# DETECTION AND CORRECTION OF SPEECH REPAIRS IN WORD LATTICES

Jörg Spilker, Hans Weber, Günther Görz
University of Erlangen-Nuremberg
spilker,weber,goerz@immd8.informatik.uni-erlangen.de
http://www.informatik.uni-erlangen.de

10.21437/Eurospeech.1999-507

## ABSTRACT

Speech repairs occur often in spontaneous spoken dialogues. The ability to detect and correct those repairs is necessary for every spoken language system. We present a framework to detect and correct speech repairs where all relevant levels of information, i.e., acoustics, lexic, syntax and semantics could be integrated. The basic idea is to reduce the search space for repairs as soon as possible by cascading filters that involve more and more features. At first an acoustic module generates hypotheses about the existence of a repair. Second a stochastic model suggests a correction for every hypothesis. Well scored corrections are inserted as new paths in the word lattice. A lattice parser then makes the final decision about accepting the repair.

## 1. INTRODUCTION

Human utterances often contain erroneous parts where speakers correct previous words in their speech. The erroneous portions are not part of the intended word sequence. A speech system must be able to deal with such phenomena – *self corrections* or *repairs* – in order to extract the correct information. Therefore it must detect the existence of a repair and find the appropriate correction. According to a widely accepted topology of self corrections the latter consist of the following parts:

- Reparandum, the erroneous part
- Interruption Point (IP boundary), the time immediately after the reparandum
- Editing Term, a word sequence or filled pause acting as a repair indicator
- Reparans, the word sequence which replaces the reparandum

This means that at least two parts, reparandum and editing term, have to be identified to get the intended word sequence. An example for a segmentation taken from the German VERBMOBIL corpus is shown in fig. 1.

Not all of the slots described have to be filled. In the VERBMOBIL corpus[1] about appointment scheduling, repairs are usually not marked with an editing term (see section 3.). From a formal perspective each non repair word boundary could be seen as a case of an empty reparandum and empty edit term[2]. This case is not of much interest in our approach. Still hesitations (filled pauses) without a self correction will be characterized as edit terms with empty reparanda.

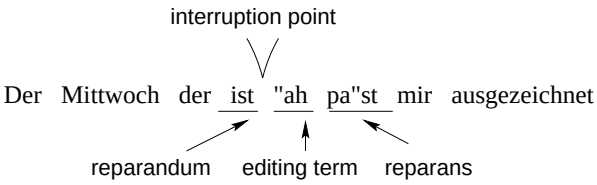



Figure 1: Segments of repairs

If the reparandum is not empty one can distinguish between *modification repairs* and *fresh starts*. Fresh starts are characterized by a disrupted sentence structure. The speaker aborts the current sentence and begins a new one, which means the scope of the reparandum will link back to the start of the utterance. Within modification repairs the speaker corrects only a part of the current sentence.[3]

## 2. RELATED WORK

A lot of work has been done to explore features for detecting the existence of repairs. Four classes of features are identified to be relevant for repair detection:

- acoustic/prosodic cues
- word fragments
- editing terms
- syntactic/semantic anomalies

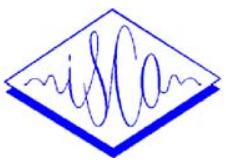
Until now most approaches concentrate on one of the topics above. Nakatani/Hirschberg [6] and Shriberg et al. [8], for instance, investigate acoustic/prosodic cues. Bear et al.[2] use editing terms and pattern matching as triggers

[1]This work is part of the VERBMOBIL project and was funded by the German Federal Ministry for Research and Technology (BMBF) in the framework of the Verbmobil Project under Grant BMBF 01 IV 701 V0. The responsibility for the contents of this study lies with the authors.

[2]Taking up the terminology of Heeman [4], a repair with an empty reparandum is called *abridged repair*. From a psychological point of view a repair could in fact take place in the speakers mind only, perhaps indicated by a filled pause.

[3]If the reparans is empty in a modification repair, the repair consists exclusively of a deletion, e.g. *I do not know if you want to try ( a IP ) Saturday.*





6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

for repair occurrences. In a second step a parser verifies the guess with syntactic and semantic knowledge. They also point out the usefulness of acoustic cues. Hindle [5] extend the grammar to correct repairs but assumes that the interruption point has been identified. Tischer [9] suggests some syntactic rules to identify repairs in the VERBMOBIL corpus, completely neglecting the problem of false alarms.

All the work mentioned above applies recognition experiments on transcribed speech and it is sometimes not obvious how it could be transferred to word lattices. An integrated approach is given by Heeman and Allen [4]. It extends the word recognition problem not only to identify the best word sequence, but also to determine the best corresponding POS-tag sequence with special repair tags. They incorporate several different knowledge sources including pauses, discourse markers and editing terms. Since the approach drives a speech recognizer by a special language model, there is no need for a special adaptation to word lattices. In contrast our approach is a three level architecture where a wider variety of features can be incorporated. During word recognition a language model trained on the original corpus including repairs is used. Using prosodic–acoustic features all word boundaries are classified into IP vs. non–IP boundaries. Thus the search space in the resulting attributed word lattice becomes small enough to apply a proper classification of the possible reparandum and reparans scopes. The approach is capable to incorporate all cue features mentioned and has already been implemented and tested in the VERBMOBIL prototype in fall 1998. First, we will give a brief introduction to the corpus used. Then an overview of the architecture follows before we enter into a detailed discussion of the three levels. Finally, the results are presented together with suggestions for a further improvement of the current system.

# 3. A CORPUS ANALYSIS

The VERBMOBIL corpus is a selection of human human dialogues about appointment scheduling and travel planning. For each dialogue two people were instructed to plan a business trip or date. In this scenario 12860 turns were recorded. Thereof 1737 turns were left out for evaluation. Among the remaining 11123 turns there are 2345 turns (21.1%) which contain at least one repair. Totally 3209 repairs take place in the reduced corpus. The minor part of them consists of fresh starts (624) whereas the majority (2585) are modification repairs. As mentioned in the introduction, most repairs are not marked by an editing term only 26.3% of all modification repairs contain an editing term. Thus it is not sufficient to concentrate exclusively on editing terms as repair indicators. Just as editing terms word fragments can help to detect repairs. 46.3% of all reparanda end in a fragment. A statistical evaluation about the repair length shows that 98% of all reparanda and reparans are shorter than five words. This corresponds to the fact that most repairs take place within a syntactic phrase. Investigating a different German corpus for example [10] found out that over 70% of all repairs are realized in the range of NPs and PPs. Considering the Part-of-Speech (POS) categories one realizes that in most cases a word is replaced by another one with the same POS-category. Using coarse categories[4] 90% of all replacements are POS matches. Even the frequency of word deletions and insertions depends on their POS-categories. Verbs for example are seldom deleted (11%) or inserted (8%), whereas adverbs have a high insertion (27%) and deletion (28%) frequency. All these structural constraints can be used to distinguish repairs from fluent speech.

# 4. THE ARCHITECTURE

For the integration of repair processing in a speech system several additional aspects have to be considered. A common interface of speech recognizers is the word lattice. It represents a big search space which cannot be fully explored by time consuming methods like parsers. Speech repairs add another dimension to this search space. On the other hand, syntactic knowledge plays an important role in repair detection. To integrate all introduced repair features, we suggest a step-by-step reduction of the search space starting with quickly determinable features and ending in complex ones. Figure 2 shows the corresponding three level architecture.

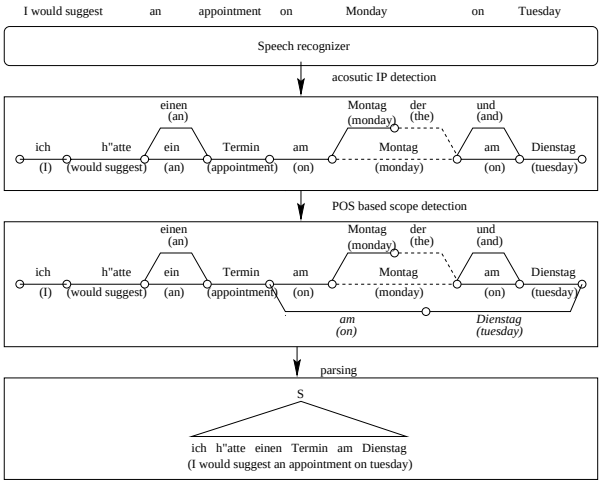



Figure 2: An architecture for detecting and correcting repairs.

In a first step a search for possible repairs is triggered. This is actually done by acoustic cues but can easily extended to other triggers like editing terms or simple pattern matching. Every edge in the word lattice is prosodically annotated with a hypothesis about an interruption point. These IP-boundary hypotheses are marked by dashed lines in the figure. Second a stochastic model, the "scope" model, tries to detect reparandum and reparans for each IP-hypothesis. If the repair score exceeds a certain threshold, an alternative path representing the word sequence of the reparans is inserted in the lattice. In our example only the paths through *am Montag am Dienstag* (*on Monday on Tuesday*) contain a plausible repair in which *am Montag* is the reparandum and *am Dienstag* is the reparans. So, a new path *am Dienstag* reaching over the complete repair is inserted[5]. A lattice parser then searches for the

[4] 20 categories

[5] Since elimination of the reparandum is not always appropriate, for

best grammatical path and hence selects the repair edge and gets the intended meaning. No repair meta rules are necessary for the parser except that it has a special control for pure hesitations (abridged repairs).

# 5. THE ACOUSTIC LEVEL

The prosodic detection[6] is done by pure acoustic features, which means that no language model is involved. For training all IPs are marked and for each IP a feature vector with over 200 features is calculated. It serves as an input for a multi -layer perceptron to determine the probability **IP-boundary** versus **non-IP-boundary**. Due to the small amount of data we do not separate word fragments and others for training. For a detailed description of the acoustic aspects see [1].

# 6. SCOPE DETECTION

The scope detection is divided into two parts. First, we will describe how the scope model determines the segments of a repair when a path is given. Second, we present an algorithm to integrate this model in lattice processing. A general assumption for this work is that neither reparandum nor reparans exceeds the length of four words[7]. This enables us to enumerate all possible reparandum(RD)/reparans(RS) pairs. Starting at an IP-boundary each pair is scored with $P(RD|RS)$. In our first approach editing terms are seen as a set of constant terms. If right of an IP-boundary a word sequence follows that is known to serve as an editing term it will be skipped. $P(RD|RS)$ determines the likelihood for a replacement of the reparandum by the reparans. It is based on the same idea used in statistical machine translation [3]. The likelihood involves the following three components:

- the length of reparandum and reparans (length model)
- the POS-tags in reparandum and reparans (replacement model)
- an alignment between the POS-tags of reparandum and reparans (position model)

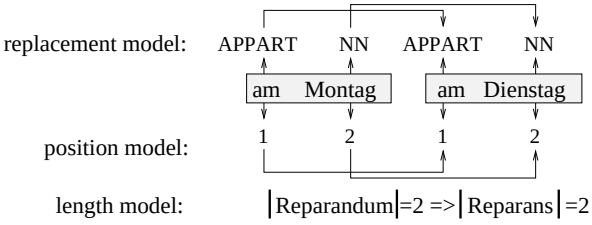



Figure 3: An example for scoring the scope

Therefore $P(RD|RS)$ could be written as

$$P(RD|RS) = P(len(RD)|len(RS)) * \sum_{a \in A} P(RD|a, RS)$$

where $A$ is the set of possible alignment functions.

$$ml = \max_{j=ml}(len(RD), len(RS))$$

$$P(RD|a, RS) = \prod_{j=1} P(a(j)|j, len(RD), len(RS)) * P(POS(RD_j)|POS(RS_{a(j)}))$$

$POS(RD_j)$ is the POS-category of the j-th word in the reparandum. At last the maximum probability of all reparandum/reparans pairs selects the best scope hypothesis.

Integrating this model in a lattice algorithm requires three steps:

- mapping the word lattice onto a tag lattice,
- selecting IP-boundaries and extracting the possible reparandum/reparans pairs,
- introducing new paths to represent the plausible reparans.

The tag graph construction is adapted from Samuelsson [7]. For every word edge and every denoted POS-tag a corresponding tag edge is created and the resulting probability is determined. If a tag edge already exists, the probabilities of both edges are merged. The original words are stored in a list associated with the tag edge. IP-boundary hypotheses are assigned to tag edges by choosing the maximum probability of the corresponding word edges.

If an IP-boundary hypothesis is above a threshold it triggers the search for a repair scope. Paths through the tag graph are scored by a POS-trigram. Using the $A^*$-algorithm all paths through an IP-boundary could be enumerated stepwise. Since the scope model takes only four words for reparandum and reparans in account it is sufficient to expand only partial paths. Each of these partial paths is then processed by the scope model. To reduce the search space, paths with a low score could be pruned.

The resulting scope hypotheses will lead to new paths in the tag graph. To ensure that these new paths are comparable to other paths we score the reparandum the same way the parser does and add this value to the first word of the reparans. As a result the original path and the one with the repair got the same score except one word transition. The (probably bad) transition in the original path from the last word of the reparandum to the first word of the reparans is replaced by a (probably good) transition from the reparandums onset to the reparans.

# 7. RESULTS & CONCLUSION

The evaluation takes place according to the three levels. In the first step we consider only the acoustic level. Because state-of-the-art speech recognizers can not hypothesize fragments, we perform three tests, one with the complete test corpus, one with a corpus containing no fragments and one with a corpus containing only repairs with no fragments. To eliminate the influence of misrecognized words, all tests are performed on transcribed speech. Table 1 shows the results. The low precision comes from to basic problems. The first one is a sparse data problem. IP-boundaries are very rare in contrast to non-IP-boundaries. The second one is demonstrated by table 2. For a small subset of our test corpus

instance if a pronoun refers to an object introduced by the reparandum (anaphoric reference), the reparandum is kept in a separate storage.

[6]All work described in this section is done by the speech group of the IMMD 5. Special thanks to Anton Batliner, Volker Warnke and Richard Huber.

[7]c.f. section 3.

| | Recall | Precision |
|---|---|---|
| complete | 76% | 3% |
| only IPs with fragment | 82% | 3% |
| only other IPs | 67% | 2% |

Table 1: Results for acoustic detection

prosodic-perceptual boundaries have been annotated. One boundary class are irregular *prosodic* boundaries (B9), i.e prosodically marked boundaries, which do not correspond with regular phrase boundaries. These boundaries are mostly marked by hesitations and so there is not necessarily a correspondence between these B9-boundaries and IP-boundaries. However table 2 shows that the major part of IP-boundaries are located at B9-boundaries. On the other hand most B9-boundaries do not signal an IP. This means that there is a great chance to detect a lot of B9 boundaries, if you try to detect IP-boundaries with pure acoustic features.

| | | IP-Boundary | | | Total |
|---|---|---|---|---|---|
| | | no-IP | IP + frag. | other | |
| perceptive boundaries | B9 | 569 | 18 | 61 | 648 |
| | other | 13990 | 16 | 12 | 14018 |
| Total | | 14559 | 34 | 73 | 14666 |

Table 2: Correlation Matrix perceptive - IP boundaries

The second evaluation determines the quality of the scope model. Therefore we assume a 100% correct recognition of IP-boundaries. For 70.7% of all repairs, reparandum and reparans were identified correctly[8]. In 84.5% of all cases at least the reparandum is correct. This is the relevant rate, because a correct reparandum detection is necessary for correcting the utterance. Since we restrict reparandum and reparans lengths to four words, some repairs could not be detected by the scope model. When filtering these repairs, the reparandum detection increases to 86.9%. The last step is the test of the complete system. The parser was simulated by a word trigram because no parser was available at evaluation time. We were able to detect 50% of all repairs with a correction rate of 91%. This makes a recall rate of 45% for correct repairs. The precision is 17%. The results are fairly low compared to the rates reported by other groups [4, 6, 2]. However a realistic comparison is rather difficult according to very different evaluation conditions. Of course the low results are related to the fact that a trigram has no linguistic knowledge. Another reason for false alarms is the deletion of function words. Such a deletion results in a correct sentence but seldom modifies the semantics of an utterance. We found that many false alarms could be avoided by integrating semantic knowledge. A replacement, for example, of *Monday* by *train* is less likely than by *Tuesday*. But all these words are nouns, so actually they get the same score. Another aspect is the integration of phrase boundaries, which are detected by another prosodic model. This could help to prevent the scope model from suggesting repairs across sentence boundaries.

[8] Editing terms are harmless here because they can be eliminated after a lookup in the constant editing term list.

## REFERENCES


[1] A. Batliner, R. Kompe, A. Kießling, M. Mast, H. Niemann, and E. Nöth. M = syntax + prosody: A syntactic-prosodic labelling schema for large spontaneous speech databases. *Speech Communication*, 25:193–22, 1998.

[2] J. Bear, J. Dowding, and E. Shriberg. Integrating multiple knowledge sources for detection and correction of repairs in human computer dialogs. In *Proceedings of the ACL*, pages 56–63, University of Delaware, Newark, Delaware, 1992.

[3] Peter F. Brown, Stehpen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical maschine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, June 1993.

[4] P. Heeman and J. Allen. Intonational boundaries, speech repairs and discourse markers: Modeling spoken dialog. In *Proceedings of the ACL*, Universidad Nacional de Educacion a Distancia (UNED), Madrid, Spain,, 1997.

[5] D. Hindle. Deterministic parsing of syntactic nonfluencies. In *Procecdings of the ACL*, MIT, Cambridge, Massachusetts, 1983.

[6] C. Nakantani and J. Hirschberg. A speech-first model for repair detection and correction. In *Proceedings of the ACL*, Ohio State University, Columbus, Ohio, 1993.

[7] C. Samulesson. A left-to-right tagger for word graphs. In *Proceedings of the 5th International workshop on Parsing technologies*, pages 171–178, Bosten, Massachusetts, 1997.

[8] Elizabeth Shriberg, Rebecca Bates, and Andreas Stolcke. A prosody-only decision-tree model for disfluency detection. In *Proc. EUROSPEECH '97*, volume 5, pages 2383–2386, Rhodes, Greece, September 1997.

[9] B. Tischer. Syntactic procedures for the detection of self–repairs in German dialogues. In W. Wahlster, editor, *ECAI 1996. 12th European Conference on Artificial Intelligence*, pages 79–82. John Wiley, 1996.

[10] Shu-Chuan Tseng. A linguistic analysis of repair signals in co-operative spoken dialogs. In *Proc. of the ISCLP*, 1998.