Generating Constituent Order in German Clauses

Katja Filippova and **Michael Strube**
EML Research gGmbH
Schloss-Wolfsbrunnenweg 33
69118 Heidelberg, Germany
http://www.eml-research.de/nlp

Abstract

We investigate the factors which determine constituent order in German clauses and propose an algorithm which performs the task in two steps: First, the best candidate for the initial sentence position is chosen. Then, the order for the remaining constituents is determined. The first task is more difficult than the second one because of properties of the German sentence-initial position. Experiments show a significant improvement over competing approaches. Our algorithm is also more efficient than these.

1 Introduction

Many natural languages allow variation in the word order. This is a challenge for natural language generation and machine translation systems, or for text summarizers. E.g., in text-to-text generation (Barzilay & McKeown, 2005; Marsi & Krahmer, 2005; Wan et al., 2005), new sentences are fused from dependency structures of input sentences. The last step of sentence fusion is linearization of the resulting parse. Even for English, which is a language with fixed word order, this is not a trivial task.

German has a relatively free word order. This concerns the order of *constituents*[1] within sentences while the order of words within constituents is relatively rigid. The grammar only partially prescribes how constituents dependent on the verb should be ordered, and for many clauses each of the $n!$ possible permutations of n constituents is grammatical.

In spite of the permanent interest in German word order in the linguistics community, most studies have limited their scope to the order of verb arguments and few researchers have implemented – and even less evaluated – a generation algorithm. In this paper, we present an algorithm, which orders not only verb arguments but all kinds of constituents, and evaluate it on a corpus of biographies. For each parsed sentence in the test set, our maximum-entropy-based algorithm aims at reproducing the order found in the original text. We investigate the importance of different linguistic factors and suggest an algorithm to constituent ordering which first determines the sentence initial constituent and then orders the remaining ones. We provide evidence that the task requires language-specific knowledge to achieve better results and point to the most difficult part of it. Similar to Langkilde & Knight (1998) we utilize statistical methods. Unlike overgeneration approaches (Varges & Mellish, 2001, inter alia) which select the best of *all* possible outputs ours is more efficient, because we do not need to generate every permutation.

2 Theoretical Premises

2.1 Background

It has been suggested that several factors have an influence on German constituent order. Apart from the constraints posed by the grammar, information structure, surface form, and discourse status have also been shown to play a role. It has also been observed that there are preferences for a particular order. The preferences summarized below have mo-

[1]Henceforth, we will use this term to refer to constituents dependent on the clausal top node, i.e. a verb, only.

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 320–327,
Prague, Czech Republic, June 2007.

tivated our choice of features:

- constituents in the nominative case precede those in other cases, and dative constituents often precede those in the accusative case (Uszkoreit, 1987; Keller, 2000);
- the verb arguments' order depends on the verb's subcategorization properties (Kurz, 2000);
- constituents with a definite article precede those with an indefinite one (Weber & Müller, 2004);
- pronominalized constituents precede non-pronominalized ones (Kempen & Harbusch, 2004);
- animate referents precede inanimate ones (Pappert et al., 2007);
- short constituents precede longer ones (Kimball, 1973);
- the preferred topic position is right after the verb (Frey, 2004);
- the initial position is usually occupied by scene-setting elements and topics (Speyer, 2005).
- there is a default order based on semantic properties of constituents (Sgall et al., 1986):
  Actor $<$ Temporal $<$ SpaceLocative $<$ Means $<$ Addressee $<$ Patient $<$ Source $<$ Destination $<$ Purpose

Note that most of these preferences were identified in corpus studies and experiments with native speakers and concern the order of verb arguments only. Little has been said so far about how non-arguments should be ordered.

German is a verb second language, i.e., the position of the verb in the main clause is determined exclusively by the grammar and is insensitive to other factors. Thus, the German main clause is divided into two parts by the finite verb: *Vorfeld (VF)*, which contains exactly one constituent, and *Mittelfeld (MF)*, where the remaining constituents are located. The subordinate clause normally has only MF. The VF and MF are marked with brackets in Example 1:

(1) [Außerdem] entwickelte [Lummer eine
Apart from that developed Lummer a
Quecksilberdampflampe, um
Mercury-vapor lamp to
monochromatisches Licht herzustellen].
monochrome light produce.

'Apart from that, Lummer developed a Mercury-vapor lamp to produce monochrome light'.

2.2 Our Hypothesis

The essential contribution of our study is that we treat preverbal and postverbal parts of the sentence differently. The sentence-initial position, which in German is the VF, has been shown to be cognitively more prominent than other positions (Gernsbacher & Hargreaves, 1988). Motivated by the theoretical work by Chafe (1976) and Jacobs (2001), we view the VF as the place for elements which modify the situation described in the sentence, i.e. for so called frame-setting topics (Jacobs, 2001). For example, temporal or locational constituents, or anaphoric adverbs are good candidates for the VF. We hypothesize that the reasons which bring a constituent to the VF are different from those which place it, say, to the beginning of the MF, for the order in the MF has been shown to be relatively rigid (Keller, 2000; Kempen & Harbusch, 2004). Speakers have the freedom of selecting the outgoing point for a sentence. Once they have selected it, the remaining constituents are arranged in the MF, mainly according to their grammatical properties.

This last observation motivates another hypothesis we make: The cumulation of the properties of a constituent determines its *salience*. This salience can be calculated and used for ordering with a simple rule stating that more salient constituents should precede less salient ones. In this case there is no need to generate all possible orders and rank them. The best order can be obtained from a random one by sorting. Our experiments support this view. A two-step approach, which first selects the best candidate for the VF and then arranges the remaining constituents in the MF with respect to their salience performs better than algorithms which generate the order for a sentence as a whole.

3 Related Work

Uszkoreit (1987) addresses the problem from a mostly grammar-based perspective and suggests weighted constraints, such as [+NOM] $\prec$ [+DAT], [+PRO] $\prec$ [–PRO], [–FOCUS] $\prec$ [+FOCUS], etc.

Kruijff et al. (2001) describe an architecture which supports generating the appropriate word order for different languages. Inspired by the findings of the Prague School (Sgall et al., 1986) and Systemic Functional Linguistics (Halliday, 1985), they focus on the role that information structure plays in constituent ordering. Kruijff-Korbayová et al. (2002) address the task of word order generation in the same vein. Similar to ours, their algorithm recognizes the special role of the sentence-initial position which they reserve for the *theme* – the point of departure of the message. Unfortunately, they did not implement their algorithm, and it is hard to judge how well the system would perform on real data.

Harbusch et al. (2006) present a generation workbench, which has the goal of producing not the most appropriate order, but all grammatical ones. They also do not provide experimental results.

The work of Uchimoto et al. (2000) is done on the free word order language Japanese. They determine the order of phrasal units dependent on the same modifiee. Their approach is similar to ours in that they aim at regenerating the original order from a dependency parse, but differs in the scope of the problem as they regenerate the order of modifers for all and not only for the top clausal node. Using a maximum entropy framework, they choose the most probable order from the set of all permutations of n words by the following formula:

$$\begin{aligned} P(1|h) &= P(\{W_{i,i+j} = 1 | 1 \le i \le n-1, 1 \le j \le n-i\}|h) \\ &\approx \prod_{i=1}^{n-1} \prod_{j=1}^{n-i} P(W_{i,i+j} = 1 | h_{i,i+j}) \\ &= \prod_{i=1}^{n-1} \prod_{j=1}^{n-i} P_{ME}(1 | h_{i,i+j}) \end{aligned} \tag{1}$$

For each permutation, for every pair of words , they multiply the probability of their being in the correct[2] order given the history h. Random variable $W_{i,i+j}$ is 1 if word w_i precedes w_{i+j} in the reference sentence, 0 otherwise. The features they use are akin to those which play a role in determining German word order. We use their approach as a non-trivial baseline in our study.

Ringger et al. (2004) aim at regenerating the order of constituents as well as the order within them for German and French technical manuals. Utilizing syntactic, semantic, sub-categorization and length features, they test several statistical models to find the order which maximizes the probability of an ordered tree. Using "Markov grammars" as the starting point and conditioning on the syntactic category only, they expand a non-terminal node C by predicting its daughters from left to right:

$$P(C|h) = \prod_{i=1}^{n} P(d_i | d_{i-1}, ..., d_{i-j}, c, h) \tag{2}$$

Here, c is the syntactic category of C, d and h are the syntactic categories of C's daughters and the daughter which is the head of C respectively.

In their simplest system, whose performance is only 2.5% worse than the performance of the best one, they condition on both syntactic categories and semantic relations (ψ) according to the formula:

$$P(C|h) = \prod_{i=1}^{n} \left[\begin{array}{l} P(\psi_i | d_{i-1}, \psi_{i-1}, ...d_{i-j}, \psi_{i-j}, c, h) \\ \times P(d_i | \psi_i, d_{i-1}, \psi_{i-1} ..., d_{i-j}, \psi_{i-j}, c, h) \end{array} \right] \tag{3}$$

Although they test their system on German data, it is hard to compare their results to ours directly. First, the metric they use does not describe the performance appropriately (see Section 6.1). Second, while the word order within NPs and PPs as well as the verb position are prescribed by the grammar to a large extent, the constituents can theoretically be ordered in any way. Thus, by generating the order for every non-terminal node, they combine two tasks of different complexity and mix the results of the more difficult task with those of the easier one.

4 Data

The data we work with is a collection of biographies from the German version of Wikipedia[3]. Fully automatic preprocessing in our system comprises the following steps: First, a list of people of a certain Wikipedia category is taken and an article is extracted for every person. Second, sentence

[2]Only reference orders are assumed to be correct.

[3]http://de.wikipedia.org

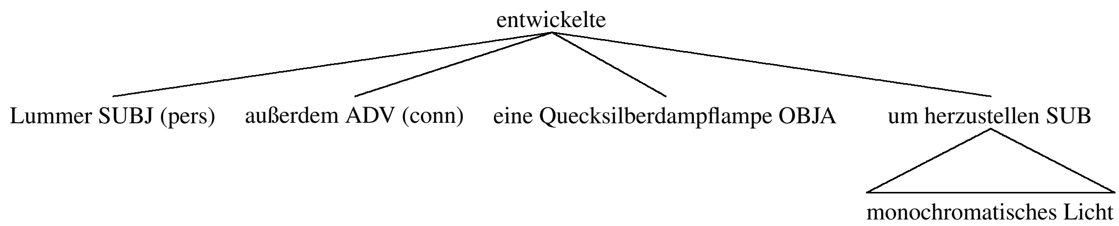

Figure 1: The representation of the sentence in Example 1

boundaries are identified with a Perl CPAN module[4] whose performance we improved by extending the list of abbreviations. Next, the sentences are split into tokens. The TnT tagger (Brants, 2000) and the TreeTagger (Schmid, 1997) are used for tagging and lemmatization. Finally, the articles are parsed with the CDG dependency parser (Foth & Menzel, 2006). Named entities are classified according to their semantic type using lists and category information from Wikipedia: person (*pers*), location (*loc*), organization (*org*), or undefined named entity (*undef_ne*). Temporal expressions (*Oktober 1915*, *danach (after that)* etc.) are identified automatically by a set of patterns. Inevitable during automatic annotation, errors at one of the preprocessing stages cause errors at the ordering stage.

Distinguishing between main and subordinate clauses, we split the total of about 19 000 sentences into training, development and test sets (Table 1). Clauses with one constituent are sorted out as trivial. The distribution of both types of clauses according to their length in constituents is given in Table 2.

	train	dev	test
main	14324	3344	1683
sub	3304	777	408
total	17628	4121	2091

Table 1: Size of the data sets in clauses

	2	3	4	5	6+
main	20%	35%	27%	12%	6%
sub	49%	35%	11%	2%	3%

Table 2: Proportion of clauses with certain lengths

[4]http://search.cpan.org/~holsten/Lingua-DE-Sentence-0.07/Sentence.pm

Given the sentence in Example 1, we first transform its dependency parse into a more general representation (Figure 1[5]) and then, based on the predictions of our learner, arrange the four constituents. For evaluation, we compare the arranged order against the original one.

Note that we predict neither the position of the verb, nor the order within constituents as the former is explicitly determined by the grammar, and the latter is much more rigid than the order of constituents.

5 Baselines and Algorithms

We compare the performance of two our algorithms with four baselines.

5.1 Random

We improve a trivial random baseline (RAND) by two syntax-oriented rules: the first position is reserved for the subject and the second for the direct object if there is any; the order of the remaining constituents is generated randomly (RAND_IMP).

5.2 Statistical Bigram Model

Similar to Ringger et al. (2004), we find the order with the highest probability conditioned on syntactic and semantic categories. Unlike them we use dependency parses and compute the probability of the top node only, which is modified by all constituents. With these adjustments the probability of an order O given the history h, if conditioned on syntactic functions of constituents ($s_1...s_n$), is simply:

$$P(O|h) = \prod_{i=1}^{n} P(s_i|s_{i-1}, h) \qquad (4)$$

Ringger et al. (2004) do not make explicit, what their set of semantic relations consists of. From the

[5]OBJA stands for the accusative object.

example in the paper, it seems that these are a mixture of lexical and syntactic information[6]. Our annotation does not specify semantic relations. Instead, some of the constituents are categorized as *pers, loc, temp, org* or *undef_ne* if their heads bear one of these labels. By joining these with possible syntactic functions, we obtain a larger set of syntactic-semantic tags as, e.g., *subj-pers, pp-loc, adv-temp*. We transform each clause in the training set into a sequence of such tags, plus three tags for the verb position (*v*), the beginning (*b*) and the end (*e*) of the clause. Then we compute the bigram probabilities[7].

For our third baseline (BIGRAM), we select from all possible orders the one with the highest probability as calculated by the following formula:

$$P(O|h) = \prod_{i=1}^{n} P(t_i|t_{i-1}, h) \qquad (5)$$

where t_i is from the set of joined tags. For Example 1, possible tag sequences (i.e. orders) are *'b subj-pers v adv obja sub e', 'b adv v subj-pers obja sub e', 'b obja v adv sub subj-pers e'*, etc.

5.3 Uchimoto

For the fourth baseline (UCHIMOTO), we utilized a maximum entropy learner (OpenNLP[8]) and reimplemented the algorithm of Uchimoto et al. (2000). For every possible permutation, its probability is estimated according to Formula (1). The binary classifier, whose task was to predict the probability that the order of a pair of constituents is correct, was trained on the following features describing the verb or h_c – the head of a constituent *c*[9]:

vlex, vpass, vmod the lemma of the root of the clause (non-auxiliary verb), the voice of the verb and the number of constituents to order;

lex the lemma of h_c or, if h_c is a functional word, the lemma of the word which depends on it;

pos part-of-speech tag of h_c;

sem if defined, the semantic class of *c*; e.g. *im April 1900* and *mit Albert Einstein (with Albert Einstein)* are classified *temp* and *pers* respectively;

syn, same the syntactic function of h_c and whether it is the same for the two constituents;

mod number of modifiers of h_c;

rep whether h_c appears in the preceding sentence;

pro whether *c* contains a (anaphoric) pronoun.

5.4 Maximum Entropy

The first configuration of our system is an extended version of the UCHIMOTO baseline (MAXENT). To the features describing c we added the following ones:

det the kind of determiner modifying h_c (*def, indef, non-appl*);

rel whether h_c is modified by a relative clause (*yes, no, non-appl*);

dep the depth of *c*;

len the length of *c* in words.

The first two features describe the discourse status of a constituent; the other two provide information on its "weight". Since our learner treats all values as nominal, we discretized the values of **dep** and **len** with a C4.5 classifier (Kohavi & Sahami, 1996).

Another modification concerns the efficiency of the algorithm. Instead of calculating probabilities for all pairs, we obtain the right order from a random one by *sorting*. We compare adjacent elements by consulting the learner as if we would sort an array of numbers. Given two adjacent constituents, $c_i < c_j$, we check the probability of their being in the right order, i.e. that c_i precedes c_j: $P_{pre}(c_i, c_j)$. If it is less than 0.5, we transpose the two and compare c_i with the next one.

Since the sorting method presupposes that the predicted relation is transitive, we checked whether this is really so on the development and test data sets. We looked for three constituents c_i, c_j, c_k from a sentence S, such that $P_{pre}(c_i, c_j) > 0.5, P_{pre}(c_j, c_k) > 0.5, P_{pre}(c_i, c_k) < 0.5$ and found none. Therefore, unlike UCHIMOTO, where one needs to make exactly $N! * N(N-1)/2$ comparisons, we have to make $N(N-1)/2$ comparisons at most.

[6]E.g. *DefDet, Coords, Possr, werden*

[7]We use the CMU Toolkit (Clarkson & Rosenfeld, 1997).

[8]http://opennlp.sourceforge.net

[9]We disregarded features which use information specific to Japanese and non-applicable to German (e.g. on postpositional particles).

5.5 The Two-Step Approach

The main difference between our first algorithm (MAXENT) and the second one (TWO-STEP) is that we generate the order in two steps[10] (both classifiers are trained on the same features):

1. For the VF, using the OpenNLP maximum entropy learner for a binary classification (VF vs. MF), we select the constituent c with the highest probability of being in the VF.

2. For the MF, the remaining constituents are put into a random order and then *sorted* the way it is done for MAXENT. The training data for the second task was generated only from the MF of clauses.

6 Results

6.1 Evaluation Metrics

We use several metrics to evaluate our systems and the baselines. The first is per-sentence *accuracy* (*acc*) which is the proportion of correctly regenerated sentences. Kendall's τ, which has been used for evaluating sentence ordering tasks (Lapata, 2006), is the second metric we use. τ is calculated as $1 - 4\frac{t}{N(N-1)}$, where t is the number of interchanges of consecutive elements to arrange N elements in the right order. τ is sensitive to near misses and assigns *abdc* (almost correct order) a score of 0.66 while *dcba* (inverse order) gets -1. Note that it is questionable whether this metric is as appropriate for word ordering tasks as for sentence ordering ones because a near miss might turn out to be ungrammatical whereas a more different order stays acceptable.

Apart from *acc* and τ, we also adopt the metrics used by Uchimoto et al. (2000) and Ringger et al. (2004). The former use *agreement rate* (*agr*) calculated as $\frac{2p}{N(N-1)}$: the number of correctly ordered pairs of constituents over the total number of all possible pairs, as well as *complete agreement* which is basically per-sentence accuracy. Unlike τ, which has -1 as the lowest score, *agr* ranges from 0 to 1. Ringger et al. (2004) evaluate the performance only in terms of *per-constituent edit distance* calculated as $\frac{m}{N}$, where m is the minimum number of moves[11] needed to arrange N constituents in the right order. This measure seems less appropriate than τ or *agr* because it does not take the distance of the move into account and scores *abced* and *eabcd* equally (0.2).

Since τ and *agr*, unlike *edit distance*, give higher scores to better orders, we compute *inverse distance*: *inv = 1 – edit_distance* instead. Thus, all three metrics (τ, *agr, inv*) give the maximum of 1 if constituents are ordered correctly. However, like τ, *agr* and *inv* can give a positive score to an ungrammatical order. Hence, none of the evaluation metrics describes the performance perfectly. Human evaluation which reliably distinguishes between appropriate, acceptable, grammatical and ingrammatical orders was out of choice because of its high cost.

6.2 Results

The results on the test data are presented in Table 3. The performance of TWO-STEP is significantly better than any other method (χ^2, $p < 0.01$). The performance of MAXENT does not significantly differ from UCHIMOTO. BIGRAM performed about as good as UCHIMOTO and MAXENT. We also checked how well TWO-STEP performs on each of the two sub-tasks (Table 4) and found that the VF selection is considerably more difficult than the sorting part.

	acc	τ	agr	inv
RAND	15%	0.02	0.51	0.64
RAND_IMP	23%	0.24	0.62	0.71
BIGRAM	51%	0.60	0.80	0.83
UCHIMOTO	50%	0.65	0.82	0.83
MAXENT	52%	0.67	0.84	0.84
TWO-STEP	61%	0.72	0.86	0.87

Table 3: Per-clause mean of the results

The most important conclusion we draw from the results is that the gain of 9% accuracy is due to the VF selection only, because the feature sets are identical for MAXENT and TWO-STEP. From this follows that doing feature selection without splitting the task in two is ineffective, because the importance of a feature depends on whether the VF or the MF is considered. For the MF, feature selection has shown **syn** and **pos** to be the most relevant features. They alone bring the performance in the MF up to 75%. In contrast, these two features explain only 56% of the

[10] Since subordinate clauses do not have a VF, the first step is not needed.

[11] A move is a deletion combined with an insertion.

cases in the VF. This implies that the order in the MF mainly depends on grammatical features, while for the VF all features are important because removal of any feature caused a loss in accuracy.

	acc	τ	agr	inv
TWO-STEP VF	68%	-	-	-
TWO-STEP MF	80%	0.92	0.96	0.95

Table 4: Mean of the results for the VF and the MF

Another important finding is that there is no need to overgenerate to find the right order. Insignificant for clauses with two or three constituents, for clauses with 10 constituents, the number of comparisons is reduced drastically from 163,296,000 to 45.

According to the *inv* metric, our results are considerably worse than those reported by Ringger et al. (2004). As mentioned in Section 3, the fact that they generate the order for every non-terminal node seriously inflates their numbers. Apart from that, they do not report accuracy, and it is unknown, how many sentences they actually reproduced correctly.

6.3 Error Analysis

To reveal the main error sources, we analyzed incorrect predictions concerning the VF and the MF, one hundred for each. Most errors in the VF did not lead to unacceptability or ungrammaticality. From lexical and semantic features, the classifier learned that some expressions are often used in the beginning of a sentence. These are temporal or locational PPs, anaphoric adverbials, some connectives or phrases starting with *unlike X, together with X, as X*, etc. Such elements were placed in the VF instead of the subject and caused an error although both variants were equally acceptable. In other cases the classifier could not find a better candidate but the subject because it could not conclude from the provided features that another constituent would nicely introduce the sentence into the discourse. Mainly this concerns recognizing information familiar to the reader not by an already mentioned entity, but one which is inferrable from what has been read.

In the MF, many orders had a PP transposed with the direct object. In some cases the predicted order seemed as good as the correct one. Often the algorithm failed at identifying verb-specific preferences: E.g., some verbs take PPs with the locational meaning as an argument and normally have them right next to them, whereas others do not. Another frequent error was the wrong placement of superficially identical constituents, e.g. two PPs of the same size. To handle this error, the system needs more specific semantic information. Some errors were caused by the parser, which created extra constituents (e.g. false PP or adverb attachment) or confused the subject with the direct verb.

We retrained our system on a corpus of newspaper articles (Telljohann et al., 2003, TüBa-D/Z) which is manually annotated but encodes no semantic knowledge. The results for the MF were the same as on the data from Wikipedia. The results for the VF were much worse (45%) because of the lack of semantic information.

7 Conclusion

We presented a novel approach to ordering constituents in German. The results indicate that a linguistically-motivated two-step system, which first selects a constituent for the initial position and then orders the remaining ones, works significantly better than approaches which do not make this separation. Our results also confirm the hypothesis – which has been attested in several corpus studies – that the order in the MF is rather rigid and dependent on grammatical properties.

We have also demonstrated that there is no need to overgenerate to find the best order. On a practical side, this finding reduces the amount of work considerably. Theoretically, it lets us conclude that the relatively fixed order in the MF depends on the salience which can be predicted mainly from grammatical features. It is much harder to predict which element should be placed in the VF. We suppose that this difficulty comes from the double function of the initial position which can either introduce the addressation topic, or be the scene- or frame-setting position (Jacobs, 2001).

Acknowledgements: This work has been funded by the Klaus Tschira Foundation, Heidelberg, Germany. The first author has been supported by a KTF grant (09.009.2004). We would also like to thank Elke Teich and the three anonymous reviewers for their useful comments.

References

Barzilay, R. & K. R. McKeown (2005). Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–327.

Brants, T. (2000). TnT – A statistical Part-of-Speech tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing,* Seattle, Wash., 29 April – 4 May 2000, pp. 224–231.

Chafe, W. (1976). Givenness, contrastiveness, definiteness, subjects, topics, and point of view. In C. Li (Ed.), *Subject and Topic*, pp. 25–55. New York, N.Y.: Academic Press.

Clarkson, P. & R. Rosenfeld (1997). Statistical language modeling using the CMU-Cambridge toolkit. In *Proceedings of the 5th European Conference on Speech Communication and Technology,* Rhodes, Greece, 22-25 September 1997, pp. 2707–2710.

Foth, K. & W. Menzel (2006). Hybrid parsing: Using probabilistic models as predictors for a symbolic parser. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics,* Sydney, Australia, 17–21 July 2006, pp. 321–327.

Frey, W. (2004). A medial topic position for German. *Linguistische Berichte*, 198:153–190.

Gernsbacher, M. A. & D. J. Hargreaves (1988). Accessing sentence participants: The advantage of first mention. *Journal of Memory and Language*, 27:699–717.

Halliday, M. A. K. (1985). *Introduction to Functional Grammar*. London, UK: Arnold.

Harbusch, K., G. Kempen, C. van Breugel & U. Koch (2006). A generation-oriented workbench for performance grammar: Capturing linear order variability in German and Dutch. In *Proceedings of the International Workshop on Natural Language Generation,* Sydney, Australia, 15-16 July 2006, pp. 9–11.

Jacobs, J. (2001). The dimensions of topic-comment. *Linguistics*, 39(4):641–681.

Keller, F. (2000). *Gradience in Grammar: Experimental and Computational Aspects of Degrees of Grammaticality*, (Ph.D. thesis). University of Edinburgh.

Kempen, G. & K. Harbusch (2004). How flexible is constituent order in the midfield of German subordinate clauses? A corpus study revealing unexpected rigidity. In *Proceedings of the International Conference on Linguistic Evidence,* Tübingen, Germany, 29–31 January 2004, pp. 81–85.

Kimball, J. (1973). Seven principles of surface structure parsing in natural language. *Cognition*, 2:15–47.

Kohavi, R. & M. Sahami (1996). Error-based and entropy-based discretization of continuous features. In *Proceedings of the 2nd International Conference on Data Mining and Knowledge Discovery,* Portland, Oreg., 2–4 August, 1996, pp. 114–119.

Kruijff, G.-J., I. Kruijff-Korbayová, J. Bateman & E. Teich (2001). Linear order as higher-level decision: Information structure in strategic and tactical generation. In *Proceedings of the 8th European Workshop on Natural Language Generation,* Toulouse, France, 6-7 July 2001, pp. 74–83.

Kruijff-Korbayová, I., G.-J. Kruijff & J. Bateman (2002). Generation of appropriate word order. In K. van Deemter & R. Kibble (Eds.), *Information Sharing: Reference and Presupposition in Language Generation and Interpretation*, pp. 193–222. Stanford, Cal.: CSLI.

Kurz, D. (2000). A statistical account on word order variation in German. In A. Abeillé, T. Brants & H. Uszkoreit (Eds.), *Proceedings of the COLING Workshop on Linguistically Interpreted Corpora,* Luxembourg, 6 August 2000.

Langkilde, I. & K. Knight (1998). Generation that exploits corpus-based statistical knowledge. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics,* Montréal, Québec, Canada, 10–14 August 1998, pp. 704–710.

Lapata, M. (2006). Automatic evaluation of information ordering: Kendall's tau. *Computational Linguistics*, 32(4):471–484.

Marsi, E. & E. Krahmer (2005). Explorations in sentence fusion. In *Proceedings of the European Workshop on Natural Language Generation,* Aberdeen, Scotland, 8–10 August, 2005, pp. 109–117.

Pappert, S., J. Schliesser, D. P. Janssen & T. Pechmann (2007). Corpus- and psycholinguistic investigations of linguistic constraints on German word order. In A. Steube (Ed.), *The discourse potential of underspecified structures: Event structures and information structures*. Berlin, New York: Mouton de Gruyter. In press.

Ringger, E., M. Gamon, R. C. Moore, D. Rojas, M. Smets & S. Corston-Oliver (2004). Linguistically informed statistical models of constituent structure for ordering in sentence realization. In *Proceedings of the 20th International Conference on Computational Linguistics,* Geneva, Switzerland, 23–27 August 2004, pp. 673–679.

Schmid, H. (1997). Probabilistic Part-of-Speech tagging using decision trees. In D. Jones & H. Somers (Eds.), *New Methods in Language Processing*, pp. 154–164. London, UK: UCL Press.

Sgall, P., E. Hajičová & J. Panevová (1986). *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. Dordrecht, The Netherlands: D. Reidel.

Speyer, A. (2005). Competing constraints on Vorfeldbesetzung in German. In *Proceedings of the Constraints in Discourse Workshop,* Dortmund, 3–5 July 2005, pp. 79–87.

Telljohann, H., E. W. Hinrichs & S. Kübler (2003). *Stylebook for the Tübingen treebank of written German (TüBa-D/Z.* Technical Report: Seminar für Sprachwissenschaft, Universität Tübingen, Tübingen, Germany.

Uchimoto, K., M. Murata, Q. Ma, S. Sekine & H. Isahara (2000). Word order acquisition from corpora. In *Proceedings of the 18th International Conference on Computational Linguistics,* Saarbrücken, Germany, 31 July – 4 August 2000, pp. 871–877.

Uszkoreit, H. (1987). *Word Order and Constituent Structure in German*. CSLI Lecture Notes. Stanford: CSLI.

Varges, S. & C. Mellish (2001). Instance-based natural language generation. In *Proceedings of the 2nd Conference of the North American Chapter of the Association for Computational Linguistics,* Pittsburgh, Penn., 2–7 June, 2001, pp. 1–8.

Wan, S., R. Dale, M. Dras & C. Paris (2005). Searching for grammaticality and consistency: Propagating dependencies in the Viterbi algorithm. In *Proceedings of the 10th European Workshop on Natural Language Generation,* Aberdeen, Scotland, *8–10 August, 2005*, pp. 211–216.

Weber, A. & K. Müller (2004). Word order variation in German main clauses: A corpus analysis. In *Proceedings of the 5th International Workshop on Linguistically Interpreted Corpora,* 29 August, 2004, Geneva, Switzerland, pp. 71–77.