MAKING THE MOST OF MULTIPLICITY: A MULTI-PARSER MULTI-STRATEGY ARCHITECTURE FOR THE ROBUST PROCESSING OF SPOKEN LANGUAGE.

Tobias Ruland[1] *C. J. Rupp*[2] *Jörg Spilker*[3] *Hans Weber*[3] *Karsten L. Worm*[2]

[1]Siemens AG, ZT IK 5, D-81730 Munich, Germany, Tobias.Ruland@mchp.siemens.de
[2]University of the Saarland, Computational Linguistics, D-66041 Saarbrücken, Germany,
{cj,worm}@coli.uni-sb.de
[3]Friedrich-Alexander-University Erlangen-Nuremberg, Computer Science Institute,
D-91058 Erlangen, Germany, {spilker,weber}@faui80.informatik.uni-erlangen.de

ABSTRACT

This paper describes ongoing research on robust spoken language understanding in the context of the Verbmobil speech-to-speech machine translation project. We focus on recent developments in the processing steps which map a word lattice to a semantic representations. The approach described firstly applies speech repair correction to word lattices. Four analysis methods of varying depth are then applied in parallel to the normalized word lattices, producing output for sub-portions of the lattice in the same semantic description language, the VIT format. These fragmentary analyses are stored and combined by a further processing component, which finally selects a sequence of semantic representations as a result.

1. INTRODUCTION

In this paper, we describe relevant modules of the linguistic analysis component of the forthcoming Verbmobil speech-to-speech translation system [11]. In particular, we discuss Verbmobil's robust multi-parser architecture and how the different parsers are controlled, the treatment of utterances containing self-repairs, the integration of partial analyses resulting from recognition errors or ungrammaticalities, and the selection of a result from competing hypotheses. The overall aim is to show how we attempt to make Verbmobil more robust against the typical problems of processing spontaneous speech.

These efforts to make the system more robust are applied at different stages of processing. The module for the treatment of self-repairs takes as input the word hypothesis graph delivered by the speech recognizer and annotated by the prosody component. It adds new edges bridging possible self-repairs in the graph. Integration of partial analyses can be viewed as a post-parsing process (although technically it takes place in parallel with parsing). The use of multiple parsers, each with its own strengths and weaknesses, taken together with a selection process that chooses from the different results available, further improves robustness. The overall flow of data can be seen in figure 1.

Verbmobil employs a semantic transfer approach to machine translation [6], i. e. an input utterance is syntactically and semantically analyzed, the resulting source language semantic representation is mapped to a target semantic representation, from which a target language utterance in generated and synthesized. Apart from this, alternative strategies such as example-based and statistical translation are explored. In this paper, we focus on the linguistic analysis for the transfer-based processing.

2. TREATMENT OF SELF-REPAIRS

From an architecture point of view the multi-parser architecture makes a preprocessing of word lattices necessary. Otherwise given a perfect acoustic word recognition, in cases of speech repairs the grammar based analysis methods would never produce an output. The "repair correction" step itself relies on the classical treatment of speech repairs as *Reparandum* (RD), Interruption point (IP), *Edit Term* (ET) and *Reparans* (RS) as in "[Monday]$_{RD}$ IP [no]$_{ET}$ [Tuesday]$_{RS}$". If such a word sequence were uttered, in the ideal case the corresponding sequence of word hypotheses [Monday no Tuesday] would be replaced by just [Tuesday]. The word lattice correction of repairs divides into two phases of search, given a preprocessed word lattice as input, where word boundaries are classified according to prosodic cues whether they might constitute a word boundary immediately following a reparandum[1].

First the word lattice is collapsed to a POS[2] lattice. Second a set of nodes prosodically marked as interruption points is selected. For each of these nodes a probabilistic model on POS sequences is used to classify the incoming and outgoing word sequences into RD, ET, and RS. We use a specialized tag set for that step which covers semantic features as well according to their linguistic relevance for the repair phenomenon. The last phase — the editing step — monotonically adds new edges to the word lattice spanning the original RD ET RS sequences but being labelled only by the RS label. We are currently concentrating on improving the POS-based classifiers which are used to detect the scope of the self-repair. Within the Verbmobil corpus with spontaneous negotiation dialogues about 20% of the utterance exhibit self-repairs. We can currently isolate about 94% of the reparanda correctly given the correct string and irregular boundary, if we restrict ourselves to the self-repairs with less than five words (95% of the corpus).

3. INTEGRATED PROCESSING

In the current state of the architecture four different parsing methods are incorporated in the "Integrated Processing" module [10]. All of these produce semantic representations in the same formalism (see 4), which can be combined with each other.

1. The first method is a *deep linguistic HPSG parser* [8], which is not very robust but produces very detailed descriptions for its inputs.
2. The second method is a *probabilistic context free grammar LR-parser*, where the grammar and the stochastic parameters are derived from a tree bank. The grammar is supplied

[1] The prosodic classification component which we use is described in [2].

[2] Part of speech.

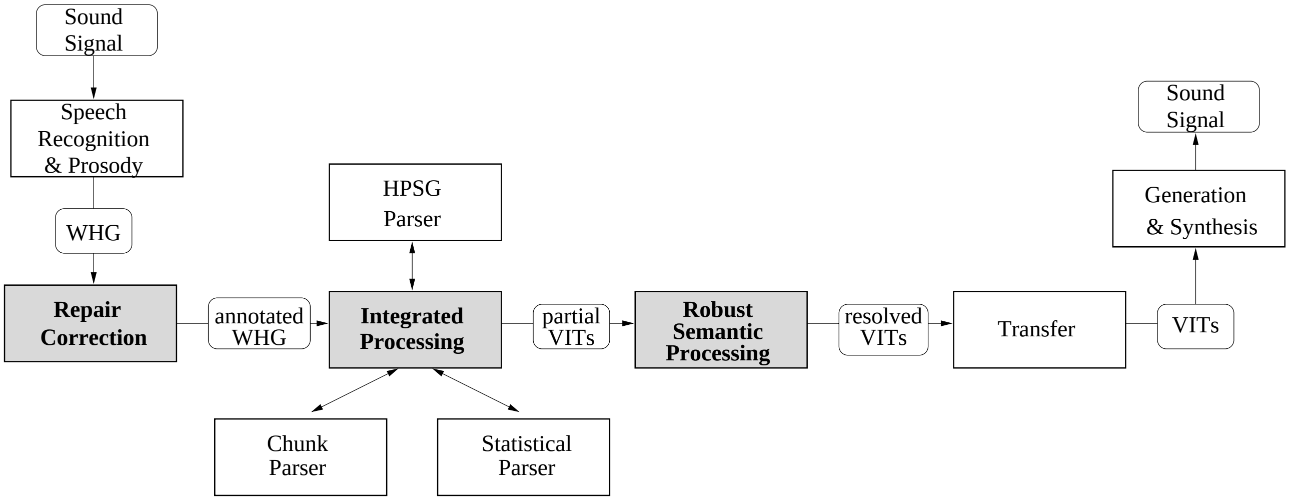

Figure 1: The relevant part of the system architecture.

with a semantic construction mechanism. However, the representations it produces are usually less detailed than those of the HPSG parser. In many cases where the HPSG fails the probabilistic grammar still produces an interpretation.

3. The third method is a *chunk parser* based on cascaded finite state automata as described in [1], producing rough interpretations on analysable fragments of the input.
4. As a fall-back an HMM-based dialogue-act recognizer is used as the fourth method. This method produces a template intepretation for the dialogue act recognized in each input where special slots like weekdays and clocktimes are filled by additional rules [9].

The backbone of the module is an A*-lattice-search with a trigram-based rest cost calculation [7] which guides the search of all the parsing methods through the input lattice. The parsing methods can be run on a single processor machine (using its own scheduling heuristics) or simultaneously on multiple processors. Since the increasing robustness of the methods (increasing from HPSG to dialogue-act based analysis) corresponds to their decreasing precision and computational resources needed, the "Integrated Processing" module as a whole can be parametrized to show an anytime behaviour.

4. INTEGRATION OF PARTIAL ANALYSES

In many cases, no parser will find an analysis spanning the whole input utterance. This may be due to speech recognizer errors, spontaneous speech phenomena which have not been caught earlier, and ungrammaticalities in the utterance itself. Although a complete analysis would be preferable, the parser can usually come up with a set of partial analyses in these cases which can often be assembled to yield larger, more meaningful units. This is the basic idea of what we call *robust semantic processing* [14, 15].

Robust semantic processing operates as a background process to the analysis performed by the parsers. While the parsers examine and analyse the paths in the word hypothesis graph they receive from integrated processing, they produce partial analyses of these, covering part of the paths. These partial results are sent to the robust semantic processing component. Since all parsers deliver their results in the common VIT format, the results are comparable and combinable.

The VIT format (VIT stands for Verbmobil Interface Term, cf. [4]) has been developed as a common semantic representation format for the different Verbmobil grammars and parsers. It can be seen as a theory-independent version of underspecified semantics [5]. An example of a VIT is given in figure 3.

The task of robust semantic processing then consists of three subtasks:

1. *store* the partial results in a chart-like data structure (which we call a *VIT Hypothesis Graph* (VHG),
2. *combine* the partial results on the basis of rules, yielding new entries in the VHG,
3. *select* a result from the VHG, i.e. a sequence of partial results (or a complete one, if available), if no parser was able to find a spanning analysis in the time available.

Consider as an example the utterance

(1) Wir treffen uns in den nächsten zwei Wochen.
(*We (will) meet during the next two weeks*)

and assume that the speech recognizer dropped the preposition *in*, as it is just a short word. In this case, the parser will analyze the input as two fragments, a sentence (*wir treffen uns*) and a nominal phrase (*den nächsten zwei Wochen*). These two fragments are stored by the robust semantic processing. A rule stating that a temporal NP such as *den nächsten zwei Wochen* can be re-interpreted as a modifier is applied, entering a new edge into the chart. This temporal modifier edge is then combined with the edge for the proposition, yielding a complete and accurate analysis of the complete utterance.

The resulting VIT hypothesis graph is shown in figure 2. The edges are numbered, the numbers correspond to the temporal order in which they were added to the chart. The edges are annotated with the substring(s) they correspond to, as well as with an internal score and a list (in PROLOG notation) of the numbers of the edges they have been built from. E. g., edges 89 and 106 have been delivered by a parser, since they have not been built

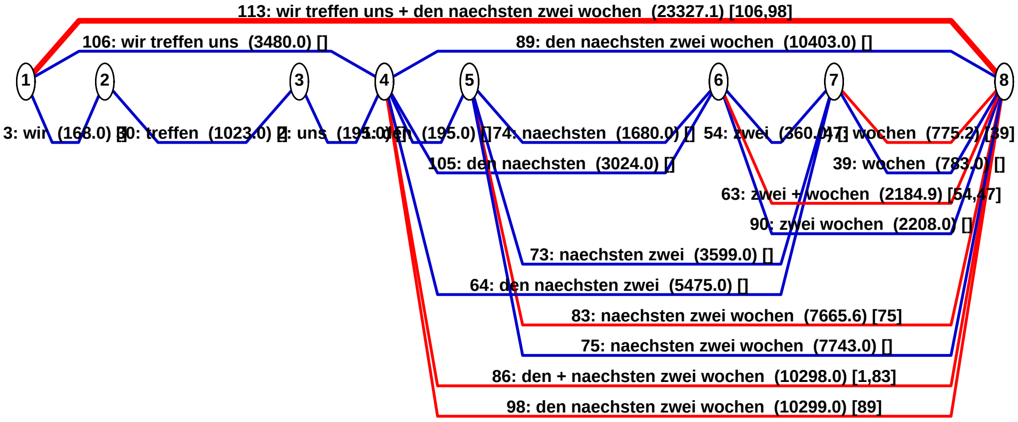
113: wir treffen uns + den naechsten zwei wochen (23327.1) [106,98]
106: wir treffen uns (3480.0) []
89: den naechsten zwei wochen (10403.0) []
1
2
3
4
5
6
7
8
3: wir (168.0) []
30: treffen (1023.0) []
2: uns (195.0) []
1: den (195.0) []
74: naechsten (1680.0) []
54: zwei (360.0) []
47: wochen (775.2) [39]
105: den naechsten (3024.0) []
39: wochen (783.0) []
63: zwei + wochen (2184.9) [54,47]
90: zwei wochen (2208.0) []
73: naechsten zwei (3599.0) []
64: den naechsten zwei (5475.0) []
83: naechsten zwei wochen (7665.6) [75]
75: naechsten zwei wochen (7743.0) []
86: den + naechsten zwei wochen (10298.0) [1,83]
98: den naechsten zwei wochen (10299.0) [89]

Figure 2: The VIT hypotheses graph for *Wir treffen uns (in) den nächsten zwei Wochen*.

from another edge (their list of components is empty: [ ]), while edge 98 has been built by robust semantic processing from edge 89 by applying the type raising rule mentioned above. Edge 113 results from applying this modifier to the proposition associated with edge 106. This edge is selected as the result.

The processing of the VHG is agenda-based. This allows us to give preference to analyses which span a larger portion of the input and/or which have been produced by a parser (as opposed to those produced by robust semantic processing). Since the parsers tend to produce analyses for smaller parts of a WHG path before they deliver analyses for larger chunks, these smaller bits are only considered as long as no larger analyses have been delivered. E. g., the parser first found analysis for the pronoun *wir* (edge 3) as an NP before it delivered a result for the sentence *wir treffen uns* (edge 106).

In addition to selection of a resulting VIT sequence as described in 5, the robust semantic processing component must determine when it is appropriate to make such a selection. Without external constraints this decision would simply amount to determining when all the parsers have delivered the information they have, either a spanning analysis or a clear indication of parse failure. However, the time constraints of the system as a whole require a more flexible strategy, since there is a sliding scale of global parameters determining intended performance.

The default strategy is to wait for the most detailed analysis which would come from the HPSG parser and would necessarily be a single analysis spanning the whole segment input. A more efficient option would be to take the first parser that claims to have processed up to the end of the segment as a cue to retrieve the best available analysis from the VHG. The results from the statistical or chunk parsers may, actually, consist of a sequence of grammatical fragments.

As the VHG also has anytime properties it would be conceivable to apply an absolute time limit, or one relative to the length of the input, but that would provide no guarantee that any single parser has processed the input and, hence could lead to too great a degradation in output quality. The objective here is to get the best out of the available resources under time constraints.

5. SELECTION OF RESULTS

The way the VIT hypotheses are combined to VIT strings covering whole utterances is a systematic adaption of methods known from lattice parsing. Like word hypotheses, VIT hypotheses have start and ending points, scores and symbolic contents. As we have learned from many approaches on word lattice parsing like [13], [12] and others, hybrid stochastic-symbolic approaches, like [3], perform well for those problems.

In search of a good spanning sequence of VITs we select VITs to combine on the basis of a stochastic model on VITs and combine the VITs themselves using symbolic rules. The main differences with respect to word lattice parsing are two properties of VIT hypotheses. Unlike word hypotheses which are produced by one decoder VITs come from four different decoding processes, whose internal scores are hardly comparable. Actually only two of those processes use probabilistic models although there is, in principle, no problem with enriching the remaining models — HPSG and Chunk Parsing — with derivation probabilities. The acoustic scores assigned to the word hypotheses, which belong to one VIT, turned out to be of little help, since in many cases competing VITs cover the same sequence of words. In order to have some empirical source of information we designed a special VIT-N-Gramm describing the probability of VIT sequences. It is used in combination with some heuristics preferring longer VITs which are more likely to represent a correct analysis. In addition, we give increasing penalties to the less precise models. The maximization formula is roughly (neglecting some details) as:

$$\mathrm{V} = \max_{\mathrm{V}_0^n}\left[\sum_{0\leq i\leq n} \mathit{LogP}(\mathrm{V}_i) + L(\mathrm{V}_i) + W(\mathrm{V}_i)\right]$$

where L stands for a length penalty and W for a penalty for certain sources (parsing methods). In first tests some empirically determined length and source weights led to acceptable results. In the future, it is planned to adjust the weights using optimization procedures on ideal outputs.

```
vit( vitID(sid(114,a,ge,25,66,1,ge,y,semantics),  % Segment ID
           [word(wir,r6,[l154]),                  % WHG String
            word(kommen,r7,[l105])]),
     index(l159,l106,i105),                       % Index
     [decl_imp_quest(l160,h103),                  % Conditions
      kommen(l105,i105),
      pron(l154,i106),
      arg1(l105,i105,i106)],
     [in_g(l160,l159),                            % Constraints
      in_g(l105,l106),
      leq(l106,h103),
      in_g(l154,l106)],
     [s_sort(i106,human),                         % Sorts
      s_sort(i105,move_sit)],
     [prontype(i106,sp_he,std)],                  % Discourse
     [num(i106,pl),                               % Syntax
      pers(i106,1)],
     [ta_mood(i105,ind),                          % Tense and Aspect
      ta_perf(i105,nonperf),
      ta_tense(i105,pres)],
     []                                           % Prosody
   )
```

Figure 3: An example of a VIT for the utterance *Wir kommen.*

6. CONCLUSION

We have presented an architecture which defines the mapping from word hypotheses to semantic representations. The architecture is fully implemented in the forthcoming version of the Verbmobil speech translation system, but not yet well tested. The key strategy is to apply multiple parsing strategies in parallel on preprocessed portions of the input and producing a new lattice per utterance where the atomic units are semantic descriptions. This architecture delays the search and decision for the final recognition result until the semantic processing level. This leads to a more robust overall behaviour and brings an effect of dynamically changing the “depth” of analysis.

7. ACKNOWLEDGEMENTS

The authors wish to thank their colleagues in the Verbmobil project for their cooperation.

This research was supported by the German Federal Ministry for Education, Science, Research and Technology (BMBF) under grant numbers 01 IV 701 R4, 01 IV 701 A3 and 01 IV 701 V0.

8. REFERENCES

1. S. Abney. Parsing by chunks. In Tenny Berwick, Abney, editor, *Principle–Based Parsing*. Kluwer, 1991.
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. towards a syntactic–prosodic labeling system for large spontaneous speech data bases. Verbmobil Memo 102, University of Erlangen–Nürnberg, 1996.
3. Rens Bod. *Enriching Linguistics with Statistics: Performance Models of Natural Language.* PhD thesis, Universiteit van Amsterdam, Department of Computational Linguistics, 1995.
4. Johan Bos, Bianka Buschbeck-Wolf, Michael Dorna, and C. J. Rupp. Managing information at linguistic interfaces. In *Proc. of the 17th COLING/36th ACL*, Montréal, Canada, 1998.
5. Johan Bos, Björn Gambäck, Christian Lieske, Yoshiki Mori, Manfred Pinkal, and Karsten Worm. Compositional semantics in Verbmobil. In *Proc. of the 16th COLING*, pages 131–136, Copenhagen, Denmark, 1996.
6. Michael Dorna and Martin C. Emele. Semantic-based transfer. In *Proc. of the 16th COLING*, pages 316–321, Copenhagen, Denmark, 1996.
7. R. Kompe, A. Kießling, H. Niemann, E. Nöth, A. Batliner, S. Schachtl, T. Ruland, and H. U. Block. Improving Parsing of Spontaneous Speech with the Help of Prosodic Boundaries. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 75–78, München, Germany, 1997. IEEE Signal Processing Society.
8. Carl Pollard and Ivan A. Sag. *Head-Driven Phrase Structure Grammar*. CSLI, Stanford, CA, and The University of Chicago Press, Chicago, London, 1994.
9. N. Reithinger and M. Klesen. Dialog act classification using language models. In *Proc. Eurospeech*, pages 2235–2238, 1997.
10. Tobias Ruland. Integrated linguistic processing. Verbmobil Technical Document 63, Siemens AG, 1997.
11. Wolfgang Wahlster. Verbmobil: Erkennung, Analyse, Transfer, Generierung und Synthese von Spontansprache. Verbmobil-Report 198, DFKI GmbH, Saarbrücken, June 1997. Available from http://www.dfki.de/verbmobil/.
12. H. Weber, J. Spilker, and G. Görz. Parsing n best trees from a word lattice. In G. Brewka, C. Habel, and B. Nebel, editors, *KI–97: Advances in Artificial Intelligence*. Springer, 1997.
13. Hans Weber. Time-synchronous chart parsing of speech integrating unification grammars with statistics. In *Proceedings of Twente Workshop on Speech and Language Engeneering*, pages 107–120, 1994.
14. Karsten L. Worm. A model for robust processing of spontaneous speech by integrating viable fragments. In *Proc. of the 17th COLING/36th ACL*, Montréal, Canada, 1998.
15. Karsten L. Worm and C. J. Rupp. Towards robust understanding of speech by combination of partial analyses. In *Proc. of the 13 th ECAI*, pages 190–194, Brighton, UK, 1998.