

Chapter 5

IMPROVING ROBUSTNESS BY MODELING SPONTANEOUS SPEECH EVENTS

Peter A. Heeman

Oregon Graduate Institute

heeman@cse.ogi.edu

James F. Allen

University of Rochester

james@cs.rochester.edu

Abstract In spontaneous speech, speakers segment their speech into intonational phrases, and make repairs to what they are saying. However, techniques for understanding spontaneous speech tend to treat these events as noise, in the same manner as they handle out-of-grammar constructions and misrecognitions. In our approach, we advocate that these events should be explicitly modeled, and that they must be resolved early in the processing stream. We put forward a statistical language model, which can be used during speech recognition, that models these events. This not only improves speech recognition perplexity and POS tagging, but also results in much richer output from the recognizer, with speech repairs resolved and intonational phrase boundaries identified. Syntactic and semantic processing can thus focus on dealing with out-of-grammar constructions and misrecognitions.

5.1 Introduction

To enable spoken dialogue systems to advance towards more collaborative interaction between humans and computers, we need to deal with language as it is actually spoken. In natural speech, speakers group words into intonational phrases and make repairs to what they are saying. Consider the following speaker's turn from the Trains corpus (Heeman and Allen 1995).

Example 1 (d93-13.3 utt63)

um it'll be there it'll get to Dansville at three a.m. and then you wanna do you take tho- want to take those back to Elmira so engine E two with three boxcars will be back in Elmira at six a.m. is that what you wanna do

From reading the word transcription, the reader should immediately notice the prevalence of *speech repairs*, where speakers go back and change or repeat something they just said. Fortunately for hearers, speech repairs tend to have a standard form. The *reparandum* is the stretch of speech that the speaker is replacing; it might end in the middle of a word, resulting in a word fragment. The end of the reparandum is called the *interruption point*. There can also be an editing term, consisting of filler words, such as 'uh' and 'um', or cue phrases, such as 'let's see', 'well', and 'okay'. This is then followed by the *alteration*, which is the replacement for the reparandum. Below, we illustrate this analysis on the first repair from the above speaker turn.

Example 2 (Repair)

um it'll be there [↑] it'll get to Dansville at three a.m.
 reparandum ip alteration

Speech repairs are very prevalent in spontaneous speech. In the Trains corpus, 10% of all words are part of the editing term or reparandum of a speech repair, and 54% of all speaker turns with at least 10 words have at least one repair. To determine the speaker's intended message, speech repairs need to be *resolved*: they need to be *detected*, by finding their interruption point, and *corrected*, by determining the extents of the reparanda and editing terms.¹

In addition to making repairs, speakers also break their turn of speaking into intonational phrases, which are signaled through variations in the pitch contour, segmental lengthening and pauses. Previous research has shown that intonational information can reduce syntactic ambiguity for humans (Beach 1991) and in computer parsers (Bear and Price 1990, Ostendorf, Wightman and Veilleux 1993). Other researchers have proposed segmenting speech into speech acts (i.e. Mast, Kompe, Harbeck, Kießling, Niemann, Nöth, Schukat-Talamazzini and Warnke 1996) or linguistically, based on having a single clause (i.e. Meteer and Iyer 1996). However, there is no clear consensus as to the right approach. Although intonational phrases might not be the ideal unit for modeling interaction in dialogue, it definitely captures the speaker's intention and will undoubtedly be a major component of any definition (Traum and Heeman 1997).

1. The reparanda and editing terms, however, should not simply be expunged from the speech, as it might contain valuable information, such as the identify of an anaphoric reference: "Peter was ... well he was fired".

Now that we have introduced the spontaneous speech events, we show our example annotated in terms of them. Repair reparanda are indicated in *italic*, with the alteration starting on a new line indented to start at the reparandum onset. Intonational phrase boundaries are marked with ‘%’.

Example 3 (d93-13.3 utt63)

um it'll be there
 it'll get to Dansville at three a.m. %
 and then you wanna
 do you *take tho-*
 want to take those back to Elmira %
 so engine E two with three boxcars will be back in Elmira at six a.m. %
 is that what you wanna do %

Although the spontaneous speech events of speech repairs and intonational phrasing are much more common in human-human speech than in human-computer speech (Oviatt 1995), this will change as people become increasingly more comfortable with human-computer interaction and start focusing on the task before them, rather than on the form of their interaction (Price 1997). Hence, spoken language systems will increasingly need to deal with these events.

5.1.1 Robust Parsing Approach

One line of research that has become popular for dealing with speech repairs is to use robust parsing techniques. For understanding spontaneous speech, speech repairs are not the only phenomena that create problems; one also needs to deal with word misrecognitions and out-of-grammar constructions. All three of these problems tend to be lumped together and given to a robust parser. Ward (1991) used a robust semantic parser to look for sequences of words that matched grammar fragments associated with slots of case frames. The parser would try to fill as many slots as possible. If a slot is only partially filled, it is abandoned. If a slot is filled more than once, the latter value is taken (Young and Matessa 1991). In this volume, ?) describe using a robust parser, which incorporated a skipping mechanism, with a feature unification grammar; and ?) describes using a skipping mechanism in parsing word graphs.

5.1.2 Modeling Spontaneous Speech Events

Rather than view spontaneous speech events as noise in the input to a robust parser, we advocate that speech repairs and intonational phrasing should be explicitly modeled. There are local cues, such as editing terms, word correspondences, pauses, that give evidence for these events. Hence, we should be able to automatically identify the intonational phrases and resolve the speech repairs. By modeling these events, we will have a richer understanding of the speech.

This will simplify later syntactic and semantic processing, since such processing can start from enriched output rather than trying to cope with the apparent ill-formedness that spontaneous speech events cause. This will also make it easier for these processes to deal with the other problems of understanding spontaneous speech: namely misrecognitions and out-of-grammar constructions.

Speech repairs and intonational phrasing are intertwined with the speech recognition problem of predicting the next word given the previous context (Heeman and Allen 1999). Hence, our approach is to redefine the speech recognition problem so that it includes the resolution of speech repairs and identification of intonational phrases. We also include the tasks of part-of-speech (POS) tagging and discourse marker identification, since these tasks are also intertwined with resolving speech repairs and identifying intonational phrasing. Since all tasks are being resolved in the same model, we can account for the interactions between the tasks in a framework that can compare alternative hypotheses for the speakers' turn. Not only does this allow us to model the spontaneous speech events, but it also results in an improved language model, evidenced by both improved POS tagging and better probability estimates of the next word. Furthermore, speech repairs and phrase boundaries have acoustic correlates, such as pauses between words. By resolving speech repairs and identifying intonational phrases during speech recognition, these acoustic cues, which otherwise would be treated as noise, can give evidence as to the occurrence of these events, and further improve speech recognition results.

5.1.3 Overview of the Chapter

We next describe the Trains corpus and the annotation of speech repairs and intonational phrases. We then introduce our baseline language model, which incorporates POS tagging and discourse marker identification, and we introduce the machine learning techniques we use for estimating the probability distributions. We then augment our baseline model with speech repair and intonational phrase detection and speech repair correction, and give a sample run of the model. We then give the results of running our model on the Trains corpus, and compare our work with previous work in modeling speech repairs and intonational phrasing. Finally, we present the conclusions and future work.

5.2 The Trains Corpus

For our research work, we used the Trains corpus, a corpus of human-human task-oriented dialogs available from the Linguistics Data Consortium. The corpus consists of six and a half hours of speech produced by 34 different speakers solving 20 different problems. Each word was transcribed using its orthographic spelling, unless it was mispronounced and the speaker subsequently repairs the mispronunciation. Contractions, including words such as 'wanna',

reparandum being repeated by the alteration.

The third type are the abridged repairs. These repairs consist of an editing term, but with no reparandum, as the following example illustrates.

Example 6 (d93-14.3 utt42)

we need to $\overset{\uparrow}{ip}$ \underbrace{um} manage to get the bananas to Dansville more quickly

editing term

For these repairs, the hearer has to determine that an editing term occurred, which can be difficult for phrases such as ‘let’s see’ or ‘well’ since they can also have a sentential interpretation. The hearer also has to determine that the reparandum is empty. As the example above illustrates, this is not necessarily a trivial task because of the spurious word correspondences between ‘need to’ and ‘manage to’. In the Trains corpus, there are 423 abridged repairs, 1302 modification repairs, 671 fresh starts.

5.3 POS-Based Language Model

In this section, we present a speech recognition language model that incorporates POS tagging. Here, POS tags are viewed as part of the output of the speech recognizer rather than as intermediate objects. Not only is this syntactic information needed for modeling the occurrence of speech repairs and intonational phrases, but it will also be useful for higher level syntactic and semantic processes. Incorporating POS tagging can also be seen as a first step in tightening the coupling between speech recognition and natural language processing so as to be able to make use of richer knowledge of natural language than simple word-based language models provide.

5.3.1 Word-based Language Models

The goal of speech recognition is to find the most probable sequence of words \hat{W} given the acoustic signal A (Jelinek 1985).

$$\begin{aligned}\hat{W} &= \arg \max_W \Pr(W|A) \\ &= \arg \max_W \Pr(A|W) \Pr(W)\end{aligned}\tag{5.1}$$

The first term, $\Pr(A|W)$, is the *acoustic model* and the second term, $\Pr(W)$, is the *language model*. We rewrite W explicitly as the sequence of words $W_1W_2W_3\dots W_N$, where N is the number of words in the sequence. For expository ease, we use $W_{i,j}$ to refer to $W_i\dots W_j$. We now rewrite $\Pr(W_{1,N})$ as follows.

$$\Pr(W_{1,N}) = \prod_{i=1}^N \Pr(W_i|W_{1,i-1})\tag{5.2}$$

The above equation gives us the probability of the word sequence as the product of the probability of each word given its previous lexical context.

5.3.2 Incorporating POS Tags

To incorporate POS tags into the language model, we redefine the speech recognition problem so as to include finding the best POS and discourse marker sequence along with the best word sequence. For the word sequence W , let P be a POS sequence. The goal of the speech recognition process is to now solve the following.

$$\begin{aligned}\hat{W}\hat{P} &= \arg \max_{WP} \Pr(WP|A) \\ &= \arg \max_{WP} \Pr(A|WP) \Pr(WP)\end{aligned}\quad (5.3)$$

The first term $\Pr(A|WP)$ is the acoustic model, which can be approximated by $\Pr(A|W)$. The second term $\Pr(WP)$ is the POS-based language model and accounts for both the sequence of words and their POS assignment. We rewrite this term as follows.

$$\begin{aligned}\Pr(W_{1,N}P_{1,N}) &= \prod_{i=1}^N \Pr(W_i P_i | W_{1,i-1} P_{1,i-1}) \\ &= \prod_{i=1}^N \Pr(W_i | W_{1,i-1} P_{1,i}) \Pr(P_i | W_{1,i-1} P_{1,i-1})\end{aligned}\quad (5.4)$$

Equation 5.4 involves two probability distributions that need to be estimated. To successfully use POS tags in a language model, we need to estimate these probability distributions as best possible.

5.3.3 Estimating the Probabilities

To estimate the probability distributions, we follow the approach of Bahl, Brown, de Souza and Mercer (1989) and use a decision tree learning algorithm (Breiman, Friedman, Olshen and Stone 1984) to partition the context into equivalence classes. The algorithm starts with a single node. It then finds a question to ask about the node in order to partition the node into two *leaves*, each more informative as to which event occurred than the parent node. Information theoretic metrics, such as minimizing entropy, are used to decide which question to propose. The proposed question is then verified using heldout data: if the split does not lead to a decrease in entropy according to the heldout data, the split is rejected and the node is not further explored. This process continues with the new leaves and results in a hierarchical partitioning of the context. After the tree is grown, relative frequencies are calculated for each node, and these probabilities are then interpolated with their parent node's probabilities using a second heldout dataset.

Using the decision tree algorithm to estimate probabilities is attractive since the algorithm can choose which parts of the context are relevant, and in what order. Hence, this approach lends itself more readily to allowing extra contextual information to be included, such as both the word identities and POS tags, and even hierarchical clusterings of them. If the extra information is not relevant, it will not be used. The approach of using decision trees will become even more critical in the next two sections where the probability distributions will be conditioned on an even richer context.

Questions about POS Tags

The context that we use for estimating the probabilities includes both word identities and POS tags. To make effective use of this information, we allow the decision tree algorithm to generalize between words and POS tags that behave similarly. To learn which ones behave similarly, Black, Jelinek, Lafferty, Magerman, Mercer and Roukos (1992) used the clustering algorithm of Brown, Della Pietra, deSouza, Lai and Mercer (1992) to build a hierarchical classification tree. The algorithm starts with each POS tag in a separate class and iteratively finds two classes to merge that results in the smallest loss of information about POS adjacency. This continues until only a single class remains. The order in which classes were merged, however, gives a binary tree with the root corresponding to the entire tagset, each leaf to a single POS tag, and intermediate nodes to groupings of the tags that are statistically similar. The path from the root to a tag gives the binary encoding for the tag. The decision tree algorithm can ask which partition a tag belongs to by asking questions about its binary encoding.

Questions about Word Identities

For handling word identities, one could follow the approach used for handling the POS tags and view the POS tags and word identities as two separate sources of information. Instead, we view the word identities as a further refinement of the POS tags (Heeman 1997). We start the clustering algorithm with a separate class for each combination of word and POS tag that exists in the training data. Classes are only merged if the POS tags are the same. The result is a word classification tree for each tag. This approach means that the trees will not be polluted by words that are ambiguous as to their tag, as exemplified by the word 'loads', which is used in the corpus as a third-person present tensed verb and as a plural noun. Furthermore, this approach simplifies the clustering task because the hand annotations of the POS tags resolve a lot of the difficulty that the algorithm would otherwise have to learn.

Other Questions

We allow two other types of information to be used as part of the context: numeric and categorical information. Although this type of information is not

used in this section, they will be used in the next two sections. For a numeric variable N , the decision tree searches for questions of the form ‘is $N \geq n$ ’, where n is a numeric constant. For a categorical variable C , it searches over questions of the form ‘is $C \in S$ ’ where S is a subset of the possible values of C . We also allow composite questions (Bahl et al. 1989), which are boolean combinations of elementary questions.

5.4 Identifying Speech Repairs and Intonational Phrases

In the previous section, we presented a POS-based language model. This model did not account for the occurrence of speech repairs nor intonational phrases. Ignoring these events when building a statistical language model leads to probabilistic estimates for the words and POS tags that are less precise, since they mix contexts that cross intonational boundaries and interruption points of speech repairs with *fluent* stretches of speech.

The problem with incorporating speech repair and intonational phrase detection into a language model is that there is not a reliable signal for detecting repairs (Bear, Dowding and Shriberg 1992) nor intonational phrases. Rather, there are a number of sources of information that give evidence as to the occurrence of these events. These sources include the presence of pauses, fillers, cue phrases, discourse markers, word fragments, word correspondences and syntactic anomalies. In this section, we augment our POS-based language model so that it also detects intonational phrases and speech repairs, along with their editing terms. To model the occurrence of intonational boundaries and speech repairs, we introduce three extra variables into the language model: the *repair* variable R_i , the *editing term* variable E_i and the *intonation* variable I_i . The probability distributions of the resulting model take into account most of the sources of evidence that signal spontaneous speech events.

5.4.1 Speech Repairs

The repair variable indicates the occurrence of speech repairs and its type: whether it is a modification repair, fresh start, or an abridged repair. The type of repair is important since the strategy that a hearer uses to correct a repair depends on the type of repair. For fresh starts, the hearer must determine the beginning of the current utterance. For modification repairs, the hearer can make use of the correspondences between the reparandum and alteration to determine the reparandum onset. For abridged repairs, there is no reparandum, and so simply knowing that it is abridged gives the correction.

For repairs that do not have an editing term, the interruption point is where the local context is disrupted, and hence is the logical place to tag such repairs. For repairs with an editing term, we tag the repair at the end of the editing term. This leads to the following definition of the repair variable R_i for

the transition between word W_{i-1} and W_i .

$$R_i = \begin{cases} \mathbf{Mod} & \text{if } W_i \text{ is the alteration onset of a modification repair} \\ \mathbf{Can} & \text{if } W_i \text{ is the alteration onset of a fresh start (or } \textit{cancel}) \\ \mathbf{Abr} & \text{if } W_i \text{ is the alteration onset of an abridged repair} \\ \mathbf{null} & \text{otherwise} \end{cases}$$

5.4.2 Editing Terms

Speech repairs often have an editing term, which follows the interruption point. Whether a word is being used as an editing term is not easy to determine. Phrases such as ‘let me see’ can be used as part of the sentential content of a sentence or as an editing term. Even fillers, such as ‘um’ and ‘uh’, only count as part of an editing term when they are not utterance initial. Hence, we need to model the occurrence of editing terms along with the occurrence of speech repairs. In our model, the variable E_i indicates the type of editing term transition between word W_{i-1} and W_i .

$$E_i = \begin{cases} \mathbf{Push} & \text{if } W_{i-1} \text{ is not part of an editing term but } W_i \text{ is} \\ \mathbf{ET} & \text{if } W_{i-1} \text{ and } W_i \text{ are both part of an editing term} \\ \mathbf{Pop} & \text{if } W_{i-1} \text{ is part of an editing term but } W_i \text{ is not} \\ \mathbf{null} & \text{otherwise} \end{cases}$$

Below, we give an example and show all non-null editing term and repair tags.

Example 7 (d93-10.4 utt30)

that’ll get there at four a.m. **Push** oh **ET** sorry **Pop Mod** at eleven a.m.

5.4.3 Intonational Phrases

The final variable is I_i , which marks the occurrence of intonational phrase boundaries.

$$I_i = \begin{cases} \% & \text{if } W_{i-1} \text{ ends an intonational phrase} \\ \mathbf{null} & \text{otherwise} \end{cases}$$

The intonation variable is separate from the editing term and repair variables since it is not restricted by the value of the other two. For instance, an editing term could end an intonational phrase, especially on the end of a cue phrase such as ‘let’s see’, as can the reparandum, as Example 8 below demonstrates.

Example 8 (d92a-2.1 utt29)

that’s the one with the bananas % **Push I ET** mean **Pop Mod** that’s taking the bananas

5.4.4 Redefining the Speech Recognition Problem

We now redefine the speech recognition problem so that its goal is to find the sequence of words and the corresponding POS, intonation, editing term and

repair tags that is most probable given the acoustic signal.

$$\begin{aligned}\hat{W}\hat{P}\hat{R}\hat{E}\hat{I} &= \arg \max_{WPREI} \Pr(WPREI|A) \\ &= \arg \max_{WPREI} \Pr(A|WPREI) \Pr(WPREI)\end{aligned}\quad (5.5)$$

The second term is the language model probability, and can be rewritten as follows.

$$\begin{aligned}\Pr(W_{1,N}P_{1,N}R_{1,N}E_{1,N}I_{1,N}) &= \prod_{i=1}^N \Pr(W_i P_i R_i E_i I_i | W_{1,i-1} P_{1,i-1} R_{1,i-1} E_{1,i-1} I_{1,i-1}) \\ &= \prod_{i=1}^N \Pr(I_i | W_{1,i-1} P_{1,i-1} R_{1,i-1} E_{1,i-1} I_{1,i-1}) \\ &\quad \Pr(E_i | W_{1,i-1} P_{1,i-1} R_{1,i-1} E_{1,i-1} I_{1,i-1}) \\ &\quad \Pr(R_i | W_{1,i-1} P_{1,i-1} R_{1,i-1} E_{1,i-1} I_{1,i-1}) \\ &\quad \Pr(P_i | W_{1,i-1} P_{1,i-1} R_{1,i-1} E_{1,i-1} I_{1,i-1}) \\ &\quad \Pr(W_i | W_{1,i-1} P_{1,i-1} R_{1,i-1} E_{1,i-1} I_{1,i-1})\end{aligned}\quad (5.6)$$

5.4.5 Representing the Context

Equation 5.6 requires five probability distributions to be estimated. The context for each includes all of the words, POS, intonation, repair and editing term tags that have been hypothesized, each as a separate piece of information. In principal, we could give all of this information to the decision tree algorithm and let it decide what information to use in constructing equivalence classes. However, repairs, editing terms and even intonation phrases do not occur in the same abundance as fluent speech and are not as constrained. Hence, it will be difficult to properly estimate the probability distributions.

Consider the following example of a speech repair without an editing term.

Example 9 (d92-1 utt53)

engine E two picks **Mod** takes the two boxcars

When predicting the first word of the alteration ‘takes’, it is inappropriate to ask about the preceding words, such as ‘picks’, without realizing that there is a modification repair in between. The same also holds for intonational boundaries and editing term pushes and pops. In the example below, a question should only be asked about ‘is’ in the realization that it ends an intonational phrase.

Example 10 (d92a-1.2 utt3)

you'll have to tell me what the problem is % I don't have their labels

Although the intonation, repair and editing term tags are part of the context and so can be used in partitioning it, the problem is that null intonation, repair and editing term tags dominate the training examples. So, we are bound to run into contexts in which there are not enough intonational phrases and repairs for the decision tree algorithm to learn the importance of using this information, and instead it might blindly subdivide the context based on some subdivision of the POS tags. The solution is analogous to what is done in POS tagging of written text: we give a view of the words and POS tags with the non-null repair, non-null intonation and editing term push and pop tags inserted. By inserting these tags into the word and POS sequence, it will be more difficult for the learning algorithm to ignore them.

Now consider the following examples, which both start with 'so we need to'.

Example 11 (d92a-2.2 utt6)

so we need to **Push** um **Pop** **Abr** get a tanker of OJ to Avon

Example 12 (d93-11.1 utt46)

so we need to get the three tankers

This is then followed by the verb 'get', except the first has an editing term in between. In predicting this word, the editing term hinders the decision tree algorithm from generalizing with non-abridged examples. The same thing happens with fresh starts and modification repairs. To allow generalizations between repairs with an editing term and those without, we use a view of the context with completed editing terms removed (cf. Stolcke and Shriberg 1996b).

To illustrate the augmented word and POS contexts given to the decision tree, consider the following example.

Example 13 (d93-18.1 utt47)

it takes one **Push** you **ET** know **Pop** **Mod** two hours %

For the correct interpretation for the POS tag of 'you', the context includes the previous words together with the tag that indicates we are starting an editing term: 'it/**PRP** takes/**VBP** one/**CD** **Push**.' The context for the editing term **Pop** is 'it/**PRP** takes/**VBP** one/**CD** **Push** you/**PRP** know/**VBP**.' The repair tag is predicted after the editing term is completed, and hence has the editing term cleaned up: 'it/**PRP** takes/**VBP** one/**CD**' (we also give it the context with the editing term not cleaned up). The context for the POS tag of 'two' is 'it/**PRP** takes/**VBP** one/**CD** **Mod**.'

We also include two variables that indicate whether we are processing an editing term without forcing it to look for an editing term **Push** in the context: **ET-state** indicates whether we are processing an editing term and whether a cue phrase was seen; and **ET-prev** indicates the number of editing term words seen so far.

The contexts given to the decision tree algorithm encode basic knowledge about the effects of speech repairs, editing terms and intonational phrase boundaries. This allows the limited amount of training data to be used more effectively in estimating the probability distributions.

5.5 Correcting Speech Repairs

The previous section focused on the detection of speech repairs, editing terms and intonational phrases. But for repairs, we have only addressed half of the problem; the other half is determining the extent of the reparandum. Hindle (1983) and Kikui and Morimoto (1994) focused on correcting speech repairs, assuming the interruption point would be already detected. Although the model of the previous section detects repairs, this model is not effective enough. One of its crucial shortcomings is that it does not use as evidence whether there is a suitable correction (Heeman, Loken-Kim and Allen 1996). Since hearers are often unaware of speech repairs (Martin and Strange 1968), they must be able to correct them as the utterance is unfolding and as an indistinguishable event from detecting them and recognizing the words involved.

Recently, Stolcke and Shriberg (1996b) presented a word-based model for speech recognition that models simple word deletion and repetition patterns. They used the prediction of the repair to clean up the context and help predict what word will occur next. Although their model is limited to simple types of repairs, it provides a starting point for incorporating speech repair correction into a statistical language model.

5.5.1 Our Approach

There are several sources of information that give evidence as to the extent of the reparandum. Probably the most widely used is the presence of word correspondences between the reparandum and alteration, both at the word level and at the level of syntactic constituents (Levelt 1983, Hindle 1983, Bear et al. 1992, Heeman and Allen 1994, Kikui and Morimoto 1994). Second, there tends to be a fluent transition from the speech that precedes the onset of the reparandum to the alteration (Kikui and Morimoto 1994). This source is very important for repairs that do not have initial retracing, and is the mainstay of the ‘parser-first’ approach (e.g. Dowding, Gawron, Appelt, Bear, Cherny, Moore and Moran 1993)—keep trying alternative corrections until one of them parses. Third, there are certain regularities for where speakers restart. Reparandum

onsets tend to be at constituent boundaries (Nooteboom 1980), and in particular, at boundaries where a co-ordinated constituent can be placed (Levelt 1983). Hence, reparandum onsets can be partially predicted without even looking at the alteration.

To model the correction of speech repairs, we add three more variables to our language model that enables us to make use of the above sources of evidence. For each non-abridged repair, we hypothesize the reparandum onset, and as we process the subsequent words, we hypothesize to which word in the reparandum it corresponds (or licences it), and the correspondence type. With this expanded model, the words of the alteration should be better predicted by the proper hypothesis of the correction variables than by some other interpretation. Consider the following example with strong word correspondences.

Example 14 ((d93-3.2 utt45))

which engine are we are we taking
 ↑
 reparandum ip

If we predicted that a modification repair occurred and that the reparandum consists of ‘are we’, then the probability of ‘are’ being the first word of the alteration would be very high since it matches the first word of the reparandum. Conversely, if we are not predicting a modification repair with reparandum ‘are we’, then the probability of seeing ‘are’ would be much lower. The same holds for predicting the next word, ‘we’: it is more likely under the repair interpretation. As we process the words of the alteration, the repair interpretation will better account for the words that follow it, strengthening the interpretation.

When predicting the first word of the alteration, we can also make use of the second source of evidence identified above: the context provided by the words that precede the reparandum. Consider the following repair in which the first two words of the alteration are inserted.

Example 15 (d93-16.2 utt66)

and two tankers to of OJ to Dansville
 ↑
 reparandum ip

Here, if we know the reparandum is ‘to’, then we know that the first word of the reparandum must be a fluent continuation of the speech before the onset of the reparandum. In fact, we see that the repair interpretation provides better context for predicting the first word of the alteration than a hypothesis that predicts either the wrong reparandum onset or predicts no repair at all.

We also make use of the third source of information. When we initially hypothesize the reparandum onset, we can take into account the a priori probability that it will occur at that point. In the following example, the words ‘should’ and ‘the’ are preferred by Levelt’s co-ordinated constituent rule (Levelt

Example 19 (d93-16.3 utt4)

what's the shortest route from engine from for engine two at Elmira
 ↑ ↑
 ip *ip*

The reparandum of the first repair is 'from engine'. In predicting the reparandum of the second, we work from the cleaned up context: 'what's the shortest route from.'

The context used in estimating how likely a word is as the reparandum onset also includes the word we are querying. We also include the words and POS tags that precede the proposed reparandum onset, thus allowing the decision tree to check if the onset is at a suitable constituent boundary. Since reparanda rarely extend over more than one utterance, we include three variables that help indicate whether an utterance boundary is being crossed. The first indicates the number of intonational phrase boundaries embedded in the proposed reparandum. The second indicates the number of discourse markers in the reparandum. Discourse markers at the beginning of the reparandum are not included, and if discourse markers appear consecutively, the group is only counted once. The third indicates the number of fillers in the reparandum.

Another source of information is the presence of other repairs in the turn. In the Trains corpus, 35.6% of non-abridged repairs overlap. If a repair overlaps a previous one then its reparandum onset is likely to co-occur with the alteration onset of the previous repair. Hence we include a variable that indicates whether there is a previous repair, and if there is, whether the proposed onset coincides with, precedes, or follows the alteration onset of the preceding repair.

5.5.3 The Active Repair

Determining word correspondences is complicated by the occurrence of overlapping repairs. To keep our approach simple, we allow at most one previous word to license the correspondence. Consider again Example 19. Here, one could argue that the word 'for' corresponds to the word 'from' from either the reparandum of the first or second repair. In either case, the correspondence to the word 'engine' is from the reparandum of the first repair. Our approach is to first decide which repair the correspondence will be to and then decide which word of that repair's reparandum will license the current word. We always choose the most recent repair that has words in its reparandum that have not yet licensed a correspondence (other than a word fragment). Hence, the active repair for predicting the word 'for' is the second repair, while the active repair for predicting 'engine' is the first repair. For predicting the word 'two', neither the first nor second repair has any unlicensed words in their reparandum, and hence it will not have an active repair.

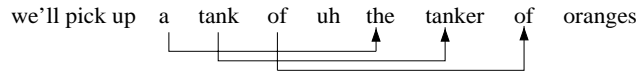


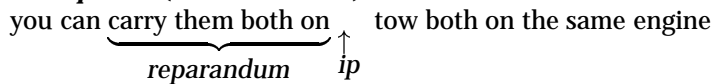
FIGURE 5.1: Cross Serial Correspondences between Reparandum and Alteration

5.5.4 Licensing a Correspondence

If we are in the midst of processing a repair, we can use the reparandum to help predict the current word W_i and its POS tag D_i . In order to do this, we need to determine which word in the reparandum of the active repair will *license* the current word. As illustrated in Figure 5.5.4, word correspondences for speech repairs tend to exhibit a cross serial dependency (Heeman and Allen 1994); in other words, if we have a correspondence between w_j in the reparandum and w_k in the alteration, any correspondence with a word in the alteration after w_k will be to a word that is after w_j . Hence, if there is already a correspondence for the repair then the licensing word will follow the last correspondence in the reparandum.

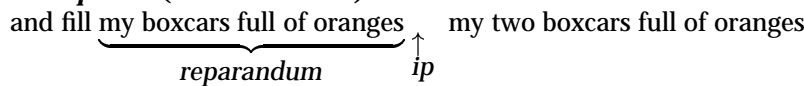
The licensing word might need to skip over words due to deleted words in the reparandum or inserted words in the alteration. In the example below, the word ‘tow’ is licensed by ‘carry’, but the word ‘them’ must be skipped over before processing the licensing between the two instances of ‘both’.

Example 20 (d92a-1.2 utt40)



The next example illustrates the opposite problem: the word ‘two’ has no correspondence with any word in the reparandum.

Example 21 (d93-15.4 utt45)



For words that have no correspondence, we define the licensing word as the first available word in the alteration, in this case ‘boxcars’. We leave it to the correspondence variable to encode that there is no correspondence. This gives us the following definition for the correspondence licenser, L_{ij} where i is the current word and j runs over all words in the reparandum of the active repair that come after the last word in the reparandum with a correspondence.

$$L_{ij} = \begin{cases} \mathbf{Corr} & W_j \text{ licenses the current word} \\ \mathbf{Corr} & W_i \text{ is inserted and } W_j \text{ is first available word in reparandum} \\ \mathbf{null} & \text{otherwise} \end{cases}$$

Just as with the reparandum onset, we estimate the probability by query-

ing each eligible word. The context for this query includes information about the proposed word, namely its POS tag. We also include information about the repair structure that has been found so far. If the previous word was a word match, there is a good chance that the current word will involve a word match to the next word. We include variables that indicates the number of words skipped in the reparandum and alteration since the last correspondence, the number of words since the onset of the reparandum and alteration, the number of words to the end of the reparandum, the type of repair and the reparandum length. We also include information about the POS and word context prior to the current word. This allows the decision tree to judge whether the proposed word is syntactically able to license the next word.

5.5.5 The Word Correspondence

Now that we have decided which word in the reparandum will potentially license the current word, we need to predict the type of correspondence. We focus on correspondences involving exact word match (identical POS tag and word), word replacements (same POS tag), or no such correspondence.

$$C_i = \begin{cases} \mathbf{m} & W_i \text{ is a word match of the word indicated by } L_i \\ \mathbf{r} & W_i \text{ is a word replacement of the word indicated by } L_i \\ \mathbf{x} & W_i \text{ has no correspondence (inserted word)} \\ \mathbf{null} & \text{No active repair} \end{cases}$$

The context used for estimating the correspondence variable is exactly the same as that used for estimating the licenser.

5.5.6 Redefining the Speech Recognition Problem

Now that we have introduced the correction tags, we redefine the speech recognition problem so that it includes finding the most probable corrections tags.

$$\begin{aligned} \hat{W} \hat{P} \hat{C} \hat{L} \hat{O} \hat{R} \hat{E} \hat{I} &= \arg \max_{WPCLOREI} \Pr(WPCLOREI|A) \\ &= \arg \max_{WPCLOREI} \Pr(A|WPCLOREI) \Pr(WPCLOREI) \end{aligned} \quad (5.7)$$

The second term is the language model and can be rewritten as we did for Equation 5.5.

In Section 5.5.2, we discussed that the word and POS context for the probability distributions can now exclude the reparanda of previous repairs. This not only applies to the three new probability distributions, but to the other five as well. Consider the following example.

Example 22 (d93-13.1 utt64)

pick up and load two um the two boxcars on engine two
 reparandum ↑ ip

In processing the word ‘the’, if we hypothesized that it follows a modification

repair with editing term ‘um’ and reparandum ‘two’, then we can now generalize with fluent examples, such as the following, in hypothesizing its POS tag and the word identity.

Example 23 (d93-12.4 utt97)

and to make the orange juice and load the tankers

For predicting the word and POS tags, we have an additional source of information, namely the values of the correspondence licenser and the correspondence type. Rather than use these two variables as part of the context that we give the decision tree algorithm, we use these tags to override the decision tree probability. If a word replacement or word match was hypothesized, we assign all of the POS probability to the appropriate POS tag. If a word match was hypothesized, we assign all of the word probability to the appropriate word.

5.6 Example

This section illustrates the workings of the algorithm. We illustrate the algorithm where it is constrained to the actual word transcription.² The algorithm incrementally considers all possible interpretations proceeding one word at a time. Low scoring paths are pruned so as to keep the search space tractible. Consider the following example.

Example 24 (d92a-2.1 utt95)

okay % uh and that will take a total of um let’s see total of s- of seven hours
 reparandum ip et reparandum ip

Rather than try to show all of the competing hypotheses, we focus on the correct interpretation, which, for this example, happens to be the winning interpretation. We contrast the probabilities of the correct tags with those of its competitors. For reference, we give a simplified view of the context that is used for each probability. Full results of the algorithm will be given in the next section.

5.6.1 Predicting ‘um’ as the Onset of an Editing Term

Below, we give the probabilities involved in the correct interpretation of the word ‘um’ given the correct interpretation of the words ‘okay uh and that will take a total of’. We start with the intonation variable. The correct tag of **null** is significantly preferred over the alternative, mainly because intonational boundaries rarely follow prepositions.

2. In other work, we have used the language model to rescore word graphs (Heeman 1999).

$$\Pr(I_{10}=\mathbf{null} \mid \text{a total of}) = 0.9997$$

$$\Pr(I_{10}=\% \mid \text{a total of}) = 0.0003$$

For $I_{10} = \mathbf{null}$, we give the alternatives for the editing term tag. Since an editing term is not in progress, the only possible values are **Push** and **null**.

$$\Pr(E_{10}=\mathbf{Push} \mid \text{a total of}) = 0.242$$

$$\Pr(E_{10}=\mathbf{null} \mid \text{a total of}) = 0.758$$

With $E_{10} = \mathbf{Push}$, the only allowable repair tag is **null**. Since no repair has been started, the reparandum onset O_{10} must be **null**. Similarly, since no repair is in progress, L_{10} , the correspondence licenser, and C_{10} , the correspondence type, must both be **null**.

We next hypothesize the POS tag. Below we list all of the tags that have a probability greater than 1%. Since we are starting an editing term, we see that POS tags associated with the first word of an editing term have a high probability, such as **UH_FP** for ‘um’, **AC** for ‘okay’, **CC_D** for ‘or’, **UH_D** for ‘well’, and **VB** for the ‘let’ in ‘let’s see’.

$$\Pr(D_{10}=\mathbf{UH_FP} \mid \text{a total of Push}) = 0.731$$

$$\Pr(D_{10}=\mathbf{AC} \mid \text{a total of Push}) = 0.177$$

$$\Pr(D_{10}=\mathbf{CC_D} \mid \text{a total of Push}) = 0.026$$

$$\Pr(D_{10}=\mathbf{UH_D} \mid \text{a total of Push}) = 0.020$$

$$\Pr(D_{10}=\mathbf{VB} \mid \text{a total of Push}) = 0.026$$

For D_{10} set to **UH_FP**, the word choices are ‘um’, ‘uh’, and ‘er’.

$$\Pr(W_{10}=\text{um} \mid \text{a total of Push UH_FP}) = 0.508$$

$$\Pr(W_{10}=\text{uh} \mid \text{a total of Push UH_FP}) = 0.488$$

$$\Pr(W_{10}=\text{er} \mid \text{a total of Push UH_FP}) = 0.004$$

Given the correct interpretation of the previous words, the probability of the filler ‘um’ along with the correct tags is 0.090.

5.6.2 Predicting ‘total’ as the Alteration Onset

We now give the probabilities involved in the second instance of ‘total’, which is the alteration onset of the first repair, whose editing term ‘um let’s see’, which ends an intonational phrase, has just finished. Again we start with the intonation variable.

$$\Pr(I_{14}=\% \mid \text{a total of Push um let’s see}) = 0.902$$

$$\Pr(I_{14}=\mathbf{null} \mid \text{a total of Push um let’s see}) = 0.098$$

For $I_{14} = \%$, the editing term probabilities are given below. Since an editing term is in progress, the only possibilities are that it is continued or that it has ended.

$$\Pr(E_{14}=\mathbf{Pop} \mid \text{a total of Push um let’s see } \%) = 0.830$$

$$\Pr(E_{14}=\mathbf{ET} \mid \text{a total of Push um let’s see } \%) = 0.170$$

For $E_{14} = \mathbf{Pop}$, we give the probabilities for the repair variable. Since an editing term has just ended, the null tag for the repair variable is ruled out. Note the modification interpretation receives a score approximately one third of that of a fresh start. However, the repair interpretation catches up after the alteration is processed.

$$\Pr(R_{14}=\mathbf{Mod} \mid \text{a total of Push um let's see \% Pop}) = 0.228$$

$$\Pr(R_{14}=\mathbf{Can} \mid \text{a total of Push um let's see \% Pop}) = 0.644$$

$$\Pr(R_{14}=\mathbf{Abr} \mid \text{a total of Push um let's see \% Pop}) = 0.128$$

For $R_{14} = \mathbf{Mod}$, we give the probabilities assigned to the possible reparandum onsets. For each, we give the proposed reparandum onset, X , and the words that precede it.

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{take a total} \quad X=\text{of} \quad R=\mathbf{Mod}) = 0.589$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{will take a} \quad X=\text{total} \quad R=\mathbf{Mod}) = 0.126$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{that will take} \quad X=\text{a} \quad R=\mathbf{Mod}) = 0.145$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{and that will} \quad X=\text{take} \quad R=\mathbf{Mod}) = 0.023$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{uh and that} \quad X=\text{will} \quad R=\mathbf{Mod}) = 0.016$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{\% uh and} \quad X=\text{that} \quad R=\mathbf{Mod}) = 0.047$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{okay \% uh} \quad X=\text{and} \quad R=\mathbf{Mod}) = 0.047$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{<turn> okay \%} \quad X=\text{uh} \quad R=\mathbf{Mod}) = 0.003$$

$$\Pr(O_{14,X}=\mathbf{Onset} \mid W=\text{<turn>} \quad X=\text{okay} \quad R=\mathbf{Mod}) = 0.003$$

With ‘total’ as the reparandum onset, there are two possibilities for which word of the reparandum will license the current word—either the word ‘total’ or ‘of’.

$$\Pr(L_{10,X}=\mathbf{Corr} \mid W=\text{will take a} \quad X=\text{total} \quad R=\mathbf{Mod}) = 0.973$$

$$\Pr(L_{10,X}=\mathbf{Corr} \mid W=\text{will take a} \quad X=\text{of} \quad R=\mathbf{Mod}) = 0.027$$

With ‘total’ as the correspondence licenser, we need to decide the type of correspondence: whether it is a word match, word replacement or otherwise.

$$\Pr(C_{14}=\mathbf{m} \mid W=\text{will take a} \quad L=\text{total} \quad R=\mathbf{Mod}) = 0.5882$$

$$\Pr(C_{14}=\mathbf{r} \mid W=\text{will take a} \quad L=\text{total} \quad R=\mathbf{Mod}) = 0.1790$$

$$\Pr(C_{14}=\mathbf{x} \mid W=\text{will take a} \quad L=\text{total} \quad R=\mathbf{Mod}) = 0.2328$$

For the correct interpretation, the word correspondence is a word match with the word ‘total’ and POS tag NN. Hence, the POS tag and identity of the current word are both fixed and hence have a probability of 1. Given the correct interpretation of the previous words, the probability of the word ‘total’ along with the correct tags is 0.0111.

5.7 Results and Comparison

In this section, we present the results of running our model on the Trains corpus. We first explain the methodology that we use throughout the experiments, we then give results that indicate modeling speech repairs and intonational phrasing improves language modeling and POS tagging. We then give results for the tasks of identifying intonational phrase endings, detecting speech repairs and correcting them. We also compare our results with those reported by other researchers. This comparison is not exact because other researchers used different corpora, and employed different inputs. Also, our approach is the only one that has combined the detection and correction of speech repairs, and identification of intonational phrase boundaries, POS tags, and discourse markers, in a speech recognition model. Hence our comparison is with systems that only address part of the problem.

5.7.1 Experimental Setup

We tested our model on the hand-collect transcripts of the Trains Corpus in order to determine how well it could detect and correct speech repairs, and identify intonational phrases. We used a six-fold cross-validation procedure. The dialogs were divided into six partitions and each was tested using a model built from the other five. Changes in speaker are marked in the word transcription with the special token <turn>. We treat contractions, such as ‘that’ll’ and ‘gonna’, as separate words, treating them as ‘that’ and ‘ll’ for the first example, and ‘going’ and ‘ta’ for the second. We also changed all word fragments into a common token <fragment>. In searching for the best sequence of POS tags for the transcribed words, we follow the technique proposed by Chow and Schwartz (1989) and only keep a small number of alternative paths by pruning the low probability paths after processing each word.

5.7.2 Perplexity, Recall and Precision

A way to measure the effectiveness of the language model is to measure the *perplexity* that it assigns to a test corpus (Bahl, Baker, Jelinek and Mercer 1977). Perplexity is an estimate of how well the language model is able to predict the next word of a test corpus in terms of the number of alternatives that need to be considered at each point. For word-based language models, with estimated probability distribution of $\hat{\text{Pr}}(w_i|w_{1,i-1})$, the perplexity of a test set $w_{1,N}$ is calculated as 2^H , where H is the entropy, which is defined as $H = -\frac{1}{N} \sum_{i=1}^N \log_2 \hat{\text{Pr}}(w_i|w_{1,i-1})$.

We report results on identifying intonational phrase boundaries and speech repairs in terms of *recall*, *precision* and *error rate*. The recall rate is the number of times that the algorithm correctly identifies an event over the total number of times that it actually occurred. The precision rate is the number of times the algorithm correctly identifies it over the total number of times it identifies it. The error rate is the number of errors in identifying an event over the number of times that the event occurred.

5.7.3 POS Tagging and Perplexity

Table 5.7.3 shows that POS tagging and word perplexity benefit from modeling intonational phrases and speech repairs. The second column gives the results of the POS-based language model of Section 5.3. Column three adds

	WP	WPCLOREI	WPCLOREIS
POS Errors	1711	1652	1563
POS Error Rate	2.93	2.83	2.68
Word Perplexity	24.04	22.96	22.35

TABLE 5.1: Comparison of POS tagging, discourse marker identification and perplexity rates

speech repair and intonational phrase modeling and results in an improvement to word perplexity and POS tagging. Column four adds silence information. The amount of silence that followed a word was used to adjust the probability distributions for the repair, editing term and intonation variables (Heeman and Allen 1999). Silence information was not used to adjust the POS nor word probability distributions. We see that modeling speech repairs and intonational phrases results in an overall perplexity reduction of 7.0% over the POS-based model. We also see a significant improvement in POS tagging with an error rate reduction of 8.6% over the POS-based model. As we further improve the modeling of the user’s utterance, we should expect to see further improvements in the language model.

5.7.4 Intonational Phrases

In Table 5.7.4, we give the results of our full model in detecting intonational phrase boundaries. We purposely divide intonational phrase boundaries that occur within a speaker’s turn from those that occur at the end. The reason for this is that our model uses the end-of-turn information as part of its input, and since almost all turns end with an intonational phrase boundary, it easily learns this regularity. As for the within turn boundaries, the model achieves a recall rate of 71.8% and a precision of 70.8%.

	Recall	Precision	Error Rate
Within Turn	71.76	70.82	57.79
End of Turn	98.05	94.17	8.00
All Boundaries	84.76	82.53	33.17

TABLE 5.2: Intonational phrase results for full model

In comparison to our work, Wightman and Ostendorf (1994) made much fuller use of acoustical information: using preboundary lengthening, pausal durations, as well as other acoustic cues to automatically label intonational phrases and word accents. They trained a decision tree to estimate the probability of a phrase boundary given the acoustic context. These probabilities were fed into a Markov model whose state is the boundary type of the previous word. For training and testing their algorithm, they used a single-speaker corpus of news stories read by a public radio announcer. With this speaker-dependent model, they achieved a recall rate of 78.1% and a precision of 76.8%. However, it is unclear how well this will adapt to spontaneous speech, where repairs might interfere with the cues that they use, and to speaker independent testing.

Wang and Hirschberg (1992) also looked at detecting intonational phrases. Using automatically-labeled features, including POS tag of the current word, category of the constituent being built, distance from last boundary, and word accent, they built decision trees to classify each word as to whether it

has an intonational boundary. With this approach, they achieved a recall rate of 79.5% and a precision rate of 82.7% on a subset of the ATIS corpus. Excluding end-of-turn data gives a recall rate of 72.2% and a precision of 76.2%. These results group speech repairs with intonational boundaries. In their corpus, there were 424 disfluencies and 405 turn-internal boundaries. The performance of the decision tree that does not classify disfluencies as intonational boundaries is significantly worse. However, these results were achieved with one-tenth the data of the Trains corpus.

Kompe, Kießling, Niemann, Nöth, Schukat-Talamazzini, Zottmann and Batliner (1995) combined acoustic cues with a statistical language model to find intonational phrases. They combined normalized syllable duration, length of pauses, pitch contour and energy using a multi-layered perceptron that estimates the probability $\Pr(v_i|c_i)$, where v_i indicates if there is a boundary after the current word and c_i is the acoustic features of the neighboring six syllables. This score is combined with the score from a statistical language model, which determines the probability of the word sequence with the hypothesized phrase boundary inserted using a backoff strategy. Building on this work, Mast et al. (1996) segmented speech into speech acts as the first step in automatically classifying them and achieved a recognition accuracy of 92.5% on turn internal boundaries using Verbmobil dialogs. This translates into a recall rate of 85.0%, a precision of 53.1% and an error rate of 90.1%. Their model, which employs rich acoustic modeling, does not account for interactions with speech repairs, POS tags, nor does it redefine the speech recognition language model.

Meteer and Iyer (1996) investigated whether modeling linguistic segments, segments with a single independent clause, improves language modeling. They computed the probability of the sequence of words with the hypothesized segment boundaries inserted into the sequence. Working on the Switchboard corpus, they found that predicting linguistic boundaries improved perplexity from 130 to 127. Similar to this work, Stolcke and Shriberg (1996a) investigated how the language-model can find the boundaries. Their best results were obtained by using POS tags as part of the input, as well as the word identities of certain word classes, in particular fillers, conjunctions, and certain discourse markers. However, this work does not incorporate the automatic POS tagging and discourse marker identification.

5.7.5 Speech Repairs

Table 5.7.5 gives the results of detecting speech repairs using our full model. The first row gives the recall and precision rates using the measure we refer to as *All Repairs*, in which we ignore errors that result from improperly identifying the type of repair, and hence scores a repair as correctly detected as long as it was identified as either an abridged repair, modification repair or fresh start. Furthermore, when multiple repairs have contiguous reparanda, we count all repairs involved (of the hand-annotations) as correct as long as the combined reparandum is correctly identified. Hence, for Example 19 given earlier, as long

as the overall reparandum was identified as ‘from engine from’, both of the hand-annotated repairs are counted as correct. We see that we are able to

	Recall	Precision	Error Rate
All Repairs	76.79	86.66	35.01
Abridged	75.88	82.51	40.18
Modification	80.87	83.37	35.25
Fresh Starts	48.58	69.21	73.02
Modification & Fresh Starts	73.69	83.85	40.49

TABLE 5.3: Speech repair detection for full model

detect speech repairs with a recall of 76.8% and a precision of 86.7%. The next three rows of Table 5.7.5 give the rates at which each individual type of repair was correctly identified. The measure we used here is slightly different from the *All Repairs* measure: misclassifications of a repair type were counted as wrong if the extent of the repair was not correctly identified. Here we see that fresh starts are the most difficult type of repair to detect. The fifth column combines the results of the modification repairs and fresh starts and does not count misclassifications between these two types of repairs.

Table 5.7.5 gives the results for correcting speech repairs. For all of the measures, a repair is counted as correctly corrected if it was identified and the extent of the reparandum was correctly determined. Our overall recall rate is 65.9% with a precision of 74.3%.

	Recall	Precision	Error Rate
All Repairs	65.85	74.32	56.88
Abridged	75.65	82.26	40.66
Modification	77.95	80.36	41.09
Fresh Starts	36.21	51.59	97.76
Modification & Fresh Starts	63.76	72.54	60.36

TABLE 5.4: Speech repair correction for full model

A number of other researchers have addressed the issue of detecting and correcting speech repairs. Bear et al. (1992) investigated the use of pattern matching of the word correspondences, global and local syntactic and semantic ill-formedness, and acoustic cues as evidence for detecting speech repairs. They tested their pattern matcher on a subset of the ATIS corpus from which they removed all *trivial* repairs, repairs that involve only the removal of a word fragment or a filler. For their pattern matching results, they achieved a detection recall rate of 76% with a precision of 62%, and a correction recall rate of 44% with a precision of 35%. They also combined syntactic and semantic knowledge in a ‘parser-first’ approach—first try to parse the input and if that fails, invoke repair strategies based on word patterns in the input. In a test set

containing 26 repairs (Dowding et al. 1993), they obtained a detection recall rate of 42% with a precision of 85%, and a correction recall rate of 31% with a precision of 62%.

Nakatani and Hirschberg (1994) proposed that speech repairs should be detected in a *speech-first* model using acoustic-prosodic cues, without relying on a word transcription. In order to test their theory, they built a decision tree using a training corpus of 148 turns of speech. They used hand-transcribed prosodic-acoustic features such as silence duration, energy, and pitch, as well as traditional text-first cues such as presence of word fragments, fillers, word matches, word replacements, POS tags, and position of the word in the turn and obtained a detection recall rate of 86.1% with a precision of 91.2%. The cues they found relevant were duration of pauses between words, word fragments, and lexical matching within a window of three words. Note that in their corpus 73.3% of the repairs were accompanied by a word fragment, as opposed to 32% of the modification repairs and fresh starts in the Trains corpus. Hence, word fragments are a stronger indicator of speech repairs in their corpus than in the Trains corpus. Also note that their training and test sets only included turns with speech repairs; hence their “findings should be seen more as indicative of the relative importance of various predictors of [speech repair] location than as a true test of repair site location.”

Stolcke and Shriberg (1996b) incorporated repair resolution into a word-based language model. They limited the types of repairs to single and double word repetitions and deletions, deletions from the beginning of the sentence and fillers. In predicting a word, they sum over the probability distributions for each type of repair (including no repair at all). For hypotheses that include a repair, the prediction of the next word is based upon a cleaned-up representation of the context, as well as taking into account if they are predicting a single or double word repetition. Surprisingly, they found that this model actually degrades performance, in terms of perplexity and word error rate. They attributed this to their treatment of fillers: utterance-medial fillers should be cleaned up before predicting the next word, whereas utterance-initial ones should be left intact, a distinction that we make in our model by modeling intonational phrases.

Siu and Ostendorf (1996) extended a language model to account for three roles that words such as fillers can play in an utterance: utterance initial, part of a non-abridged repair, or part of an abridged repair. By using training data with these roles marked and a function-specific variable n -gram model (i.e. use different context for the probability estimates depending on the function of the word), and summing over each possible role, they achieved a perplexity reduction of 82.9 to 81.1.

5.8 Conclusion and Future Work

In this chapter, we redefined the speech recognition language model so that it also identifies intonational phrases and resolves speech repairs. This allows the language model to better account for the words involved in a speaker's turn and allows it to return a more meaningful analysis of the speaker's turn for later processing. The model incorporates identifying intonational phrases, POS tags, and detecting and correcting speech repairs; hence, interactions that exist between these tasks, as well as the task of predicting the next word, can be modeled.

Constraining our model to the hand transcription, it is able to identify 71.8% of all turn-internal intonational boundaries with a precision of 70.8%, and detect and correct 65.9% of all speech repairs with a precision of 74.3%. These results are partially attributable to accounting for the interaction between these tasks (Heeman and Allen 1999). Speech repairs and intonational phrases create discontinuities that traditional speech recognition language models and POS taggers have difficulty modeling. Modeling speech repairs and intonational phrases results in an 8.6% improvement in POS tagging and a 7.0% improvement in perplexity. Part of this improvement is from exploiting silences to give evidence of the speech repairs and intonational phrase boundaries.

More work still needs to be done. First, with the exception of pauses, we have not consider acoustic cues. This is a rich source of information for detecting (and distinguishing between) intonational phrases and interruption points of speech repairs. It would also help in determining the reparandum onset of fresh starts, which tend to occur at intonational boundaries. Acoustic modeling is also needed to identify word fragments. The second area is extending the model to incorporate higher level syntactic and semantic processing. This would not only allow us to give a much richer output from the model, but it would also allow us to account for interactions between this higher level knowledge and modeling speakers' utterances, especially in detecting the ill-formedness that often occur with speech repairs. It would also aid in finding richer correspondences between the reparandum and alteration, such as between the noun phrase and pronoun in the following example.

Example 25 (d93-14.3 utt27)

the engine can take as many $\underbrace{\hspace{10em}}$ \uparrow $\underbrace{\hspace{2em}}$ $\underbrace{\hspace{2em}}$ up to three loaded boxcars
reparandum *ip* *et* *alteration*

The third and most important area is to incorporate our work into a speech recognizer. We have already used our POS-based model to rescore word-graphs, which results in a one percent absolute reduction in word error rate in comparison to a word-based model. Our full model, which accounts for intonational phrases and speech repairs, leads to a further reduction, as well as returns a

richer understanding of the speech (Heeman 1999).

Acknowledgments

Funding gratefully received from NSERC Canada, NSF under grant IRI-9623665, DARPA—Rome Laboratory under research contract F30602-95-1-0025, ONR/DARPA under grant N00014-92-J-1512, ONR under grant N0014-95-1-1088, ATR Interpreting Telecommunications Laboratory and CNET, France Télécom.

References

- Bahl, L. R., Baker, J. K., Jelinek, F. and Mercer, R. L.: 1977. Perplexity—a measure of the difficulty of speech recognition tasks. *Proceedings of the 94th Meeting of the Acoustical Society of America*.
- Bahl, L. R., Brown, P. F., de Souza, P. V. and Mercer, R. L.: 1989. A tree-based statistical language model for natural language speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing* **36**(7), 1001–1008.
- Beach, C. M.: 1991. The interpretation of prosodic patterns at points of syntactic structure ambiguity: Evidence for cue trading relations. *Journal of Memory and Language* **30**(6), 644–663.
- Bear, J. and Price, P.: 1990. Prosody, syntax, and parsing. *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*. Pittsburgh. pp. 17–22.
- Bear, J., Dowding, J. and Shriberg, E.: 1992. Integrating multiple knowledge sources for detection and correction of repairs in human-computer dialog. *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*. pp. 56–63.
- Bear, J., Dowding, J., Shriberg, E. and Price, P.: 1993. A system for labeling self-repairs in speech. *Technical Note 522*. SRI International.
- Black, E., Jelinek, F., Lafferty, J., Magerman, D., Mercer, R. and Roukos, S.: 1992. Towards history-based grammars: Using richer models for probabilistic parsing. *Proceedings of the DARPA Speech and Natural Language Workshop*. Morgan Kaufman. pp. 134–139.
- Breiman, L., Friedman, J. H., Olshen, R. A. and Stone, C. J.: 1984. *Classification and Regression Trees*. Wadsworth & Brooks. Monterrey, CA.
- Brown, P. F., Della Pietra, V. J., deSouza, P. V., Lai, J. C. and Mercer, R. L.: 1992. Class-based n -gram models of natural language. *Computational Linguistics* **18**(4), 467–479.
- Chow, Y. and Schwartz, R.: 1989. The n -best algorithm: An efficient procedure for finding top n sentence hypotheses. *Proceedings of the DARPA Speech and Natural Language Workshop*. Morgan Kaufman. San Mateo, California. pp. 199–202.
- Dowding, J., Gawron, J. M., Appelt, D., Bear, J., Cherny, L., Moore, R. and Moran, D.: 1993. Gemini: A natural language system for spoken-language understanding. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*. pp. 54–61.
- Heeman, P. A.: 1997. Speech repairs, intonational boundaries and discourse markers: Modeling speakers' utterances in spoken dialog. *Technical Report 673*. Department of Computer Science, University of Rochester. Doctoral dissertation.
- Heeman, P. A.: 1999. Modeling speech repairs and intonational phrasing to improve speech recognition. *Automatic Speech Recognition and Understanding Workshop*. Keystone Colorado.
- Heeman, P. A. and Allen, J. F.: 1995. The Trains spoken dialog corpus. *CD-ROM*. Linguistics Data Consortium.
- Heeman, P. A. and Allen, J. F.: 1999. Speech repairs, intonational phrases and discourse markers: Modeling speakers' utterances in spoken dialog. *Computational Linguistics* **25**(4), 527–572.
- Heeman, P. A., Loken-Kim, K. and Allen, J. F.: 1996. Combining the detection and correction of speech repairs. *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP-96)*. Philadelphia. pp. 358–361. Also appears in *International Symposium on Spoken Dialogue*, 1996, pages 133–136.

- Heeman, P. and Allen, J.: 1994. Detecting and correcting speech repairs. *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*. Las Cruces, New Mexico. pp. 295–302.
- Hindle, D.: 1983. Deterministic parsing of syntactic non-fluencies. *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*. pp. 123–128.
- Jelinek, F.: 1985. Self-organized language modeling for speech recognition. *Technical report*. IBM T.J. Watson Research Center, Continuous Speech Recognition Group. Yorktown Heights, NY.
- Kikui, G.-i. and Morimoto, T.: 1994. Similarity-based identification of repairs in Japanese spoken language. *Proceedings of the 3rd International Conference on Spoken Language Processing (ICSLP-94)*. pp. 915–918.
- Kompe, R., Kießling, A., Niemann, H., Nöth, E., Schukat-Talamazzini, E. G., Zotmann, A. and Batliner, A.: 1995. Prosodic scoring of word hypotheses graphs. *Proceedings of the 4th European Conference on Speech Communication and Technology (Eurospeech)*. Madrid. pp. 1333–1336.
- Levelt, W. J. M.: 1983. Monitoring and self-repair in speech. *Cognition* **14**, 41–104.
- Martin, J. G. and Strange, W.: 1968. The perception of hesitation in spontaneous speech. *Perception and Psychophysics* **53**, 1–15.
- Mast, M., Kompe, R., Harbeck, S., Kießling, A., Niemann, H., Nöth, E., Schukat-Talamazzini, E. G. and Warnke, V.: 1996. Dialog act classification with the help of prosody. *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP-96)*. Philadelphia. pp. 1728–1731.
- Meteer, M. and Iyer, R.: 1996. Modeling conversational speech for speech recognition. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Philadelphia. pp. 33–47.
- Nakatani, C. H. and Hirschberg, J.: 1994. A corpus-based study of repair cues in spontaneous speech. *Journal of the Acoustical Society of America* **95**(3), 1603–1616.
- Nooteboom, S. G.: 1980. Speaking and unspeaking: Detection and correction of phonological and lexical errors. in V. A. Fromkin (ed.), *Errors in Linguistic Performance: Slips of the Tongue, Ear, Pen, and Hand*. Academic Press. New York. pp. 87–96.
- Ostendorf, M., Wightman, C. and Veilleux, N.: 1993. Parse scoring with prosodic information: an analysis/synthesis approach. *Computer Speech and Language* **7**(2), 193–210.
- Oviatt, S.: 1995. Predicting spoken disfluencies during human-computer interaction. *Computer Speech and Language* **9**, 19–35.
- Price, P.: 1997. Spoken language understanding. in R. Cole, J. Mariani, H. Uszkoreit, A. Zaenen and V. Zue (eds), *Survey of the State of the Art in Human Language Technology*. Cambridge University Press.
- Rosé, C. P. and Lavie, A.: 2001. Balancing robustness and efficiency in unification-augmented context-free parsers for large practical applications. *Robustness in Language and Speech Technology*. Kluwer Academic Publishers.
- Silverman, K., Beckman, M., Pitrelli, J., Ostendorf, M., Wightman, C., Price, P., Pierrehumbert, J. and Hirschberg, J.: 1992. ToBI: A standard for labelling English prosody. *Proceedings of the 2nd International Conference on Spoken Language Processing (ICSLP-92)*. pp. 867–870.
- Siu, M. and Ostendorf, M.: 1996. Modeling disfluencies in conversational speech. *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP-96)*. pp. 382–391.
- Stolcke, A. and Shriberg, E.: 1996a. Automatic linguistic segmentation of conversational speech. *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP-96)*. pp. 1001–1004.
- Stolcke, A. and Shriberg, E.: 1996b. Statistical language modeling for speech disfluencies. *Proceedings of the International Conference on Audio, Speech and Signal Processing (ICASSP)*. pp. 405–408.
- Traum, D. R. and Heeman, P. A.: 1997. Utterance units in spoken dialogue. in E. Maier, M. Mast and S. LuperFoy (eds), *Dialogue Processing in Spoken Language Systems*. Lecture Notes in Artificial Intelligence. Springer-Verlag. Heidelberg. pp. 125–140.
- van Noord, G.: 2001. Robust parsing of word graphs. *Robustness in Language and Speech Technology*. Kluwer Academic Publishers.

- Wang, M. Q. and Hirschberg, J. 1992. Automatic classification of intonational phrase boundaries. *Computer Speech and Language* **6**, 175–196.
- Ward, W.: 1991. Understanding spontaneous speech: The Phoenix system. *Proceedings of the International Conference on Audio, Speech and Signal Processing (ICASSP)*. pp. 365–367.
- Wightman, C. W. and Ostendorf, M. 1994. Automatic labeling of prosodic patterns. *IEEE Transactions on Speech and Audio Processing* **2**(4), 469–481.
- Young, S. R. and Matessa, M.: 1991. Using pragmatic and semantic knowledge to correct parsing of spoken language utterances. *Proceedings of the 2nd European Conference on Speech Communication and Technology (Eurospeech)*. Genova, Italy. pp. 223–227.