gefördert vom
Bundesministerium
für Forschung
und Technologie

Verbmobil
Verbundvorhaben

Improving Parsing by Incorporating 'Prosodic Clause Boundaries' into a Grammar

G. Bakenecker, U. Block
A. Batliner
R. Kompe, E. Nöth
P. Regel-Brietzmann

Report 37
November 1994

November 1994

G. Bakenecker, U. Block
A. Batliner
R. Kompe, E. Nöth
P. Regel-Brietzmann

Siemens AG
ZFE ST SN 74
Otto-Hahn-Ring 6
D–81730 München

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

Institut für Deutsche Philologie
Ludwig–Maximilian Universität München
Schellingstr. 3
D–80799 München

Daimler–Benz AG
Forschungsinstitut Ulm
Wilhelm–Runge Str. 11
D–89081 Ulm

Tel.: (089) 636 - 49249
e-mail: {Gabriele.Bakenecker}@zfe.siemens.de

Gehört zum Antragsabschnitt: 3.6, 3.11, 3.12, 6.4

Das diesem Bericht zugrundeliegende Forschungsvorhaben wurde mit Mitteln des Bundesministers für Forschung und Technologie unter dem Förderkennzeichen 01 IV 102 H/0 und 01 IV 102 C 6 gefördert. Die Verantwortung für den Inhalt dieser Arbeit liegt bei dem Autor.

IMPROVING PARSING BY INCORPORATING 'PROSODIC CLAUSE BOUNDARIES' INTO A GRAMMAR

G. Bakenecker, U. Block
Siemens AG, Otto-Hahn-Ring 6, 81730 München, Germany
A. Batliner
L.-M. Universität, Institut für Deutsche Philologie, Schellingstr. 3, 80799 München, Germany
R. Kompe, E. Nöth
Universität Erlangen-Nürnberg, Lehrstuhl für Mustererkennung (Informatik 5), Martensstr. 3, 91058 Erlangen, Germany
P. Regel-Brietzmann
Daimler-Benz AG, Wilhelm-Runge Str. 11, 89081 Ulm, Germany

ABSTRACT

In written language, punctuation is used to separate main and subordinate clause. In spoken language, ambiguities arise due to missing punctuation, but clause boundaries are often marked prosodically and can be used instead. We detect PCBs (**P**rosodically marked **C**lause **B**oundaries) by using prosodic features (duration, intonation, energy, and pause information) with a neural network, achieving a recognition rate of 82%. PCBs are integrated into our grammar using a special syntactic category 'break' that can be used in the phrase-structure rules of the grammar in a similar way as punctuation is used in grammars for written language. Whereas punctuation in most cases is obligatory, PCBs are sometimes optional. Moreover, they can in principle occur everywhere in the sentence due e.g. to hesitations or misrecognition. To cope with these problems we tested two different approaches: A slightly modified parser for word chains containing PCBs and a word graph parser that takes the probabilities of PCBs into account. Tests were conducted on a subset of infinitive subordinate clauses from a large speech database containing sentences from the domain of train table inquiries. The average number of syntactic derivations could be reduced by about 70 % even when working on recognized word graphs.

I. INTRODUCTION

Prosody is used to mark phrase boundaries while speaking. This structures the utterance and helps the listener to understand and disambiguate the meaning. To our knowledge so far nobody has really integrated information about prosodic phrase boundaries in automatic speech understanding systems. This paper presents a first step towards this goal (for similar work cf. [11, 2]).

Two years ago we built a complete automatic speech understanding (ASU) system consisting of a word recognizer, a parser, and a module for semantic interpretation; the domain is train time table inquiry [6]. Only utterances consisting of a single main clause could be processed. Currently we are adapting this system to the domain of appointment scheduling in the VERBMOBIL project [12]. In a corpus of 50 VERBMOBIL dialogues about 70 % of the utterances contained more than a single sentence, cf. [10]. In speech the boundary between two clauses is not marked by punctuation as in written language and because of missing punctuation the position of a boundary is quite often ambiguous. For example consider two of the at least 36 different syntactic parses for the potential VERBMOBIL utterance

"*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.*"

The possible boundary position is indicated by "|". Note that an utterance with the second interpretation can also be produced with a rise at the end (signal for take over of the turn). In this case only the PCB disambiguates between the two different semantic meanings and pragmatic interpretations. Also, the missing punctuation leads to a high amount of ambiguity and therefore to a strong increase of computation time.

Thus we recently started on integrating a prosody module into our ASU system. Input to the prosody module is a word graph generated by the word recognizer (Section 3). For each word in the word graph the probability that it is succeeded by a clause boundary is computed. This information is intended to help the parser reduce the number of ambiguities. Since in VERBMOBIL training databases for word recognizers and classifiers have only been made available recently, we started our studies on the train table inquiry domain. For this a training speech database is available where utterances only consist of one main clause and sometimes an additional subordinate clause. In the case of infinitive subordinate clauses the boundary position is rather ambiguous. We concentrated on such sentences, because we expect speakers to mark the boundaries between main clause and subordinate clause prosodically similar to the boundaries between two main clauses. For example in *"Schaffe ich es | noch | heute | um sechs Uhr | in Hamburg zu sein."* (word by word translation: *can I | still | today | at six o'clock | in Hamburg to be*) there are four possible boundary positions. If only the word chain is given, four alternative parse trees are thus possible. The boundary position decides whether the 'can' or the auxiliary verb 'to be' is modified by the adverb and/or the PPs. Contrary to the VERBMOBIL example given above, here the semantics is almost the same. However, disambiguating the syntax could also speed up the parse process. This in turn could lead to the right interpretation that would not have been found due to resource limitations in the search without prosodic information.

In this paper we first present an overview of the speech database (Section 2), the word recognizer (Section 3), the grammar and the parser (Section 4), and the reference boundary markers (Section 5), Then in Section 6 the classifier (an artificial neural network) for the prosodic boundaries is characterized. In Section 7 we describe how the linguistic grammar is extended so that it "knows" about PCBs and how the hypotheses about boundaries computed by the prosody module can be integrated in an A^*–search in order to disambiguate the parsing. Experiments with the extended parser were performed using the spoken word chains (Section 8) and recognized word graphs (Section 9).

II. SPEECH DATABASE

The material we investigated is part of the German speech database ERBA, "**Er**langer **B**ahn **A**nfragen" (Erlangen train inquiries). A stochastic sentence generator was used based on a context free grammar and 38 sentence templates to create a large text corpus with utterances consisting of one sentence with or without a subordinate clause and a short elliptic sentence.

10,000 unique sentences were recorded in quiet office environments (100 untrained speakers, 100 utterances each) resulting in a speech database of about 14 hours. The speakers were given the word sequences with punctuation marks, but without the prosodic phrase boundary markers. 69 speakers (25 female, 6,900 sentences) were used for training and 21 speakers (9 female, 2,100 sentences) for testing the word recognition module as well as the classifier for the prosodic boundaries. For more details concerning ERBA see [1].

The grammar described in Section 4 is only able to parse 4,480 of the 10,000 spoken word chains, because e.g. it was not designed to handle elliptic sentences. In the set of 4,480 sentences there are 1,504 sentences with subordinate clauses (and thus with a potential PCB), 272 of them being infinitive clauses (i.e. the PCB could help disambiguate between the possible parses). As mentioned above, the low percentage of interesting sentences is due to the construction of the corpus and way higher for the VERBMOBIL data. Due to technical problems only 242 of these 272 sentences could be used in the experiments described in Section 9. Note that about two third of these sentences were in the training set for both the word recognizer and the classifier for the PCBs. However, in the case of the PCB the recognition rates of training and test set do not differ significantly.

III. WORD RECOGNIZER

The Daimler-Benz speech recognition system is based on semi-continuous Hidden Markov Models (SCHMM) of subword units (generalized triphones and functional words) [4, 5]. We use 13 normalized mel–based cepstral features with a centisecond frame rate. Combinations of feature vectors from 9 adjacent frames (dimension 117) are transformed by a linear transform generated by linear discriminant analysis. This results in a 32 dimensional feature vector which implicitly includes temporal dynamic effects. The multistage training procedure is described in [4]. The output of our recognizer is a word hypotheses graph.

IV. GRAMMAR AND PARSER

We use a **T**race and **U**nification **G**rammar (TUG) [3] and a modification of the parsing algorithm of Tomita [9]. 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. After processing word w_i the top nodes of this stack keep track of all partial derivations for $w_1...w_i$. In [8], a parsing-scheme for word graphs is presented using this parser. It combines different knowledge sources when searching the word graph for the spoken utterance: a TUG, a statistical bigram model and the score of the acoustic component. 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 search an agenda of score-ranked orders to extend a partial sentence hypothesis ($hypo_i$ = $hypo(w_1,...,w_i)$) by a word w_{i+1} is processed: The best entry is taken; if the associated graph-structured stack of the parser can be extended by w_{i+1} new orders are inserted in the agenda for combining the extended hypothesis $hypo_{i+1}$ with the then following words. Otherwise, no entries will be inserted. Thus, the grammar makes hard decisions on whether a hypothesis is accepted or not. The other two knowledge sources (the acoustic and the bigram model) deliver scores which are combined to give the score for an entry of the agenda:

$$score(hypo_i \text{ \& } word) = score(hypo_i) + acoustic\_score(word) + alpha * bigram\_score(w_i, word) + \text{`}score\ of\ optimal\ continuation'$$

Alpha is determined heuristically. Prior to parsing a Viterbi-like backward pass computes the exact scores of optimal continuations of partial sentence hypotheses (A^*–search). After a certain time has elapsed the search is abandoned.

V. REFERENCE BOUNDARY MARKERS

Prosodic phrase boundaries can be predicted quite accurately using syntactic knowledge. Syntactic boundaries were therefore marked in the context free grammar (Section 2) and included in the sentence generation process with some context-sensitive post-processing (cf. below: B1 boundaries). The text read by the speakers did not contain these markers.

We distinguish four types of phrase boundaries: Boundary B3 is placed between elliptic clause and clause or between main and subordinate clause, B2 is positioned between constituents or at coordinating particles between constituents, B1 belongs syntactically to the normal constituent boundary B2 but is most certainly not marked prosodically because it is close to a B3 boundary or to the beginning/end of the utterance, and B0 is any other word boundary that does not belong to B1, B2 , B3; for more details see [1]. The following sentence shows examples for these boundary types: "*Guten Morgen* B3 *ich hätte gerne* B1 *einen Zug* B3 *der München* B2 *zwischen sechs* B2 *und sieben Uhr* B1 *verläßt*" (word by word translation: "*Good morning* B3 *I would like* B1 *a train* B3 *that Munich* B2 *between six* B2 *and seven o'clock* B1 *leaves*". In the ERBA corpus, 62097 B0, 18657 B1, 22616 B2, and 3877 B3 boundaries are generated automatically.

A perception experiment was conducted with "naive" listeners [1]. It showed that there is a very high agreement between the automatically generated reference boundaries and perceived boundaries.

VI. AUTOMATIC BOUNDARY CLASSIFICATION

The classification of prosodic phrase boundaries is based on the time alignment of the recognized/spoken words. For the final syllable of each word prosodic features (duration, intonation, intensity, and pause information) are computed; as for more details, cf. [7]. There we reported an average error rate of 60% for the three classes B01, B2, and B3 using a Gaussian classifier. Meanwhile we improved the recognition rate to 72% with an improved feature set, modeling multi–modality, and using multi–layer perceptrons (MLP).

The durational features highly depend on the underlying syllable. Thus when working on word graphs for each of the words ending in the same node a different feature vector is computed as input for the MLP. The probability for a PCB in a node depends very much on the word under consideration and is computed for each of the words separately. Our ASU system works in bottom–up manner, i.e. first the word graph is computed for the whole utterance, then the MLP computes likelihoods for PCBs, and finally the word graph is parsed. Thus when computing the likelihoods for the PCBs the syllables succeeding the final syllable of a word are not known (or the likelihood would have to be calculated for each of the potential successor words). Therefore the length of a syllable to the right of the potential PCB is defined as 200 msec for the computation of the F0 and intensity features. In comparison to the recognition rate using the time alignment of the best word chain this decreases the recognition rate on the three classes by about 2%, i.e. 70% for the three classes. For the two class problem B012 vs. B3 that is relevant for this paper we currently have 82%. This MLP was used for the experiments in Section 9. The experiments on the spoken word chain (Section 8) were performed some time ago using an older version of a Gaussian classifier with which a recognition rate of 56% for the three classes was achieved.

VII. EXTENSION OF THE GRAMMAR AND THE PARSER

We incorporated PCBs into grammar and parser in order to reduce the number of syntactic derivations and speed up parsing. In written language, commas are used to separate two main clauses and main and subordinate clause. Without punctuation marks, sentences like the following are ambiguous (translation word by word):

Welche Möglichkeiten habe ich heute nach Hamburg zu kommen
which possibilities have I today to Hamburg to go

'heute' ('today') can modify the main clause or the subordinate clause. In such cases PCBs can be used to replace punctuation. In analogy to punctuation in written language, a special syntactic category **break** for PCBs is introduced that can be used in the phrase structure part of the grammar. E.g. the rule

INPUT → subclause-inf , main-clause

is modified to

INPUT → subclause-inf , break , main-clause .

A **grammar with obligatory PCBs** only includes the rules with the break category, a **grammar with optional PCBs** includes the rules with and without the break category. In [2] a similar approach was followed, but a special category was inserted between every two adjacent symbols on the right hand side of each grammar rule. In the average this resulted in an increase in parsing time.

We implemented two basically different approaches. In the **first approach** (see experiment E1 in Section 8) the input to the parser is an ASCII-string that includes the probability of a PCB between adjacent words:

welche 0.21 Möglichkeiten 0 habe 0.11 ich 0.14 nach 0.16 drei 0 Uhr 0 nach 0.01 Goslar 0.1 zu 0 kommen
which possibilities have I after three o'clock to Goslar to go

welche 0.23 Möglichkeiten 0.01 habe 0.11 ich 0.14 in 0.77 fünf 0.28 Wochen 0.02 von 0 Eberswalde 0.01 nach 0 Bodenwöhr_Nord 0.82 über 0 Mannheim 0 zu 0 fahren
which possibilities have I in five weeks from Eberswalde to Bodenwöhr_Nord via Mannheim to go

Using a threshold of 0.5 the probabilities are first transformed into hard decisions yielding:

(1) *welche Möglichkeiten habe ich nach drei Uhr nach Goslar zu kommen*

(2) *welche Möglichkeiten habe ich in* **B3** *fünf Wochen von Eberswalde nach Bodenwöhr_Nord* **B3** *über Mannheim zu fahren*

(1) indicates that PCBs are not always produced or recognized and according to (2) they can in principal be hypothesized everywhere in the sentence due e.g. to hesitations or misrecognition. To account for (1) the parser uses a grammar with optional PCBs and to account for additional PCBs as in (2) the parser is slightly modified: In case of failure it skips the previously consumed PCB and continues parsing from that point on. In (2) the parser fails to consume the first B3 because infinitive subordinate clauses can not start after a preposition. The first B3 is skipped and the parser succeeds.

There can be several possibilities which PCBs to skip and which to consume. One disadvantage of this approach is that the parser does not choose the optimal possibility.

In the **second approach** (see experiments E2 and E3 in Section 8) we use the word graph parser. Each hypothesis of the word graph includes two additional scores, one for 'a PCB is following' and the other for 'no PCB is following'. The word graph parser has been modified to take PCBs into account: After adding a word w_{i+1} to a partial sentence hypothesis $hypo_i = hypo(w_1,...,w_i)$ an order for combining the new partial sentence hypothesis $hypo_{i+1}$ with a B3 (1) and orders for combining it with the then following words (2) are inserted into the agenda. When adding B3 to a hypothesis only orders for combining it with following words (3) are inserted. The scoring function has been modified to take the prosodic score into account:

$$score(hypo_i \;\&\; B3) = score(hypo_i) + \quad (1)$$
$$beta * PCB\_score(w_i) + `score\ of\ optimal\ continuation'$$

$$score(hypo_i \& w_{i+1}) = score(hypo_i) + \quad (2)$$
$$beta * no\_PCB\_score(w_i) + acoustic\_score(w_{i+1}) +$$
$$alpha * bigram\_score(w_i, w_{i+1})) +$$
$$`score\ of\ optimal\ continuation'$$

$$score(hypo(w_1, .., w_i, B3) \;\&\; w_{i+1}) = \quad (3)$$
$$score(hypo(w_1, .., w_i, B3)) + acoustic\_score(w_{i+1}) +$$
$$alpha * bigram\_score(w_i, w_{i+1})) +$$
$$`score\ of\ optimal\ continuation'$$

The exact scores of optimal continuations of partial sentence hypotheses are determined using the best of the two prosodic scores.

Table 1. Results for the spoken word chain

	E0	E1	E2	E3
number of failed sentences				
4480 sentences	0	0	71	206
1504 sentences	0	0	25	62
272 infinitive-sentences	0	0	5	27
comparison for the 245 infinitive clauses				
average number of derivations	8.1	4.03	4.14	2.44
max number of derivations	209	39	39	38
average runtime secs	8.06	4.31	4.59	4.04
max runtime	235.02	33.07	36.78	29.38

VIII. EXPERIMENTS USING THE SPOKEN WORD CHAIN

In the following experiments the spoken word chain was used. The probability of PCBs are inserted automatically between adjacent words. The word chain can easily be transformed into a word graph: Each word is assigned the acoustic score 0. When searching on this word graph the bigram model is of no use and the bigram score can be ignored, i.e. the score of a hypothesis is the prosodic score. When using a **grammar with optional PCBs** the word graph parser searches for the best PCBs within the sentence that are accepted by the grammar, eventually omitting all PCBs. When using a **grammar with obligatory PCBs** the word graph parser is forced to decide where to place the boundary and chooses the optimal position accepted by the grammar. It can fail to parse the sentence in case a PCB is recognized with probability 1 at a position in the sentence where the grammar would not allow any. When using a grammar with obligatory PCBs it can also fail if PCBs are recognized with probability 0 at all possible positions in the utterance.

In the following four experiments (E0 - E3) are compared:

E0: uses the original sentence parser and grammar and ignores PCBs

E1: uses the modified sentence parser and grammar with optional PCBs

E2: uses the modified word graph parser and grammar with optional PCBs

E3: uses the modified word graph parser and grammar with obligatory PCBs

In Table 1 the number of failed sentences, the average number of derivations and runtime and the maximal number of derivations and runtime are given for these four experiments and for the 245 sentences which could be processed successfully in all experiments. As can be seen the average number of derivations and runtime is reduced by about 50 % when incorporating PCBs as hard decisions (E1). Taking into account the probability of PCBs and using a grammar with optional PCBs (E2) results in a similar improvement, though about 2 % (5 cases) of the sentences fail. When forcing the parser to find the best position to place the PCBs (E3) the average number of derivations is even reduced by about 70 %, but the number of failed sentences increases to 10 % (27 cases). We hope that improving recognition rates for PCBs will decrease the number of failed sentences.

Table 2. Results for real word graphs

	derivations	runtime
without prosody		
242 sentences	-	13.74
145 correct	8.97	5.31
prosody 0.001		
242 sentences	-	14.35
144 + 0 correct	2.65	5.76
prosody 0.01		
242 sentences	-	14.15
143 + 1 correct	2.65	5.89
prosody 1		
242 sentences	-	17.29
142 + 1 correct	2.64	6.04
prosody 10		
242 sentences	-	21.99
86 + 4 correct	2.12	7.55

IX. EXPERIMENTS USING THE RECOGNIZED WORD GRAPH

In these experiments prosodic scores were attached to the word hypotheses of 242 word graphs for sentences with infinitive subordinate clauses. A grammar with obligatory PCBs was used. A bigram model was trained on about 4500 ERBA-sentences including the 242 sentences yielding a perplexity of 37 for the 242 sentences. In 173 cases the spoken word chain was included in the word graph. Using no prosodic information 145 sentences were correctly recognized, in 75 cases a different sentence was accepted by the grammar (25 of these included the spoken word chain) and in the remaining 22 cases the time limit was reached. The prosodic score was included in the scoring-function with different factors: 0.001,0.01,1 and 10. Table 2 shows the average number of syntactic derivations and the average runtime, the first one only given for correctly recognized sentences. The number of correct sentences when using prosodic information is given as $n_1 + n_2$, n_1 sentences being part of the 145 sentences which were correctly recognized without using prosodic information. (Note that runtimes of Table 1 and Table 2 can not be compared because different machines were used). As can be seen the number of correctly recognized sentences decreases and the runtime increases when the factor with which the prosodic information is taken into account increases. Thus in these preliminary results the prosodic information did not lead to an increase in sentence recognition rate, but it helped to reduce the number of derivations by about 70 %. Up to a weighting factor of 1 the reduction in the recognition rate and the increase in the runtime are minute.

X. CONCLUSION

In this paper we presented a first step towards the use of prosodic information during parsing. We showed that the number of alternative parse trees can be reduced significantly (about 70 %). However, so far it did not speed up the time needed for parsing word graphs. On the one hand, we believe that there is still room for improvement using other sets of features and improving the so far not optimized time alignment of the words in the word graph. On the other hand, when switching to spontaneous speech we will have to deal with hesitations and changes of speaking rate within the same utterance. These phenomena must first be modeled before we can hope for similar reduction factors.

Furthermore we intend to use prosodic boundary information for resolving other types of ambiguities such as the attachment of prepositional phrases, of appositions and of adverbials. Especially the attachment of prepositional phrases is rather ambiguous without information about phrase boundaries; e.g. *"I saw the man with a telescope"*, or *"I want to take the train to Munich"*.

Acknowledgements

This work was supported by the German Ministry for Research and Technology (*BMFT*) in the joint research projects ASL and VERBMOBIL. Only the authors are responsible for the contents.

REFERENCES

[1] A. Batliner, R. Kompe, A. Kießling, E. Nöth, H. Niemann, and U. Kilian. The prosodic marking of phrase boundaries: Expectations and Results. In A. Rubio, editor, *New Advances and Trends in Speech Recognition and Coding*, NATO ASI Series F. Springer–Verlag, Berlin, Heidelberg, New York, 1994. (to appear).

[2] J. Bear and P.J. Price. Prosody, syntax and parsing. In *Proceedings of the ACL Conference*, pages 17–22, 1990.

[3] H. Block and S. Schachtl. Trace & unification grammar. In *PROC. OF COLING-92*, pages 87–93, Nantes, AUG 1992.

[4] F. Class, A. Kaltenmeier, and P. Regel. Evaluation of an HMM Speech Recognizer with Various Continuous Speech Databases. In *Proc. European Conf. on Speech Communication and Technology*, volume 2, pages 803–806, Berlin, September 1993.

[5] F. Class, A. Kaltenmeier, and P. Regel. Optimization of an HMM-based continuous Speech Recognizer. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 1587–1590, Berlin, September 1993.

[6] P. Heisterkamp, S. McGlashan, and N.J. Youd. Dialogue Semantics for a Spoken Dialogue System. In *Proc. Int. Conf. on Spoken Language Processing*, October 1992. Banff, Canada.

[7] R. Kompe, A. Batliner, A. Kießling, U. Kilian, H. Niemann, E. Nöth, and P. Regel-Brietzmann. Automatic Classification of Prosodically Marked Phrase Boundaries in German. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 173–176, Adelaide, 1994.

[8] L. Schmid. Parsing Word Graphs Using a Linguistic Grammar and a Statistical Language Model. In *Proceedings ICASSP*, pages II–41–II44, 1994.

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

[10] H. Tropf. Spontansprachliche syntaktische Phänomene: Analyse eines Korpus aus der Domäne "Terminabsprache". Technical report, Siemens AG, München, April 1994.

[11] N.M. Veilleux, M. Ostendorf, and C.W. Wightman. Parse Scoring with Prosodic Information. In *Int. Conf. on Spoken Language Processing*, volume 3, pages 1605–1608, Banff, 1992.

[12] W. Wahlster. Verbmobil — Translation of Face–To–Face Dialogs. In *Proc. European Conf. on Speech Communication and Technology*, volume "Opening and Plenary Sessions", pages 29–38, Berlin, September 1993.