets of about 1/3, with almost unchanged 100% coverage. Our hypothesis was that pruning improves the quality of the grammar by eliminating noise imported by automatic treebank conversion. This is confirmed, in all binary grammars, by improved accuracy measures. Since in binary grammars generic field rules are binarised and frequently occuring, rule pruning is likely to eliminate noise.

[12] Thus, parameterisation corresponds to the notion of internal and external tagsets in (Brants, 1997).

#	version (trained on 16476 sents.)	gram size	coverage in %	coverage len	perf. match in %	perf. match len	LP in %	LR in %	0CB in %	2CB in %
1	**para+.bin+.pnct+.prun+**									
	a) ≤ 40	**867**	**100.0**	**14.6**	**80.4**	**13.1**	**93.4**	**92.9**	**92.1**	**98.9**
	b) all	867	99.8	15.9	78.6	13.7	92.4	92.2	90.7	98.5
2	para+.bin+.pnct+.prun-	2308	99.9	14.6	79.1	13.0	93.3	92.7	92.1	99.1
3	para+.bin+.pnct-.prun+	679	100.0	14.6	80.8	13.1	92.8	91.7	89.1	98.0
4	para+.bin+.pnct-.prun-	1917	99.9	14.6	79.6	13.0	92.2	91.5	89.0	97.9
5	para+.bin-.pnct+.prun+	2962	57.5	10.3	49.7	5.7	63.2	79.9	59.3	87.6
6	para+.bin-.pnct+.prun-	19536	88.4	13.6	37.5	6.5	54.0	73.1	48.0	78.8
7	para+.bin-.pnct-.prun+	2839	67.2	11.6	45.8	6.0	59.8	76.5	52.7	83.3
8	para+.bin-.pnct-.prun-	18365	92.5	13.9	38.9	6.8	55.2	73.6	47.5	78.6
9	para-.bin+.pnct+.prun+	634	100.0	14.6	74.9	12.4	89.3	89.0	87.5	97.9
10	para-.bin+.pnct+.prun-	1827	99.9	14.6	72.7	12.3	88.3	88.2	86.7	97.7
11	para-.bin+.pnct-.prun+	489	100.0	14.6	71.6	11.9	86.0	84.5	80.6	95.7
12	para-.bin+.pnct-.prun-	1528	99.9	14.5	70.4	11.8	85.6	84.3	80.9	95.4
13	para-.bin-.pnct+.prun+	2756	76.4	12.8	37.4	5.6	53.4	71.7	46.6	80.1
14	para-.bin-.pnct+.prun-	18979	94.9	14.2	34.6	6.4	53.4	71.5	46.9	80.4
15	para-.bin-.pnct-.prun+	2675	80.4	13.3	36.9	5.8	53.2	71.1	45.7	80.5
16	para-.bin-.pnct-.prun-	17885	96.6	14.2	35.4	6.5	53.7	70.7	46.8	82.3

Table 1: Results for systematic grammar variations (sentence length ≤ 40, except 1b)

Category	LP in %	LP quota	LR in %	LR quota
CL	88.9	24.3	92.2	23.2
MF	93.2	23.8	93.1	23.7
LB	99.6	17.9	99.4	17.8
VF	96.1	16.3	91.8	16.9
RB	96.3	13.7	95.8	13.7
NF	82.6	3.6	73.4	4.1
S	4.8	0.3	5.3	0.3
DF	16.7	0.1	6.7	0.2
all	93.4	100.0	92.9	100.0

Table 2: Category-specific evaluation (v1,≤40)[13]

eval param	eval punct	perf. match in %	perf. match len	LP in %	LR in %
−	−	80.4	13.1	93.4	92.9
+	−	79.6	13.1	92.7	92.2
−	+	78.5	12.8	92.1	91.6
+	+	77.7	12.8	91.5	91.0

Table 3: Different evaluation schemes (v1,≤40)

In sum, our best performing model (v1) makes use of a maximally discriminative symbolic grammar (parameterised categories, punctuation), resolves sparseness problems by rule binarisation, and can afford rule pruning to eliminate noise. Applied to full sentence lengths (v1b) we note a drop in performance, but insignificantly so for coverage, and only by 1% in LP and 0.7% in LR.

Table 3 details alternative evaluation measures. Evaluation on parameterised categories incurs a slight drop in accuracy, but in high ranges.[14] Evaluation of punctuation attachment – which is of little importance – yields a further drop.

The *learning curve* in Fig. 4 is surprising in that we obtain relatively high performance from rather small training corpora and grammar sizes (size grows almost linearly from 313 to 2308).[15] Saturation regarding coverage and accuracy is obtained around training size 6000.

Finally, we determined phenomena that call for stronger contextualisation or lexicalisation. A case in point are verb-second (V2) sentences with a fronted V2 clause in Vorfeld position (i.e. with VF-V2 categories), which allow an alternative analysis as coordinate clauses with shared subjects. This type of construction was frequently mis-analysed as a coordination structure since this structural ambiguity cannot be

[13] S-categories were used for non-standard base clauses, e.g. gapping, that did not fit the topological model.

[14] These measures are relevant for integration of shallow and deep NLP (Crysmann et al., 2002), as parameterised categories provide highly discriminative information that can be used to guide deep syntactic processing.

[15] Note, however, that the curve pertains to a robust, binarised grammar. We chose v2 (prun−) in order not to unduly penalise small grammars. Lack of pruning could explain the scattered values for lower training sizes.

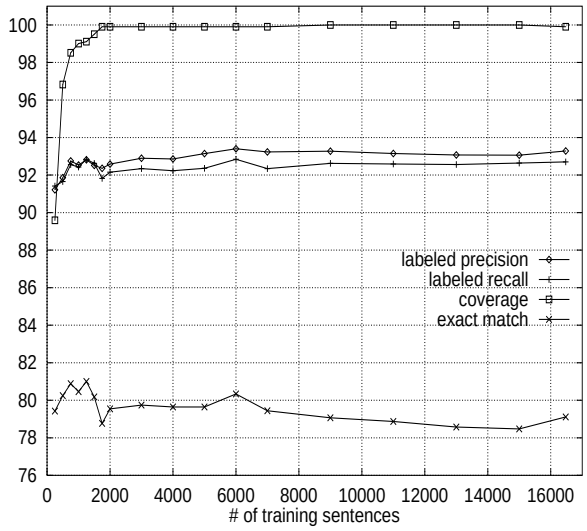


Figure 4: Learning curve (version v2)

resolved on the basis of morphological or topological criteria. A promising strategy to enhance our model is (targeted) lexicalisiation, as these constructions typically occur with a specific type of "reporting" verbs.

8 Conclusion and Future Work

We presented a topological parser for German, using a standard PCFG model trained on an annotated corpus. We have shown that for the task of topological parsing a non-lexicalised PCFG model yields competitive results. We investigated various grammar versions to illustrate problematic aspects in stochastic topological parsing. Category parameterisation (i.e. contextualisation) and punctuation were shown to increase accuracy. Binarisation results in high coverage figures. Pruning of single rule occurrences eliminates noise in the automatically constructed training corpus.

The complexity of topological parsing lies somewhere between the complexity of chunk parsing and full constituent structure parsing. Our results indicate that a standard PCFG model is appropriate for the chosen task, but could possibly be enhanced by lexicalisation.

In future work we will explore extension to a lexicalised model, and investigate cascaded stochastic parsing, by applying a specialised stochastic chunk parsing model to phrasal fields, to obtain full constituent structure parses. Further we will integrate the TnT tagger (Brants, 2000) to investigate the robustness of the parser with respect to tagging errors, and extend the model to a free parsing architecture.

References

S. Ait-Mokhtar and J. Chanod. 1997. Incremental Finite-State Parsing. In *Proceedings of ANLP-97*.

B. Crysmann, A. Frank, B. Kiefer, S. Müller, G. Neumann, J. Piskorski, U. Schäfer, M. Siegel, H. Uszkoreit, F. Xu, M. Becker, and H-U. Krieger. 2002. An Integrated Architecture for Deep and Shallow Processing. In *Proceedings of ACL 2002*, University of Pennsylvania, Philadelphia.

A. Belz. 2001. Optimisation of corpus-derived probabilistic grammars. In *Proceedings of Corpus Linguistics 2001*, pp. 46–57.

T. Brants, W. Skut, and B. Krenn. 1997. Tagging Grammatical Functions. In *Proceedings of EMNLP*, Providence, RI, USA.

T. Brants. 1997. Internal and external tagsets in part-of-speech tagging. In *Proceedings of Eurospeech*, Rhodes, Greece.

T. Brants. 2000. TnT - A Statistical Part-of-Speech Tagger. In *Proceedings of the ANLP-2000*, Rhodes, Greece.

C. Braun. 1999. Flaches und robustes Parsen Deutscher Satzgefüge. Master's thesis, Saarland University.

E. Charniak. 1996. Tree-bank Grammars. In *AAAI-96. Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pp. 1031–1036. MIT Press.

M. Collins. 1997. Three generative models for statistical parsing. In *Proceedings of the ACL-97*, pp. 16–23.

A. Frank. 2000. Automatic F-structure Annotation of Treebank Trees. In M. Butt and T.H. King, (eds), *Proceedings of the LFG00 Conference*, CSLI Online Publications, Stanford, CA.

N. Gala-Pavia. 1999. Using the Incremental Finite-State Architecture to create a Spanish Shallow Parser. In *Proceedings of XV Congres of SEPLN*, Lleida, Spain.

T. Höhle. 1983. Topologische Felder. University of Cologne.

B. Kiefer and O. Scherf. 1996. Gimme more HQ parsers. The generic parser class of DISCO. Ms., DFKI, Saarbrücken, Germany.

G. Neumann, C. Braun, and J. Piskorski. 2000. A Divide-and-Conquer Strategy for Shallow Parsing of German Free Texts. In *Proceedings of ANLP*, pp. 239–246, Seattle, Washington.

O. Wauschkuhn. 1996. Ein Werkzeug zur partiellen syntaktischen Analyse deutscher Textkorpora. In D. Gibbon, (ed), *Proceedings of the Third KONVENS Conference*, pp. 356–368, Berlin. Mouton de Gruyter.