A broad-coverage parser for German based on defeasible constraints

Kilian A. Foth Michael Daum Wolfgang Menzel
Natural Language Systems Group
Hamburg University
D-22527 Hamburg
Germany
{foth,micha,menzel}@nats.informatik.uni-hamburg.de

No Institute Given

Abstract. We present a parser for German that achieves a competitive accuracy on unrestricted input while maintaining a coverage of 100%. By writing well-formedness rules as declarative, defeasible constraints that integrate different sources of linguistic knowledge, very high robustness is achieved against all sorts of language error.

1 Introduction

Although most linguists agree that natural language to a large degree follows well-defined rules, it has proven exceedingly difficult to actually define these rules well enough that a computer could reliably assign the intuitively correct structure to natural language input. Therefore, almost all current approaches to parsing constitute compromises of some kind:

1. Often the goal of full syntactic analysis is abandoned in favour of various kinds of *shallow parsing*, which is all that is needed for many applications. Instead of a full syntax tree, e.g., only the boundaries of major constituents [Schmid and Schulte im Walde2000] or topological fields [Becker and Frank2002] are computed.
2. Instead of casting the language description into linguistically motivated rules, a probability model is induced automatically from a large amount of past utterances, which is then used to maximize the similarity to previously seen structures [Collins1999]. This approach is robust and efficient, but relies on a large amount of existing data, and the resulting model is difficult to comprehend and extend.
3. Formalisms that do perform deep analysis by following explicit rules typically often have to be restricted to particular subsets of linguistic

constructions or to particular domains. Also, their robustness and coverage (the performance for ungrammatical and extragrammatical input, respectively) is often rather low on realistic data.

We present a parsing system that tries to avoid all three compromises to the extent possible at the current time. Instead of a derivation grammar, we employ a declarative formalism in which well-formedness conditions are expressed as explicit constraints (which take the form of logical formulas) on word-to-word dependency structures. Since there is no limit to the complexity of these formulas, *every* conceivable well-formedness condition can be expressed as a constraint. Although the formal power of this approach would theoretically make the parsing problem intractable, we have found that approximative solutions methods yield good results in practice.

All grammar rules are ordered by a preference measure that distinguishes e.g. less important rules of style from more important fundamental grammar rules. This not only ensures robust behaviour against all kinds of deviant input, but also allows us to integrate the information from external shallow parsers, such as a part-of-speech tagger, without becoming dependent on their results.

The output of the system consists of labelled dependency trees rather than constituent-based structures. Although this is mainly a consequence of casting the parsing problem in the form of a constraint optimization problem, it also has benefits for the processing of languages like German with a relatively free word order.

2 The WCDG parsing system

Weighted constraint dependency grammar (WCDG) [Schröder2002] is an extension of the CDG formalism first described by [Maruyama1990]. It describes the structure of natural language as a set of labelled subordinations: each word is either subordinated to exactly one other word or considered the root of the syntax tree (also called a NIL subordination). See Figure 1 for an example. Each subordination is annotated with one of a fixed set of labels such as ‘subject’, ‘direct object’ or ‘indirect object’, but no larger groups of words carry labels. This means that there is no direct equivalent to the constituents postulated by phrase structure grammar.

Since there are no constituents, there are also no generative rules along the lines of ‘S $\rightarrow$ NP VP’; these are often problematic for languages

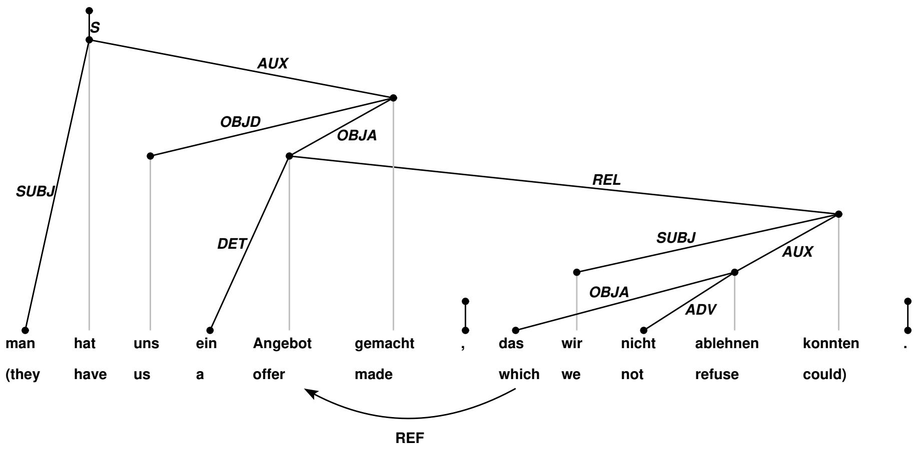

Fig. 1. Dependency analysis of a German sentence.

with free or semi-free word order since they mingle dominance principles with linear precedence rules. Instead, the grammar rules take the form of declarative constraints about permissible subordinations. These constraints can reference the position, reading and lexical features of the concerned word forms, as well as features of neighbouring dependency edges. Every subordination that is not forbidden by any constraint is considered valid. The goal of the parser is to select a set of subordinations that satisfies all constraints of the grammar.[1]

As an example of a constraint, consider the rule that normal nouns of German require a determiner of some kind, either an article or a nominal modifier, unless they are mass nouns. This rule can be formulated as a constraint as follows:[2]

```
{X:SYN} : 'missing determiner' : 0.2 :
  X@cat = NN
  ->
  exists(X@mass_noun) |
  has(X@id, DET) |
  has(X@id, GMOD);
```

It states that for each subordination on the syntax level (`SYN`), a word with the category 'normal noun' (`NN`) must either bear the feature 'mass

[1] Categorial and morphological ambiguity must also be resolved for each word in a sentence. While often glossed over, this task is quite difficult in languages with a rich morphology; the average German adjective form has 10 morphosyntactic variants that are relevant for agreement.

[2] This constraint is considerably simplified for exposition purposes, since the rule actually has many systematic exceptions.

noun' or be modified by a determiner (label `DET`) or a genitive modifier (label `GMOD`).

Each constraint bears a score between `0.0` and `1.0` that indicates its importance relative to other constraints. The acceptability of an analysis is defined as the product of the scores of all instances of constraints that it violates. This means that constraints with a score of `0.0` must be satisfied if at all possible, since violating them would yield the worst possible score.[3] Note that the score of the determiner constraint is `0.2`, which means that missing determiners are considered wrong but not totally unacceptable. In fact, many other constraints are considered more important than the determiner rule.

By assigning values other than `0.0` to all rules that might conceivably be broken, a prescriptive grammar can easily be converted to one that is robust against all kinds of language errors, while still preferring conforming over deviant structures wherever possible. In fact, a coverage of 100% can be guaranteed with an appropriately written grammar. This is achieved by giving those constraints which could forbid *all* structures for a given utterance a higher value.

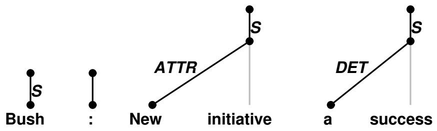

Fig. 2. Analysis of an elliptical utterance.

One easy way of ensuring this property is to allow surplus words to form the roots of isolated subtrees. This is possible because WCDG by itself does not enforce a unique NIL subordination. Note that in some cases a set of fragments actually is the most faithful rendition that is possible in a dependency formalism. Consider the the typical elliptical headline *"Bush: New initiative a success"*, where the verbs "said" and "is" have been elided. This should actually be modelled as a forest of three tree fragments (cf. Figure2). Assigning a heavy penalty to fragments that are not finite verbs ensures that they will be avoided unless absolutely necessary. *Partial parsing* can thus be achieved simply by not constraining the number of NIL subordinations.

[3] However, the parser will still return such a structure if no analysis with a positive score can be found.

Since there are no generative rules, WCDG does not have a derivation component. Instead, parsing a sentence defines a multidimensional optimization problem that can be solved by all known methods of constraint optimization. Where the problem instance is too large to be solved via complete search, good success has been achieved with transformation-based heuristic solution methods [Daum and Menzel2002] [Foth and Menzel2000] that approximate the acceptability model defined by the constraints. In a large-scale grammar with many defeasible constraints, different analyses of the same sentence usually carry at least slightly different scores, so that full disambiguation is achieved. If different analyses happen to have the same score, the one that the transformation process finds first is usually selected as the output of the parser.

3 A comprehensive grammar of German

We have developed a grammar for the WCDG formalism that is intended to cover the entire realm of modern German. So far we have considered phenomena from about 28,000 sentences from various text types, such as newspaper text, Verbmobil dialogues, classical literature, law texts, and online newscasts. Although obscure extragrammatical constructions are still occasionally encountered, the great majority of new input can be modelled accurately, and we do not expect major changes to the existing rules to become necessary. Many problem cases involve constructions that are probably impossible to represent accurately in a word-based dependency tree, such as complicated conjunctions or heavy ellipsis.

As an example of our dependency representation, consider again the analysis of the sentence *"Man hat uns ein Angebot gemacht, das wir nicht ablehnen konnten." (They made us an offer we could not refuse.)* in Figure 1. Note that there are two instances of *nonprojective* structure in this sentence, i.e. the vertical projection lines must cross dependency edges in two places no matter how the tree is drawn. WCDG does not require its structures to be projective, although constraints to that effect can easily be written and then have a considerable disambiguating effect. We can thus represent phenomena such as German auxiliary groups or extraposed relative sentences faithfully by writing constraints that enforce projectivity in general, but make exceptions for edges with these particular labels.

The referential relationship between the relative pronoun 'das' and its antecedent 'Angebot' cannot be represented on the syntax level, since both words already function as direct objects. Instead, it is represented by an additional edge which connects the two words on an extrasyntactical 'reference' level. The agreement of gender and number that must hold between the two words can thus be checked directly.[4]

The grammar currently consists of about 700 handwritten constraints, although much information is lexicalized, in particular valence information of verbs and prepositions. An extensive lexicon of full forms is used that includes all closed-class forms of German. Among the open-class forms, around 6,000 verb stems and 25,000 noun stems are covered; compound forms can be deduced from their base forms on the fly to deal with the various types of German compounds. As a last resort, lexicon templates provide skeleton entries for totally unknown words; these contain not much more than the syntactic category.

We estimate the overall effort for our grammar of German at about 5 work-years. A large part of this work, however, consisted in creating manual syntax annotations for testing the predictions of the constraint model; this corresponds to the effort needed to create the treebank that a stochastical parser is then trained on. Another very time-consuming task was to collect and classify open-class lexicon entries with respect e.g. to their inflection and valence features. While helpful for disambiguation in many cases, this information is not strictly necessary for running the parser.

4 Evaluation

Exact solution of the optimization problem posed by a large constraint grammar is often computationally infeasible for long input sentences, so that incomplete solution methods must usually be employed. Therefore, in addition to *model errors*, where the linguistic intuition contradicts the model defined by the actually implemented grammar, *search errors* can occur, where the parser output in turn differs from the modelled optimum because the heuristic optimization algorithm terminated in a local peak.

In the case of the earlier example sentence, no problems occur: the grammar assigns the highest score to the desired analysis shown in Figure 1, and the solution algorithm actually finds it. In general, this is not always the case, particularly for long sentences. Although all individual cases we have investigated suggest that the accuracy of the model established by our grammar is higher than the actual performance of the

[4] Although other pronouns, and possibly nouns, can refer to other words, these relationships are usually outside the scope of a sentence-based parser because they transcend sentence boundaries. Therefore, this grammar describes reference edges *only* for the relative pronouns.

parser (i.e., search errors decrease rather than increase the overall accuracy), this is not of practical use because the better solutions cannot be efficiently computed. Therefore, all results reported in this paper use the strictest possible measure: only the accuracy of the computed structures with respect to the intended (annotated) ones is reported.

Since our parser always returns exactly one complete syntax structure, recall and precision are both identical to the percentage of correct versus incorrect dependency edges. According to this measure, our parser typically computes between 80% and 90% of correct dependency attachments for written German. This is near the performance of the Collins parser for English [Collins1999], but somewhat below the results of [Tapanainen and Järvinen1997], who report precisions above 95% (also for English), but do not always attach all words. More relevant are the results described by [Dubey and Keller2003], who analysed 968 sentences (with at most 40 words) from the NEGRA corpus of German newspaper text. Reimplementing Collins' parser for German, they only achieved a labelled precision and recall of 66.0%/67.9%. This suggests that German syntax is considerably more difficult to analyse for this kind of parser than English. [Dubey and Keller2003] improved the parsing model to 70.9%/71.3% by considering sister-head dependencies instead of head-head dependencies.

For purposes of comparison we analysed the same section of the NEGRA treebank with our own parser. We first translated the phrase treebank to dependencies automatically, with only few edges corrected to conform to our annotation guidelines [Daum et al.2004]. The accuracy of parsing is then measured by counting the number of correctly computed dependency edges. Input is first annotated with part-of-speech tags by the statistical trigram tagger TnT, and some typical errors of the trigram model are automatically corrected by about 20 handwritten rules. The resulting scores are integrated into the constraint grammar with a constraint that disprefers word forms with improbable categories.

We employ a heuristic transformation-based solution method that first constructs a complete syntax tree in linear time, and then tries to transform it so as to remove errors that were diagnosed by violated constraints. Three passes are made over each sentence; in the first phase, only subordinations between nearby words are allowed, and the resulting partial trees are recombined by their roots in the second phase. A third pass then tries to repair any remaining errors; only this phase investigates the entire space of possible subordinations. We have found that the phase-based approach yields better results than tackling the full optimization problem to begin with [Foth and Menzel2003].

Table 1 gives the results; altogether 87.1% of all dependency edges are attached correctly by the parser (85.0% if we demand correct attachment and the correct label). Since the previous work reported constituent-based precision and recall figures, a direct comparison is not possible, but it seems safe to say that both parsing accuracy and coverage are somewhat higher in our system. A more telling comparison would be achieved by transforming the phrase structure results of the sister-head-dependency model into dependency structures with the same tool used for transforming the treebank, and measuring the edge accuracy directly.

Sentence length correlates with a slowly increasing number of parsing errors; to a certain degree this simply reflects the greater number of plausible but wrong subordinations in long sentences. However, the incompleteness of the solution algorithm also contributes to this: as the search space increases, more alternatives are overlooked, so that the method effectively becomes less and less complete.

Corpus	attachment	labelled attachment
all sentences	87.1%	85.0%
<60 words	87.2%	85.1%
<40 words	87.8%	85.7%
<20 words	91.2%	88.8%
<10 words	93.5%	91.5%

Table 1. Parsing results for NEGRA sentences.

By default, the algorithm terminates when it has repaired, or failed to repair, all important constraint violations (those whose score lies below a configurable threshold). However, the algorithm exhibits the *anytime property*: processing can be interrupted at any time, if necessary, and the current analysis returned. This allows a trade-off to be made between parsing time and quality of results, since improvements to the score of an analysis generally coincide with a better accuracy. For the results in Table 1, we allowed the parser to spend up to 600 seconds on a sentence (on 1533-MHz Athlon PC processors), although for over 90% of all sentences it terminates before this time limit. If the maximal processing time is decreased, accuracy slowly decreases, as shown in Table 2.

Although our grammar of German was originally developed to parse the text of online technical newscasts, it has been successfully used on a variety of other text types. Table 3 gives results of analysing various

Time limit	attachment	labelled attachment
600 seconds	87.1%	85.0%
400 seconds	86.9%	84.7%
200 seconds	86.1%	83.9%
100 seconds	84.5%	82.3%
50 seconds	81.9%	79.4%

Table 2. Parsing results with reduced runtime limit.

corpora under the same conditions as those in Table 1. In general, the accuracy is comparable to that on the NEGRA corpus; however, performance is measurably affected by text type as well as sentence length. In particular, classical literature appears more difficult to parse, since it employs many constructions considered marked or extragrammatical by our current model of German. Transcriptions of spontaneous speech also pose some problems, since they contain many more sentence fragments than written text and no disambiguating punctuation.

Source	Text type	avg. length	attachment	labelled attachment
Trivial literature	predominantly dialogue	14 words	92.2%	89.9%
Law text	legal language	19 words	88.3%	86.6%
Verbmobil dialogues	spontaneous speech	8 words	88.5%	83.9%
Online news	technical jargon	23 words	87.7%	86.0%
Serious literature	German of 1900	34 words	78.0%	75.4%

Table 3. Parsing results on different text types.

5 Related Work

Except for [Dubey and Keller2003] there seems to be no other attempt to evaluate a syntactic parser of German on a gold standard annotation using the common PARSEVAL methodology. Accordingly, there is no standard setting available so far which could facilitate a direct parser comparison, similar to the established Penn Treebank scenario for English. However, there have been evaluation efforts for more shallow types of syntactic information, e.g. chunks or topological fields.

Parsing into topological fields like Vorfeld, Mittelfeld, and Nachfeld usually is considered a somewhat easier task than the construction of a full-fledged constituency analysis, because it clearly profits from the easily identifiable position of the finite verb in a German sentence and avoids any decision about the assignment of syntactic functions to the constituents. [Braun2003] presents a rule-based approach to topological parsing of German and reports a coverage of 93.0%, an ambiguity rate of 1.08, and a precision/recall of 86.7% and 87.3% respectively on a test set of 400 sentences. [Becker and Frank2002] trained a stochastic topological parser on structures extracted from the NEGRA-Treebank and tested it on 1058 randomly selected sentences with a maximum length of 40 words. They achieved a labelled precision/recall of 93.4% and 92.1%.

Chunking, i.e. the segmentation of a sentence into pieces of particular type (usually NPs and PPs), is an approach to shallow sentence processing sufficient for many practical purposes. [Brants1999] used cascaded Hidden Markov Models to chunk sentences into a hierarchy of structurally embedded NPs and PPs with a maximum tree depth of nine. He achieved a precision/recall of 91.4% and 84.8% in a cross evaluation on the 17,000 sentences of the NEGRA-Treebank. [Schmid and Schulte im Walde2000] trained a probabilistic context-free grammar on unlabelled data and used it to segment sentences into NP chunks. They evaluated the system on 378 sentences from newspaper text and obtained a precision/recall of 93% and 92% if only the range of a chunk is considered, which decreased to 84% and 83% if also the case of an NP has to be determined.

Special evaluation methodologies have been designed for several parsers of German. Among them is the one used to evaluate the quality improvement during the development of the Gramotron parser [Beil et al.2002]. The method is partly motivated by lexical semantic clustering, the ultimate goal of the project, although the authors admit that a direct evaluation of head-to-head dependencies as proposed by [Lin1995] would be more appropriate. Besides avoiding overtraining by monitoring the cross-entropy on held-out data, a linguistic evaluation was carried out focussing on selected clauses types. It is based on a randomly sampled subcorpus of 500 clauses and measured the quality of determining noun chunks and subcategorization types. The precision of chunking reached 98% if only the range was considered and decreased to 92% if the chunk type was also included. The subcategorization type of verbs was computed with a precision between 63% and 73% depending on the kind of clauses under consideration.

[Langer2001] evaluated a GPSG parser with the capability of additional processing rules (of unlimited generative power) on 62,073 sentences from German newspaper text. Of the sentences with 60 words or less, 34.5% could be analysed, with an average of 78 solutions per sentence. On a self-selected test corpus with an average length of 6 words, coverage rose to 83%. Only an indirect measure of accuracy is given: in

80.8% of the newspaper sentences, giving the parser access to partially annotated target structures (chunks and some POS information) did not change its output.

[Neumann and Flickinger2002] evaluated their HPSG-parser based on the DOP-model with 1000 sentences from the Verbmobil-domain. They measured coverage (70.4% if combined), and the runtime performance of the system.

[Wauschkuhn1995] developed a two step parser computing a clause level structure which clause-internally is kept completely flat. The system is evaluated on 1097 newspaper sentences only with respect to its coverage (86.7% if run on manually tagged input and 65.4% for automatically tagged text) and the achieved degree of disambiguation (76.4% for manually and 57.1% for automatically tagged input).

6 Conclusions

A competitive parsing system for unrestricted German input has been described that is largely independent of domain and text type. Total coverage is ensured by means of defeasible, graded constraints rather than hard grammar rules. We find that using a powerful formalism that allows all conceivable rules to be actually used allows for better grammar development than using a tractable but less expressive one: even a theoretically intractable formalism is useful in practice if its model can be approximated well enough. While casting syntax analysis as a multidimensional optimization problem leads to high requirements of processing time, this inefficiency is somewhat mitigated by the anytime property of the transformational optimizer that allows the user to limit the desired processing time.

A greater concern is certainly the great amount of expert knowledge needed to create a complete constraint grammar. We propose that for deep analysis of German syntax, the greater precision achievable through handwritten rules justifies the extra effort compared to approaches based purely on machine learning. Although automatic extraction of constraints from annotated corpora remains a goal of further research, so far we have achieved greater success by hand-selecting and scoring all grammar rules, since this allows the grammar writer to make conscious decisions about the relative importance of different principles.

An online demonstration of our system can be accessed at `http://nats-www.informatik.uni-hamburg.de/Papa/ParserDemo`.

The source code of the parser and the grammar is available under the General Public License at `http://nats-www.informatik.uni-hamburg.de/download`.

References

[Becker and Frank2002] Markus Becker and Anette Frank. 2002. A stochastic topological parser for German. In *Proc. 19th Int. Conf. on Computational Linguistics, Coling-2002*, Taipeh, Taiwan.

[Beil et al.2002] Franz Beil, Detlef Prescher, Helmut Schmid, and Sabine Schulte im Walde. 2002. Evaluation of the Gramotron parser for German. In *Proceedings of the LREC Workshop: Beyond PARSEVAL*, Las Palmas, Gran Canaria.

[Brants1999] Thorsten Brants. 1999. Cascaded markov models. In *Proc. 9th Conf. of the European Chapter of the ACL, EACL-1999*, Bergen, Norway.

[Braun2003] Christian Braun. 2003. Parsing German text for syntactico-semantic structures. In *Prospects and Advances in the Syntax/Semantics Interface, Proc. of the Lorraine-Saarland Workshop*, Nancy.

[Collins1999] Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing.* PhD thesis, University of Pennsylvania, Philadephia, PA.

[Daum and Menzel2002] Michael Daum and Wolfgang Menzel. 2002. Parsing natural language using guided local search. In *Proc. 15th European Conference on Artificial Intelligence, ECAI-2002*, pages 435–439, Lyon, France.

[Daum et al.2004] Micha Daum, Kilian Foth, and Wolfgang Menzel. 2004. Automatic transformation of phrase treebanks to dependency trees. In *Proc. 4th international Language Resources and Evaluation Conference*, Lisbon, Portugal.

[Dubey and Keller2003] Amit Dubey and Frank Keller. 2003. Probabilistic parsing for German using sister-head dependencies. In *Proc. 41st Annual Meeting of the Association of Computational Linguistics, ACL-2003*, Sapporo, Japan.

[Foth and Menzel2000] Kilian Foth and Wolfgang Menzel. 2000. A transformation-based parsing technique with anytime properties. In *Proc. International Workshop on Parsing Technologies (IWPT-2000)*, pages 89–100, Trento, Italy.

[Foth and Menzel2003] Kilian Foth and Wolfgang Menzel. 2003. Subtree parsing to speed up deep analysis. In *Proc. 8th Int. Workshop on Parsing Techniques*, pages 91–102, Nancy, France.

[Langer2001] Hagen Langer. 2001. *Parsing-Experimente: praxisorientierte Untersuchungen zur automatischen Analyse des Deutschen.* Peter Lang, Frankfurt am Main.

[Lin1995] Dekang Lin. 1995. A dependency-based method for evaluating broad-coverage parsers. In *IJCAI*, pages 1420–1427.

[Maruyama1990] Hiroshi Maruyama. 1990. Constraint dependency grammar. Technical Report RT0044, IBM Research, Tokyo Research Laboratory.

[Neumann and Flickinger2002] Günter Neumann and Dan Flickinger. 2002. HPSG-DOP: Data-oriented parsing with HPSG. In *Proc. of the 9th Int. Conf. on HPSG, HPSG-2002*, Seoul, South Korea.

[Schmid and Schulte im Walde2000] Helmut Schmid and Sabine Schulte im Walde. 2000. Robust German noun chunking with a probabilistic context-free grammar. In *Proc. 18th Int. Conf. on Computational Linguistics, Coling-2000*, Saarbrücken, Germany.

[Schröder2002] Ingo Schröder. 2002. *Natural Language Parsing with Graded Constraints*. PhD thesis, Hamburg University, Hamburg, Germany.

[Tapanainen and Järvinen1997] Pasi Tapanainen and Timo Järvinen. 1997. A non-projective dependency parser. In *Proc 5th Conference on Applied Natural Language Processing*, Washington, D.C.

[Wauschkuhn1995] Oliver Wauschkuhn. 1995. The influence of tagging on the results of partial parsing in German corpora. In *Proc. 4th Int. Workshop on Parsing Technologies, IWPT-1995*, pages 260–270, Prague/Karlovy Vary, Czech Republic.