Gefördert vom
bmb+f
Bundesministerium für
Bildung, Wissenschaft,
Forschung und Technologie

Verbmobil
Verbundvorhaben

Improving Parsing of Spontaneous Speech with the Help of Prosodic Boundaries

R. Kompe[1], A. Kießling[1], H. Niemann[1], E. Nöth[1], A. Batliner[1], S. Schacht[2], T. Ruland[2], H. U. Block[2]

[1]F.–A.–Universität Erlangen–Nürnberg
[2]Siemens AG, München

Vm
Report 210
August 1997

August 1997

R. Kompe[1], A. Kießling[1], H. Niemann[1], E. Nöth[1], A. Batliner[1],
S. Schacht[2], T. Ruland[2], H. U. Block[2]

Lehrstuhl für Mustererkennung (Inf. 5)
Friedrich–Alexander–Universität Erlangen–Nürnberg
Martensstr. 3
D–91058 Erlangen

Tel.: (09131) 85 - 7883
e-mail: kompe@informatik.uni-erlangen.de

Gehört zum Antragsabschnitt: 1.7

Das diesem Bericht zugrundeliegende Forschungsvorhaben wurde mit Mitteln des Bundesministers für Bildung, Wissenschaft, Forschung und Technologie unter dem Förderkennzeichen 01 IV 701 K/5 gefördert. Die Verantwortung für den Inhalt dieser Arbeit liegt bei den Autoren.

IMPROVING PARSING OF SPONTANEOUS SPEECH WITH THE HELP OF PROSODIC BOUNDARIES

R. Kompe[1] *A. Kießling*[1] *H. Niemann*[1] *E. Nöth*[1] *A. Batliner*[2] *S. Schachtl*[3] *T. Ruland*[3] *H. U. Block*[3]

[1]Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, Martensstr. 3, 91058 Erlangen, Germany
[2] Institut für Deutsche Philologie, L.–M. Universität München, Schellingstr. 3, 80799 München, Germany
[3] Siemens AG, ZT IK 5, Otto-Hahn-Ring 6, 81730 München, Germany
e–mail: noeth@informatik.uni-erlangen.de www: http://www5.informatik.uni-erlangen.de/

ABSTRACT

Parsing can be improved in automatic speech understanding if prosodic boundary marking is taken into account, because syntactic boundaries are often marked by prosodic means. Because large databases are needed for the training of statistical models for prosodic boundaries, we developed a labeling scheme for syntactic–prosodic boundaries within the German VERBMOBIL project (automatic speech–to–speech translation). We compare the results of classifiers (multi–layer perceptrons and language models) trained on these syntactic–prosodic boundary labels with classifiers trained on perceptual–prosodic and purely syntactic labels. Recognition rates of up to 96% were achieved. The turns that we need to parse consist of 20 words on the average and frequently contain sequences of partial sentence equivalents due to restarts, ellipsis, etc. For this material, the boundary scores computed by our classifiers can successfully be integrated into the syntactic parsing of word graphs; currently, they improve the parse time by 92% and reduce the number of parse trees by 96%. This is achieved by introducing a special **P**rosodic **S**yntactic **C**lause **B**oundary symbol (PSCB) into our grammar and guiding the search for the best word chain with the prosodic boundary scores.

1. INTRODUCTION

Prosody structures utterances and helps the listeners to understand and disambiguate their meaning. To our knowledge, however, so far nobody has really integrated this information into a complete automatic speech understanding system. We will present a syntactic analysis of word hypotheses graphs using prosodic clause boundary information. Our research is carried out in the speech–to–speech translation project VERBMOBIL [19, 6] (domain: appointment scheduling) where the influence of prosody can already be evaluated in an end–to–end system; for the integration of prosody in the VERBMOBIL system, cf. [12], for the linguistic processing of VERBMOBIL, cf. [4].

A corpus analysis of VERBMOBIL data (human–human dialogs) showed that about 70 % of the utterances contain more than one single sentence [18]. About 25 % of the utterances are longer than 10 seconds. Especially for such a material, the use of prosody in parsing is crucial for two reasons:

First, to ensure that most of the words that were spoken are recognized, a large word hypotheses graph (currently about 10 hypotheses per spoken word) has to be generated. Finding the correct (or approximately correct) path through a word hypotheses graph is thus an enormous search problem.

Second, spontaneous speech contains many elliptic constructions. So even if the spoken word sequence has been recovered by word recognition correctly, there still might be many different parses possible, especially with longer turns. Consider the following two of the at least 36 different syntactic readings for a word sequence taken from the VERBMOBIL corpus

"*Ja zur Not. Geht's auch am Samstag?*"
vs. "*Ja zur Not geht's auch am Samstag.*"

The appropriate English translations are

"*O.K., if necessary. Is Saturday possible as well?*"
vs. "*Well, if necessary, Saturday is possible as well.*"

In these examples, only the prosodically marked boundaries can disambiguate between the two different semantic meanings and pragmatic interpretations.

We use prosody only to guide the search for the best syntactic parse through the word graph; no hard decisions are made. Partial parses are ranked in an agenda according to a score which takes into account the prosodic probability for a clause boundary. At each step of the search the best partial parse is extended. So the main use of prosodic information will be to speed–up the search for the best complete parse. However, in a system with limited resources (i.e. the syntax has to produce a parse after $n\times$turn length or it will receive a time out signal), this speed–up will also increase the recognition rate of the syntax module.

2. PROSODIC SYNTACTIC BOUNDARY MARKERS — THE M–LABEL SYSTEM

We developed a syntactic–prosodic labeling scheme for German that provides a coarse labeling of syntactic boundaries. It can be done fast and fairly reliable because it is based solely on the transliteration of the turn; i.e., we do not have to listen to the turns. Prosodic knowledge is used, i.e., syntactic boundaries are marked differently depending on whether they are likely to be marked prosodically. Typical spontaneous speech phenomena are taken into account as well. Currently we distinguish 10 labels which are grouped into three major classes:

- M3: clause boundary (between main clauses, subordinate clauses, elliptic clauses, etc.)
- M0: no clause boundary
- MU: undefined, i.e. M3 or M0 cannot be assigned to this word boundary without context knowledge and/or perceptual analysis.

The labeling scheme is described in more detail in [2, 3]. In [2] we compared these labels with purely prosodic labels (B–labels)[2] [14], and precise syntactic labels (S–labels) [7].

[1]This work was partly funded by the German Federal Ministry of Education, Science, Research and Technology (BMBF) in the framework of the VERBMOBIL Project under Grant 01 IV 101 AO and funded under Grants 01 IV 102 F/4 and 01 IV 102 H/0. The responsibility for the contents lies with the authors.

[2]In the following we use B3 for a word boundary, which is perceived as a major prosodic boundary.

© IEEE 1997

cases	B3 vs. ¬B3 165 vs. 1284	M3 vs. M0 177 vs. 1169	MB3 vs. MB0 190 vs. 1259
MLP	87/87	87/83	85/82
LM	92/85	95/86	92/84
MLP+LM	94/89	96/89	94/88

Table 1. Percentage of correct classified word boundaries for different combinations of classifiers: total vs. class–wise average

This comparison showed that there is a high agreement between these labels and, hence, justifies our rather coarse labeling scheme. The advantage of the M–labels is that a high number of labeled data can be produced within a short time, because they do not require a complete syntactic analysis and they do not rely on perceptual evaluation. Meanwhile, there are 7,286 turns (about 150,000 words) labeled with the Ms, which took only a few months.

3. SPEECH DATABASE

For the classification experiments in Section 4, we used 3 dialogs of the VERBMOBIL database for testing (64 turns of 3 male and 3 female speakers, 1513 words, 12 minutes in total). For the training of the multi–layer perceptron (MLP) all the available data labeled with the B–labels were used (797 turns) except for the test set; for the language model (LM), trained with the M labels, 6297 turns were used. For the parsing experiments in Section 5 we chose 594 turns out of 122 dialogs. These turns had been selected for evaluation purposes by the DFKI (Saarbrücken), which was responsible for the integration of the VERBMOBIL demonstrator. For all of these turns, word graphs were provided by DFKI using the word recognizer of the University of Karlsruhe[3]. The word graphs contained 9.3 hypotheses per spoken word. The word accuracy, i.e., the highest accuracy of any of the paths contained in the graph, was 73.3%. 117 word graphs were correct, i.e. they contained the spoken word chain.

4. AUTOMATIC BOUNDARY CLASSIFICATION

We will now compare classification results obtained with a multi–layer perceptron (MLP), a stochastic (n-gram) language model (LM), and a combination of both classifiers. The MLP serves as an acoustic–prosodic classifier getting acoustic and few lexical features as its input. The LM estimates probabilities for boundaries given a few words in the context of the word. With these classifiers for each of the words in a word chain or in a word graph a probability for a clause boundary being after the word is computed.

The computation of the acoustic–prosodic features is based on an automatic time alignment of the phoneme sequence corresponding to the spoken or recognized words. For the boundary classification experiments we only use the aligned spoken words thus simulating 100% word recognition. For each word a vector of prosodic features is computed automatically from the speech signal. The feature set is described in [3] and, in more detail, in [9]. In order to balance for the a priori probabilities of the different classes, during training the MLP was presented with an equal number of feature vectors from each class. For the experiments, MLPs with 40/20 nodes in the first/second hidden layer showed best results. During training B3 vs. ¬B3 was taken as reference.

Trigram language models (LM) were additionally used for the classification of boundaries. They model partial word chains where M3 and M0 boundaries have been inserted. This method as well as the combination of LM and MLP scores is described in more detail in [11, 10].

In Table 1, we compare the results of different classifiers for the two main classes boundary vs. not–boundary using two different types of reference boundaries: B, M, and MB, which is a combination of both. In the case of M3 vs. M0, the 'undefined' boundaries MU are not taken into account. As for MB, MB3 represents all word boundaries which are either labeled with M3 or with MU and B3; MB0 refers to all other word boundaries. These combined labels represent best what the syntax would like to get delivered by the prosody. The first number in each row of the table shows the overall recognition rate, the second is the average of the class–wise recognition rates. The recognition rates take all word boundaries except the end of turns into account; the latter can be classified in a trivial way. It can be noticed that, roughly, the results get better from top left to bottom right. Best results can be achieved with a combination of the MLP with the LM no matter whether the perceptual B or the syntactic–prosodic M labels serve as reference. The LM alone is already very good; we have, however, to consider that it cannot be applied to the 'undefined' classes MU, which are of course very important for a correct syntactic/semantic processing and which account for about 4% of all word boundaries and for 23% of all non–M0 boundaries. Especially for these cases, we need a classifier trained with perceptual–prosodic labels. Note, however, that even on the M3/M0–task the combination of the two classifiers, MLP+LM, shows slightly better results than the LM alone.

Due to the different a priori probabilities, the boundaries are recognized worse than the non–boundaries with the LMs (e.g., 80.8% for M3 vs. 97.7% for M0 for the MLP+LM classifier); this causes the lower average of the class–wise recognition rates compared to the overall recognition rates. It is of course possible to adapt the classification to various demands, e.g., in order to get better recognition rates for the M3 boundaries if more false alarms can be tolerated.

In the following section, prosodically scored word graphs are used for parsing. This means, that for each of the word hypotheses contained in the graph the probability for a clause boundary following this word is computed. The computation of the acoustic features as well as of the LM score is based on ± 2 context words. In the case of the word graphs, the best scored word hypotheses being in the context of a word hypothesis are used. This approach is sub–optimal, but we could show in [11], that the recognition rate does not decrease very much when classifying word graphs instead of the spoken word chain.

5. GRAMMAR AND PARSER

In this paper, we describe the interaction of prosody with the syntax–module developed by Siemens (Munich); for the interaction with another syntax–module developed by IBM (Heidelberg) cf. [1]. In the module described here, we use a **T**race and **U**nification **G**rammar (TUG) [5] and a modification of the parsing algorithm of Tomita [17]. The basis of a TUG is a context free grammar augmented with PATR-II-style feature equations. The Tomita parser uses a graph-structured stack as central data structure [16]. After processing word w_i the top nodes of this stack keep track of all partial derivations for $w_1 ... w_i$. In [15], a parsing-scheme for word graphs is presented using this parser. It combines different knowledge sources when searching the word graph for the optimal word sequence: a TUG, a statistical trigram or bigram model and the score of the acoustic component. In the work described here we added another knowledge source for clause boundaries computed as indicated in Section 4.

When searching the word graph, partial sentence hypotheses are organized as a tree. A graph-structured stack of the Tomita parser is associated with each node. In the

[3]We would like to thank Andreas Klüter, who provided us with these word graphs using the word recognizer described in [20].

(rule1)	`input`	$\rightarrow$	`phrase`	`input`	.
(rule2)	`phrase`	$\rightarrow$	`s`	`PSCB`	.
(rule3)	`phrase`	$\rightarrow$	`s_ell`	`PSCB`	.
(rule4)	`phrase`	$\rightarrow$	`np`	`PSCB`	.
(rule5)	`phrase`	$\rightarrow$	`excl`	`PSCB`	.
(rule6)	`phrase`	$\rightarrow$	`excl` .		

Table 2. Grammar 1 for multiple phrase utterances

search an agenda of score–ranked orders to extend a partial sentence hypothesis ($\text{hypo}_i = \text{hypo}(w_1,...,w_i)$) by a word w_{i+1} or by the `PSCB` symbol, respectively, is processed: The best entry is taken; if the associated graph–structured stack of the parser can be extended by w_{i+1} or by `PSCB`, respectively, new orders are inserted in the agenda for combining the extended hypothesis hypo_{i+1} with the words, which then follow in the graph, and, furthermore, the hypothesis hypo_{i+1} is extended by the `PSCB` symbol. Otherwise, no entries will be inserted. Thus, the parser makes hard decisions and rejects hypotheses which are ungrammatical.

The acoustic, prosodic and trigram knowledge sources deliver scores which are combined to give the score for an entry of the agenda. In the case the hypothesis $hypo_i$ is extended by a word w_{i+1} the score of the resulting hypothesis is computed by

$$\begin{aligned} score(hypo_{i+1}) \quad = \quad & score(hypo_i) \\ & +acoustic\_score(w_{i+1}) \\ & +\alpha \cdot trigram\_score(w_{i-1}, w_i, w_{i+1}) \\ & +\beta \cdot prosodic\_score(w_{i+1}, B) \\ & +'score\ of\ optimal\ continuation' \ . \end{aligned}$$

where B can be `PSCB` or $\neg$`PSCB`. $prosodic\_score(w, \texttt{PSCB})$ is a 'good' score if the prosodic classifier detected a clause boundary after word w, a 'bad' score otherwise. $prosodic\_score(w, \neg\texttt{PSCB})$ is 'good' if the prosodic classifier has evidence that there was no prosodic clause boundary after word w, 'bad' otherwise.

The weights α and β are determined heuristically. Prior to parsing, a Viterbi–like backward pass approximates the scores of optimal continuations of partial sentence hypotheses (A^*–search). After a certain time has elapsed, the search is abandoned. With these scoring functions, hard decisions about the positions of clause boundaries are only made by the grammar but not by the prosody module. If the grammar rules are ambiguous given a specific hypothesis $hypo_i$, the prosodic score guides the search by ranking the agenda.

In order to make use of the prosodic information, the grammar had to be slightly modified. The best results were achieved by a grammar that neatly designed the occurrence of `PSCB`s between the multiple phrases of the utterance. A context–free grammar for spontaneous speech has to allow for a variety of possible input phrases following each other in a single utterance, cf. (rule1) in Table 2. Among those count normal sentences, (rule2), sentences with topic ellipsis (rule3), elliptical phrases like PPs or NPs (rule4), or presentential particle phrases (rule5 and rule6). Those phrases were classified as to whether they require an *obligatory* or *optional* `PSCB` behind them. The grammar fragment in Table 2 says that the phrases `s`, `s-ell` and `np` require an obligatory `PSCB` behind them, whereas `excl`(amative) may also attach immediately to the succeeding phrase (rule 6). The segmentation of utterances according to a grammar like in Table 2 is of relevance to the text understanding components that follow the syntactic analysis, cf. the following two examples which differ w.r.t. the attachment of the exclamative particle *ja*. In the first example it is followed immediately by a sentence (rule6), whereas in the second it is separated by a `PSCB` from the following sentence (rule5). Semantic analysis or dialog can make use of these different

(rule 7)	`input`	$\rightarrow$	`phrase`	,	`PSCB`	,	`input`	.
(rule 8)	`phrase`	$\rightarrow$	`s` .					
(rule 8)	`phrase`	$\rightarrow$	`s_ell` .					
(rule 9)	`phrase`	$\rightarrow$	`np` .					
(rule 10)	`phrase`	$\rightarrow$	`excl` .					

Table 3. Grammar 2 for multiple phrase utterances

rules. The exclamative particle in example (1) might be identified as introduction, in example (2) it might be interpreted as affirmation.

(1) Path found in VM1/N011K/NHW3K002.A16:
`[ja,also,bei,mir,geht,prinzipiell,jeder,Montag,`
`und,jeder,Donnerstag,PSCB]`
Well, as far as I'm concerned, in principle every Monday or Thursday is possible.

(2) Path found in VM4/G275A/G275A002.B16:
`[ja,PSCB,das,pa"st,mir,Dienstag,PSCB,ist,`
`der,f"unfzehnte,PSCB]`
Yes. This Tuesday, that suits me. That is the fifteenth.

The occurrence of the second `PSCB` in example (2) does not mirror the intention of the speaker: Here the `PSCB` divides the subject *Dienstag* from its matrix clause *ist der fünfzehnte*. A hesitation in the input that did not get detected as false alarm might be responsible for this. However (2) is a syntactically correct segmentation since a grammar for spoken language has to allow for topic ellipsis and the phrase *ist der fünfzehnte* constitutes a correct sentence according to (rule 3). The grammar therefore retrieves the interpretation for this lattice as indicated by the English translation.[4]

6. EXPERIMENTAL RESULTS

In experiments using a preliminary version of the sub–grammars for the individual types of phrases, we compared the grammar explained in Section 5 with a grammar that *obligatorily* required a `PSCB` behind every input phrase, see Table 3.

With the grammar shown in Table 2 149 word graphs could successfully be analyzed; with the one given in Table 3, only 79 word graphs were analyzed. This indicates that often the prosody module computes a high score for $\neg$`PSCB` after exclamative particles so that parsing fails if a `PSCB` is obligatorily required as in the grammar of Table 3.

With an improved version of the grammar for the individual phrases, we repeated the experiments using the grammar of Table 2 and compared them with the parsing results using a grammar *without* `PSCB`s. For the latter, we took the category `PSCB` out of the grammar and allowed all input phrases to adjoin recursively to each other. The graphs were parsed without taking notice of the prosodic `PSCB` information contained in the lattice. In this case, the number of readings increases and the efficiency decreases drastically, cf. Table 4. The statistics show that on the average, the number of readings decreases by 96% when prosodic information is used, and the parse time drops by 92%. If the lattice parser does not pay attention to the information on possible `PSCB`s, the grammar has to determine by itself where the phrase boundaries in the utterance

[4] For this word chain, it would make no difference for the text understanding component, whether the `PSCB` is before or after *Dienstag*. Actually, the spoken word chain is: *Ja, das paßt. Nur Dienstag ist der fünfzehnte.* and the dialog goes like this: A: *What about Tuesday the sixteenth?* B: *Yes. That's ok. But Tuesday is the fifteenth.* A: *Sorry. Then let's say Wednesday the sixteenth.* B: *OK. Fine.* B thus only confirms *the sixteenth*, but not *Tuesday*.

	with PSCBs	without PSCBs
# successful analyses	359	368
⊘# syntactic readings	5.6	137.7
⊘ parse time (secs)	3.1	38.6

Table 4. Parsing statistics for 594 word graphs

might be. It may rely only on the coherence and completeness restrictions of the verbs that occur somewhere in the utterance. These restrictions are furthermore softened by topic ellipsis, etc. Any simple utterance like *Er kommt morgen* results therefore in a lot of possible segmentations, see Table 5.

`[er,kommt,morgen]`	*He comes tomorrow.*
`[er],[kommt,morgen]`	*He? Comes tomorrow!*
`[er kommt],[morgen]`	*He comes. Tomorrow!*
`[er],[kommt],[morgen]`	*He? Comes! Tomorrow.*

Table 5. Syntactically possible segmentations

The fact that 9 word graphs (i.e. 2%) could not be analyzed with the use of prosody is due to the fact, that the search space is explored differently and that the fixed time limit has been reached before the analysis succeeded. However, this small number of non–analyzable word graphs is neglectable considering the fact that without prosody, the average real–time factor is 6.1 for the parsing. With prosodic information the real–time factor drops to 0.5; the real–time factor for the computation of prosodic information is 1.0 (with word graphs of about 10 hypotheses per spoken word).

Empty categories are an even more serious problem. They are used by the grammar in order to deal with verb movement and topicalisation in German. The binding of these empty categories has to be checked inside a single input phrase, i.e., the main sentence. No movement across phrase boundaries is allowed. Now, whenever a PSCB signals the occurrence of a boundary, the parser checks whether all binding conditions are satisfied and accepts or rejects the path that was found so far. This mechanism works efficiently in the case prosodic information was used. For the grammar without PSCBs, no signal where to check the binding restrictions is available. Therefore, the uncertainty about segmentation of multiple phrase utterances led to indefinite parsing time for some of the lattices in the corpus. Those lattices were analyzed correctly with PSCBs.

7. CONCLUSION

We showed that prosodic clause boundary information can reduce the parse time of word graphs computed for spontaneous speech by 92%. The number of parse trees of the resulting analyses decreases by 96%. This is especially due to the high number of elliptic and interrupted phrases contained in spontaneous speech, which cause that the position of clause boundaries is highly ambiguous. Apart from differences in the particular technical solutions of some sub–problems, our approach differs from the prosodic parse–rescoring described in [13, 8] mainly in the fact that we first compute prosodic scores based on the word hypotheses generated by the word recognizer. These scores are then integrated directly into the parsing process which does not only reduce the number of readings but also the parse time.

REFERENCES

[1] A. Batliner, A. Feldhaus, S. Geißler, T. Kiss, R. Kompe, and E. Nöth. Prosody, Empty Categories and Parsing — A Success Story. In *Proc. ICSLP*, volume 2, pages 1169–1172, Philadelphia, 1996.

[2] A. Batliner, R. Kompe, A. Kießling, M. Mast, and E. Nöth. All about Ms and Is, not to forget As, and a comparison with Bs and Ss and Ds. Verbmobil Memo 102, 1996.

[3] A. Batliner, R. Kompe, A. Kießling, H. Niemann, and E. Nöth. Syntactic–prosodic Labelling of Large Spontaneous Speech Data–bases. In *Proc. ICSLP*, volume 3, pages 1720–1723, Philadelphia, 1996.

[4] H.U. Block. The Language Components in Verbmobil. In *Proc. ICASSP*, München, 1997.

[5] H.U. Block and S. Schachtl. Trace & Unification Grammar. In *Proc. COLING*, volume 1, pages 87–93, Nantes, 1992.

[6] T. Bub and J. Schwinn. Verbmobil: The Evolution of a Complex Large Speech-to-Speech Translation System. In *Proc. ICSLP*, volume 4, pages 1026–1029, Philadelphia, 1996.

[7] A. Feldhaus and T. Kiss. Kategoriale Etikettierung der Karlsruher Dialoge. Verbmobil Memo 94, 1995.

[8] A. Hunt. A Generalised Model for Utilising Prosodic Information in Continuous Speech Recognition. In *Proc. ICASSP*, volume 2, pages 169–172, Adelaide, 1994.

[9] A. Kießling. Extraktion und Klassifikation prosodischer Merkmale in der automatischen Sprachverarbeitung. Dissertation. Technische Fakultät der Universität Erlangen–Nürnberg, 1996.

[10] R. Kompe. Prosody in Speech Understanding Systems. Dissertation. Technische Fakultät der Universität Erlangen–Nürnberg, 1996.

[11] R. Kompe, A. Kießling, H. Niemann, E. Nöth, E.G. Schukat-Talamazzini, A. Zottmann, and A. Batliner. Prosodic Scoring of Word Hypotheses Graphs. In *Proc. EUROSPEECH*, volume 2, pages 1333–1336, Madrid, 1995.

[12] H. Niemann, E. Nöth, A. Kießling, R. Kompe, and A. Batliner. Prosodic Processing and its use in Verbmobil. In *Proc. ICASSP*, München, 1997.

[13] M. Ostendorf, C.W. Wightman, and N.M. Veilleux. Parse Scoring with Prosodic Information: an Analysis/Synthesis approach. *Computer Speech & Language*, 7(3):193–210, 1993.

[14] M. Reyelt and A. Batliner. Ein Inventar prosodischer Etiketten für Verbmobil. Verbmobil Memo 33, 1994.

[15] L.A. Schmid. Parsing Word Graphs Using a Linguistic Grammar and a Statistical Language Model. In *Proc. ICASSP*, volume 2, pages 41–44, Adelaide, 1994.

[16] N. Sikkel. *Parsing Schemata*. CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, 1993.

[17] M. Tomita. *Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems*. Kluwer Academic Publishers, 1986.

[18] H. Tropf. Spontansprachliche syntaktische Phänomene: Analyse eines Korpus aus der Domäne "Terminabsprache". Technical report, Siemens AG, ZFE ST SN 54, München, 1994.

[19] W. Wahlster, T. Bub, and A. Waibel. Verbmobil: The Combination of Deep and Shallow Processing for Spontaneous Speech Translation. In *Proc. ICASSP*, München, 1997.

[20] A. Waibel, M. Finke, D. Gates, M. Gavalda, T. Kemp, A. Lavie, L. Levin, M. Maier, L. Mayfield, A. McNair, K. Shima, T. Sloboda, M. Woszczyna, T. Zeppenfeld, and P. Zhan. JANUS–II — Translation of Spontaneous Conversational Speech. In *Proc. ICASSP*, volume 1, pages 409–412, Atlanta, 1996.