Tagging Speech Repairs

Peter A. Heeman and James Allen

Department of Computer Science
University of Rochester
Rochester, New York, 14627
{heeman,james}@cs.rochester.edu

ABSTRACT

This paper describes a method of detecting speech repairs that uses a part-of-speech tagger. The tagger is given knowledge about category transitions for speech repairs, and so is able to mark a transition either as a likely repair or as fluent speech. Other contextual clues, such as editing terms, word fragments, and word matchings, are also factored in by modifying the transition probabilities.

1. Introduction

Interactive spoken dialog provides many new challenges for spoken language systems. One of the most critical is the prevalence of speech repairs. Speech repairs are dysfluencies where some of the words that the speaker utters need to be removed in order to correctly understand the speaker's meaning. These repairs can be divided into three types: *fresh starts*, *modifications*, and *abridged*. A fresh start is where the speaker abandons what she was saying and starts again.

> the current plan is we take – okay let's say we start with the bananas (d91-2.2 utt105)

A modification repair is where the speech repair modifies what was said before.

> after the orange juice is at – the oranges are at the OJ factory (d93-19.3 utt59)

An abridged repair is where the repair consists solely of a fragment and/or editing terms.

> we need to – um manage to get the bananas to Dansville more quickly (d93-14.3 utt50)

In these examples, the "–" marks the interruption point, the point that marks the end of the removed text (including word fragments), and precedes the editing terms, if present. In our corpus of problem solving dialogs, 25% of turns contain at least one repair, 67% of repairs occur with at least one other repair in the turn, and repairs in the same turn occur on average within 6 words of each other. As a result, no spoken language system will perform well without an effective way to detect and correct speech repairs.

We propose that speech repairs can be detected and corrected within the local context of the repair. So, clues are needed for detecting repairs that do not depend on such global properties as the syntactic or semantic well-formedness of the entire utterance. But this does not mean that syntactic clues cannot be used. One powerful predictor of modification repairs is the presence of a syntactic anomaly (c.f. Bear, Dowding and Shriberg, 1992) at the interruption point. The anomaly occurs because the text after the interruption point is not intended to follow the text before the interruption, but to replace it, so there is no reason why the text before and the text after need to be syntactically well-formed. In this paper, we describe how the syntactic anomalies of modification repairs can be detected by a part-of-speech tagger, augmented with category transition probabilities for modification repairs. Because we use a statistical model, other clues, such as the presence of editing terms, word fragments, and word correspondence can be factored in by appropriately modifying the transition probabilities.

Focusing on the detection of modification repairs does not mean we are ignoring abridged repairs. Assuming that word fragments and editing terms can be detected, abridged repairs are easy to detect and correct. What is not trivial about these repairs is differentiating them from modification repairs, especially where there are incidental word correspondences. It is this distinction that makes such repairs easy to detect, but potentially difficult to correct. Since our approach looks for syntactic anomalies, other than those caused by word fragments and editing terms, it can distinguish abridged repairs from modification repairs, which should make both types of repairs easier to correct.

An ulterior motive for not using higher level syntactic or semantic knowledge is that the coverage of parsers and semantic interpreters is not sufficient for unrestricted dialogs. Recently, Dowding et al. (1993) reported syntactic and semantic coverage of 86% for the Darpa Airline reservation corpus. Unrestricted dialogs will present even more difficulties; not only will the speech be more ungrammatical, but there is also the problem of segmenting the dialog into utterance units (c.f. Wang and Hirschberg, 1992). If speech repairs can be detected and corrected before parsing and semantic interpretation, this should simplify those modules as well as make them more robust.

2. Previous Work

Several different strategies have been discussed in the literature for detecting and correcting speech repairs. One way to compare the effectiveness of these approaches is to look at their recall and precision rates. For detecting repairs, the recall rate is the number of correctly detected repairs compared to the number of repairs, and the precision rate is the number of detected repairs compared to the number of detections (including false positives). But the true measures of success are the correction rates. Correction recall is the number of repairs that were properly corrected compared to the number of repairs. Correction precision is the number of repairs that were properly corrected compared to the total number of corrections.

One of the first computational approaches was that taken by

Hindle (1983), who used a deterministic parser augmented with rules to look for matching categories and matching strings of words. Hindle achieved a correction recall rate of 97% on his corpus; however, this was obtained by assuming that speech repairs were marked by an explicit "edit signal" and with part-of-speech tags externally supplied.

The SRI group (Bear, Dowding and Shriberg, 1992) removed the assumption of an explicit edit signal, and employed simple pattern matching techniques for detecting and correcting modification repairs (they removed all utterances with abridged repairs from their corpus). For detection, they were able to achieve a recall rate of 76%, and a precision of 62%, and they were able to find the correct repair 57% of the time, leading to an overall correction recall of 43% and correction precision of 50%. They also tried combining syntactic and semantic knowledge in a "parser-first" approach—first try to parse the input and if that fails, invoke repair strategies based on their pattern matching technique. In a test set of 756 utterances containing 26 repairs (Dowding et al., 1993), they obtained a detection recall rate of 42% and a precision of 84.6%; for correction, they obtained a recall rate of 30% and a precision rate of 62%.

Nakatani and Hirschberg (1993) investigated using acoustic information to detect the interruption point of speech repairs. In their corpus, 74% of all repairs are marked by a word fragment. Using hand-transcribed prosodic annotations, they trained a classifier on a 172 utterance training set to identify the interruption point (each utterance contained at least one repair). On a test set of 186 utterances containing 223 repairs, they obtained a recall rate of 83.4% and a precision of 93.9% in detecting speech repairs. The clues that they found relevant were duration of pause between words, presence of fragments, and lexical matching within a window of three words. However, they do not address the problem of determining the correction or distinguishing modification repairs from abridged repairs.

3. The Corpus

As part of the TRAINS project (Allen and Schubert, 1991), which is a long term research project to build a conversationally proficient planning assistant, we are collecting a corpus of problem solving dialogs. The dialogs involve two participants, one who is playing the role of a user and has a certain task to accomplish, and another, who is playing the role of the system by acting as a planning assistant (Gross, Allen and Traum, 1992). The entire corpus consists of 112 dialogs totaling almost eight hours in length and containing about 62,000 words and 6300 speaker turns. These dialogs have been segmented into utterance files (c.f. Heeman and Allen, 1994c); words have been transcribed and the speech repairs have been annotated. For a training set, we use 40 of the dialogs, consisting of 24,000 words; and for testing, 7 of the dialogs, consisting of 5800 words.

In order to provide a large training corpus for the statistical model, we use a tagged version of the Brown corpus, from the Penn Treebank (Marcus, Santorini and Marcinkiewicz, 1993). We removed all punctuation in order to more closely approximate unsegmented spoken speech. This corpus provides us with category transition probabilities for fluent speech. These probabilities have also been used to bootstrap our algorithm in order to determine the category probabilities for speech repairs from our training corpus.[1]

[1] We found that the tagset used in the Penn Treebank did not always provide a fine enough distinction for detecting syntactic anomalies. We have made the following changes: (1) we separated prepositions from subordinating conjunctions; (2) we separated uses of "to" as a preposition from its use as part of a to-infinitive; (3) rather than classify verbs by tense, we classified them into four groups, conjugations of "be", conjugations of "have", verbs that are followed by a to-infinitive, and verbs that are followed immediately by another verb.

	Total	with Frag.	with Edit Term
Modification Repair	450	14.7%	19.3%
Word Repetition	179	16.2%	16.2%
Larger Repetition	58	17.2%	19.0%
Word Replacement	72	4.2%	13.9%
Other	141	17.0%	26.2%
Abridged Repair	267	46.4%	54.3%
Total	717	26.5%	32.4%

Table 1: Occurrence of Types of Repairs

Speech repairs can be divided into three intervals (c.f. Levelt, 1983), the removed text, editing terms, and the resumed text. The removed text and the editing terms are what need to be deleted in order to determine what the speaker intended to say.[2] There is typically a correspondence between the removed text and the resumed text, and following Bear, Dowding and Shriberg (1992), we annotate this using the labels **m** for word matching and **r** for word replacements (words of the same syntactic category). Each pair is given a unique index. Other words in the removed text and resumed text are annotated with an **x**. Also, editing terms (filled pauses and clue words) are labeled with **et**, and the interruption point with **int**, which will be before any editing terms associated with the repair, and after the fragment, if present. (Further details of our annotation scheme can be found in (Heeman and Allen, 1994a).) Below is a sample annotation, with removed text "go to oran-", editing term "um", and resumed text "go to".

```
go| to|  oran-| um| go| to| Corning
m1| m2| x| int| et| m1| m2|
```

Table 1 gives a breakdown of the modification speech repairs (that do not interfere with other repairs) and the abridged repairs, based on hand-annotations. Modification repairs are broken down into four groups, word repetitions, larger repetitions, one word replacing another, and others. Also, the percentage of repairs that include fragments and editing terms is also given. Two trends emerge from this data. First, fragments and editing terms mark less than 34% of all modification repairs. Second, the presence of a fragment or editing term does not give conclusive evidence as to whether the repair is a modification or an abridged repair.

4. Part-of-Speech Tagging

Part-of-speech tagging is the process of assigning to a word the category that is most probable given the sentential context (Church, 1988). The sentential context is typically approximated by only a set number of previous categories, usually one or two. Since the context is limited, we are making the Markov assumption, that the next transition depends only on the input, which is the word that we

[2] The removed text and editing terms might still contain pragmatical information, as the following example displays, "Peter was ... well ... he was fired."

are currently trying to tag and the previous categories. Good part-of-speech results can be obtained using only the preceding category (Weischedel et al., 1993), which is what we will be using. In this case, the number of states of the Markov model will be N, where N is the number of tags. By making the Markov assumption, we can use the Viterbi Algorithm to find a maximum probability path in linear time.

Figure 1 gives a simplied view of a Markov model for part-of-speech tagging, where C_i is a possible category for the ith word, w_i, and C_{i+1} is a possible category for word w_{i+1}. The category transition probability is simply the probability of category C_{i+1} following category C_i, which is written as $P(C_{i+1}|C_i)$, and the probability of word w_{i+1} given category C_{i+1} is $P(w_{i+1}|C_{i+1})$. The category assignment that maximizes the product of these probabilities is taken to be the best category assignment.

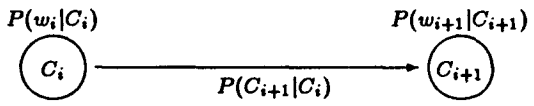

Figure 1: Markov Model of Part-of-Speech Tagging

5. A Simple Model of Speech Repairs

Modification repairs are often accompanied by a syntactic anomaly across the interruption point. Consider the following example, "so it takes two hours to go to – from Elmira to Corning" (d93-17.4 utt57), which contains a "to" followed by a "from". Both should be classified as prepositions, but the event of a preposition followed by another preposition is very rare in well-formed speech, so there is a good chance that one of the prepositions might get erroneously tagged as some other part of speech. Since the category transitions across interruption points tend to be rare events in fluent speech, we simply give the tagger the category transition probabilities around interruption points of modification repairs. By keeping track of when this information is used, we not only have a way of detecting modification repairs, but part-of-speech tagging is also improved.

To incorporate knowledge about modification repairs, we let R_i be a variable that indicates whether the transition from word w_i to w_{i+1} contains the interruption point of a modification repair, and rather than tag each word, w_i, with just a category, C_i, we will tag it with $R_{i-1}C_i$, the category and the presence of a modification repair.[3] This effectively multiplies the size of the tagset by two. From Figure 1, we see that we will now need the following probabilities, $P(R_iC_{i+1}|R_{i-1}C_i)$ and $P(w_i|R_{i-1}C_i)$.

To keep the model simple, and ease problems with sparse data, we make several independence assumptions.

(1) Given the category of a word, a repair before it is independent of the word. (R_{i-1} and w_i are independent, given C_i.) So $P(w_i|R_{i-1}C_i) = P(w_i|C_i)$.

(2) Given the category of a word, a repair before that word is independent of a repair following it and the category of the next word. (R_{i-1} is independent of R_iC_{i+1}, given C_i.) So $P(R_iC_{i+1}|R_{i-1}C_i) = P(R_iC_{i+1}|C_i)$.

[3]Changing each tag to C_iR_i would result in the same model.

One manipulation we can do is to use the definition of conditional probabilities to rewrite $P(R_iC_{i+1}|C_i)$ as $P(R_i|C_i) * P(C_{i+1}|C_iR_i)$. This manipulation allows us to view the problem as tagging null tokens between words as either the interruption point of a modification repair, $R_i = \tau_i$, or as fluent speech, $R_i = \phi_i$. The resulting Markov model is shown in Figure 2. Note that the context for category C_{i+1} is both C_i and R_i. So, R_i depends (indirectly) on the joint context of C_i and C_{i+1}, thus allowing syntactic anomalies to be detected.[4]

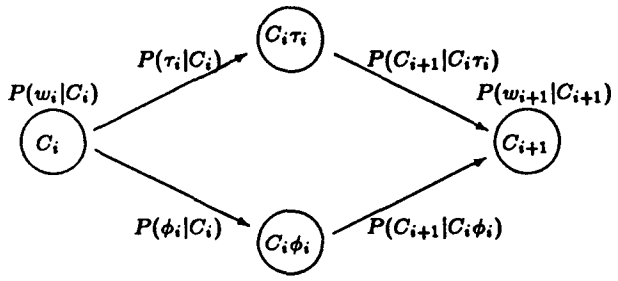

Figure 2: Markov Model of Repairs

Table 3 (Section 6.4) gives results for this simple model running on our training corpus. In order to remove effects due to editing terms and word fragments, we temporarily eliminate them from the corpus. Also, for fresh starts and change-of-turn, the algorithm is reset, as if it was an end of sentence. To eliminate problems due to overlapping repairs, we include only data points in which the next word is not intended to be removed (based on our hand annotations). This gives us a total of 19587 data points, 384 were modification repairs, and the statistical model found 169 of these, and a further 204 false positives. This gives us a recall rate of 44.2% and a precision of 45.3%. In the test corpus, there are 98 modification repairs, of which the model found 30, and a further 23 false positives; giving a recall rate of 30.6% and a precision rate of 56.6%.

From Table 1, we can see that the recall rate of fragments as a predictor of a modification repair is 14.7% and their precision is 34.7%.[5] So, the method of statistically tagging modification repairs has more predictive power, and so can be used as a clue for detecting them. Furthermore, this method is doing something more powerful than just detecting word repetitions or category repetitions. Of the 169 repairs that it found, 109 were word repetitions and an additional 28 were category repetitions. So, 32 of the repairs that were found were from less obvious syntactic anomalies.

6. Adding Additional Clues

In the preceding section we built a model for detecting modification repairs by simply using category transitions. However, there are other sources of information that can be exploited, such as the presence of fragments, editing terms, and word matchings. The problem is that

[4]Probabilities for fluent transitions are from the Brown corpus and probabilities for repair transitions are from the training data.

[5]The precision rate was calculated by taking the number of fragments in a modification repair (450 * 14.7%) over the total number of fragments (450 * 14.7% + 267 * 46.4%).

these clues do not always signal a modification repair. For instance, a fragment is twice as likely to be part of an abridged repair than it is to be part of a modification repair. One way to exploit these clues is to try to *learn* how to combine them, using a technique such as CART (Brieman, Friedman and Olshen, 1984). However, a more intuitive approach is to adjust the transition probabilities for a modification repair to better reflect the more specific information that is known. Thus, we combine the information such that the individual pieces do not have to give a 'yes' or a 'no', but rather, all can contribute to the decision.

6.1. Fragments

Assuming that fragments can be detected automatically (c.f. Nakatani and Hirschberg, 1993), the question arises as to what the tagger should do with them. If the tagger treats them as lexical items, the words on either side of the fragment will be separated. This will cause two problems. First, if the fragment is part of an abridged repair, category assignment to these words will be hindered. Second, and more important to our work, is that the fragment will prevent the statistical model from judging the syntactic well-formedness of the word before the fragment and the word after, preventing it from distinguishing a modification repair from an abridged repair. So, the tagger needs to skip over fragments. However, the fragment can be viewed as the "word" that gets tagged as a modification repair or not. (The 'not' in this case means that the fragment is part of an abridged repair.) When no fragment is present between words, we view the interval as a null word. So, we augment the model pictured in Figure 2 with the probability of the presence of a fragment, F_i, given the presence of a repair, R_i, as is pictured in Figure 3.

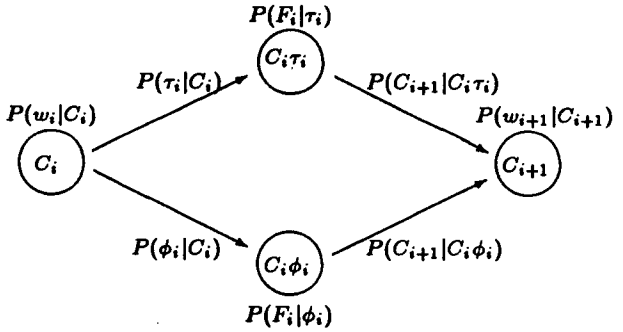

Figure 3: Incorporating Fragments

Since there are two alternatives for F_i—a fragment, f_i, or not, $\overline{f}_i$—and two alternatives for R_i—a repair or not, we need four statistics. From our training corpus, we have found that if a fragment is present, a modification repair is favored—$P(f_i|\tau_i)/P(f_i|\phi_i)$—by a factor of 28.9. If a fragment is not present, fluent speech is favored—$P(\overline{f}_i|\phi_i)/P(\overline{f}_i|\tau_i)$, by a factor of 1.17.

6.2. Editing Terms

Editing terms, like fragments, give information as to the presence of a modification repair. So, we incorporate them into the statistical model by viewing them as part of the "word" that gets tagged with R_i, thus changing the probability on the repair state from $P(F_i|R_i)$ to $P(F_iE_i|R_i)$, where E_i indicates the presence of editing terms. To simplify the probabilities, and reduce problems due to sparse data, we make the following independence assumption.

(3) Given that there is a modification repair, the presence of a fragment or editing terms is independent. (F_i and E_i are independent, given R_i.) So $P(F_iE_i|R_i) = P(F_i|R_i) * P(E_i|R_i)$.

An additional complexity is that different editing terms do not have the same predictive power. So far we have investigated "um" and "uh". The presence of an "um" favors a repair by a factor of 2.7, while for "uh" it is favored by a factor of 9.4. If no editing term is present, fluent speech is favored by a factor of 1.2.

6.3. Word Matchings

In a modification repair, there is often a correspondence between the text that must be removed and the text that follows the interruption point. The simplest type of correspondence is word matchings. In fact, in our test corpus, 80% of modification repairs have at least one matching. This information can be incorporated into the statistical model in the same way that editing terms and fragments are handled. So, we change the probability of the repair state to be $P(F_iE_iM_i|R_i)$, where M_i indicates a word matching. Again, we assume that the clues are independent of each other, allowing us to treat this clue separately from the others.

Just as with editing terms, not all matches make the same predictions about the occurrence of a modification repair. Bear, Dowding and Shriberg (1992) looked at the number of matching words versus the number of intervening words. However, this ignores the category of the word matches. For instance, a matching verb (with some intervening words) is more likely to indicate a repair than say a matching preposition or determiner. So, we classify word matchings by category and number of intervening words. Furthermore, if there are multiple matches in a repair, we only use one, the one that most predicts a repair. For instance in the following repair, the matching instances of "take" would be used over the matching instances of "will", since main verbs were found to more strongly signal a modification repair than do modals.

> how long will that take – will it take for engine one at Dansville (d93-18.3 utt43)

Since the statistical model only uses one matching per repair, the same is done in collecting the statistics. So, our collection involves two steps. In the first we collect statistics on all word matches, and in the second, for each repair, we count only the matching that most strongly signals the repair. Table 2 gives a partial list of how much each matching favors a repair broken down by category and number of intervening words. Entries that are marked with "–" do not contain any datapoints and entries that are blank are below the baseline rate of 0.209, the rate at which a modification repair is favored (or actually disfavored) when there is no matching at all.

The problem with using word matching is that it depends on identifying the removed text and its correspondences to the text that follows the interruption point. However, a good estimate can be obtained by using all word matches with at most eight intervening words.

6.4. Results

Table 3 summarizes the results of incorporating additional clues into the Markov model. The first column gives the results without any clues, the second with fragments, the third with editing terms,

Cat	Number of Intervening Words					
	0	1	2	3	4	5
DT	935.5	38.5	2.7	2.2	0.7	0.8
IN	–	171.7	59.6	22.9	10.4	6.3
IS	490.0	55.8	5.9	3.2		
MD	–	6706.5	199.8	37.1	12.4	2.4
NN	–	68.0	32.2	10.4	0.3	0.2
NNP	144.3	9.2	6.2	6.7	3.3	2.8
PREP	16433.6	2.8				
PRP	8242.3	15.2	2.9	1.2	0.5	
RB	25.2	19.4	6.9	6.4	3.9	3.6
TO	5170.7	1.6	0.5	0.4		
VB	5170.6	216.3	71.5	31.2	18.1	7.0

Table 2: Factor by which a repair is favored

the fourth with word matches, and the fifth, with all of these clues incorporated. Of the 384 modification repairs in the training corpus, the full model predicts 305 of them versus 169 by the simple model. As for the false positives, the full model incorrectly predicted 207 versus the simple model at 204. So, we see that by incorporating additional clues, the statistical model can better identify modification repairs.

	Simple Model	Fragments	Edit Terms	Word Match	Full
Training:					
Recall	44.0%	50.0%	45.1%	76.5%	79.4%
Precision	45.3%	47.8%	46.5%	54.9%	59.6%
Testing:					
Recall	30.6%	43.9%	32.7%	74.5%	76.5%
Precision	56.6%	62.3%	59.3%	58.4%	62.0%

Table 3: Results of Markov Models

7. Correcting Repairs

The actual goal of detecting speech repairs is to be able to correct them, so that the speaker's utterance can be understood. We have argued for the need to distinguish modification repairs from abridged repairs, because this distinction would be useful in determining the correction. We have implemented a pattern builder (Heeman and Allen, 1994b), which builds potential repair patterns based on word matches and word replacements. However, the pattern builder has only limited knowledge which it can use to decide which patterns are likely repairs. For instance, given the utterance "pick up uh fill up the boxcars" (d93-17.4 utt40), it will postulate that there is a single repair, in which "pick up" is replaced by "fill up". However, for an utterance like "we need to um manage to get the bananas" (d93-14.3 utt50), it will postulate that "manage to" replaces "need to". So, we use the statistical model to filter repairs found by the pattern builder. This also removes a lot of the false positives of the statistical model, since no potential repair pattern would be found for them. On the training set, the model was queried by the pattern builder on 961 potential modification repairs, of which 397 contained repairs. The model predicted 365 of these, and incorrectly detected 33 more, giving a detection recall rate of 91.9% and a precision of 91.7%. For the test corpus, it achieved a recall rate of 83.0% and a precision of 80.2%.

The true measure of success is the overall detection and correction rates. On 721 repairs in the training corpus, which includes overlapping repairs, the combined approach made the right corrections for 637, it made incorrect corrections for 19 more, and it falsely detected (and falsely corrected) 30 more. This gives an overall correction recall rate of 88.3% and a precision of 92.9%. On the test corpus consisting of 142 repairs, it made the right correction for 114 of them, it incorrectly corrected 4 more, and it falsely detected 14 more, for a correction recall rate of 80.3% and a precision of 86.4%. Table 4 summarizes the overall results for both the pattern builder and statistical model on the training corpus and on the test set.

	Training Corpus	Test Corpus
Detection		
Recall	91%	83%
Precision	96%	89%
Correction		
Recall	88%	80%
Precision	93%	86%

Table 4: Overall Results

The results that we obtained are better than others reported in the literature. However, such comparisons are limited due to differences in both the type of repairs that are being studied and in the datasets used for drawing results. Bear, Dowding, and Shriberg (1992) use the ATIS corpus, which is a collection of queries made to an automated airline reservation system. As stated earlier, they removed all utterances that contained abridged repairs. For detection they obtained a recall rate of 76% and a precision of 62%, and for correction, a recall rate of 43% and a precision of 50%. It is not clear whether their results would be better or worse if abridged repairs were included. Dowding et al. (1993) used a similar setup for their data. As part of a complete system, they obtained a detection recall rate of 42% and a precision of 85%; and for correction, a recall rate of 30% and a precision of 62%. Lastly, Nakatani and Hirschberg (1993) also used the ATIS corpus, but in this case, focused only on detection, but detection of all three types of repairs. However, their test corpus consisted entirely of utterances that contained at least one repair. This makes it hard to evaluate their results, reporting a detection recall rate of 83% and precision of 94%. Testing on an entire corpus would clearly decrease their precision. As for our own data, we used a corpus of natural dialogues that were segmented only by speaker turns, not by individual utterances, and we focused on modification repairs and abridged repairs, with fresh starts being marked in the input so as not to cause interference in detecting the other two types.

8. Discussion

We have described a statistical model for detecting speech repairs. The model detects repairs by using category transition probabilities around repair intervals and for fluent speech. By training on actual examples of repairs, we can detect them without having to set arbitrary cutoffs for category transitions that might be insensitive to rarely used constructs. If people actually use syntactic anomalies as a clue in detecting speech repairs, then training on examples of them

makes sense.

In doing this work, we were faced with a lack of training data. The eventual answer is to have a large corpus of tagged dialogs with the speech repairs annotated. Since this was not available, we used the Brown corpus for the fluent category-transition probabilities. As well, these transition probabilities were used to 'bootstrap' our tagger in determining the part-of-speech tags for our training corpus. The tags of the 450 or so hand-annotated modification repairs were then used for setting the transition probabilities around modification repairs.

Another problem that we encountered was interference between adjacent utterances in the same turn. Subsequent utterances often build on, or even repeat what was previously said (Walker, 1993). Consider the following utterance.

> that's all you need
> you only need one tanker (d93-8.3 utt79)

The tagger incorrectly hypothesized that this was a modification repair with an interruption point after the first occurrence of the word "need". Even a relatively simple segmentation of the dialogs into utterances would remove some of the false positives and improve performance.

Speech repairs do interact negatively with part-of-speech tagging, and even with statistical modeling of repairs, inappropriate tags are still sometimes assigned. In the following example, the second occurrence of the word "load" was categorized as a noun, and the speech repair went undetected.

> it'll be seven a.m. by the time we load in – load the bananas (d93-12.4 utt53)

9. Conclusions

This paper described a method of detecting repairs that uses a part-of-speech tagger. Our work shows that a large percentage of speech repairs can be detected and corrected prior to parsing. Prosodic clues can be easily incorporated into our statistical model, and we are currently investigating methods of automatically extracting simple prosodic features in order to further improve the performance of the algorithm.

Our algorithm assumes that the speech recognizer produces a sequence of words and identifies the presence of word fragments. With the exception of identifying fresh starts, all other processing is automatic and does not require additional hand-tailored transcription. We will be incorporating this method of detecting and correcting speech repairs into the next version of the TRAINS system, which will use spoken input.

10. Acknowledgments

We wish to thank Bin Li, Greg Mitchell, and Mia Stern for their help in both transcribing dialogs and giving us useful comments on the annotation scheme. We also wish to thank Hannah Blau, Elizabeth Shriberg, and David Traum for enlightening conversations. Funding gratefully received from the Natural Sciences and Engineering Research Council of Canada, from NSF under Grant IRI-90-13160, and from ONR/DARPA under Grant N00014-92-J-1512.

References

Allen, J. F. and Schubert, L. K. (1991). The TRAINS project. Technical Report 382, Department of Computer Science, University of Rochester.

Bear, J., Dowding, J., and Shriberg, E. (1992). Integrating multiple knowledge sources for detection and correction of repairs in human-computer dialog. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, pages 56–63.

Brieman, L., Friedman, J. H., and Olshen, R. A. (1984). *Classification and Regression Trees*. Wadsworth & Brooks, Monterrey, CA.

Church, K. (1988). A stochastic parts program and noun phrase parser for unrestricted text. In *Preceedings of the 2nd Conference on Applied Natural Language Processing*, pages 136–143.

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. In *Proceedings of the 31th Annual Meeting of the Association for Computational Linguistics*, pages 54–61.

Gross, D., Allen, J., and Traum, D. (1992). The TRAINS 91 dialogues. Trains Technical Note 92-1, Department of Computer Science, University of Rochester.

Heeman, P. A. and Allen, J. (1994a). Annotating speech repairs. unpublished manuscript.

Heeman, P. A. and Allen, J. (1994b). Detecting and correcting speech repairs. To appear in the 31th Meeting of the Association for Computational Linguistics.

Heeman, P. A. and Allen, J. (1994c). Dialogue transcription tools. unpublished manuscript.

Hindle, D. (1983). Deterministic parsing of syntactic non-fluencies. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pages 123–128.

Levelt, W. J. M. (1983). Monitoring and self-repair in speech. *Cognition*, 14:41–104.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of english: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Nakatani, C. and Hirschberg, J. (1993). A speech-first model for repair detection and correction. In *Proceedings of the 31th Annual Meeting of the Association for Computational Linguistics*, pages 46–53.

Walker, M. A. (1993). Informational redundancy and resource bounds in dialogue. Doctoral dissertion, Institute for Research in Cognitive Science report IRCS-93-45, University of Pennsylvania.

Wang, M. Q. and Hirschberg, J. (1992). Automatic classification of intonational phrase boundaries. *Computer Speech and Language*, 6:175–196.

Weischedel, R., Meteer, M., Schwartz, R., Ramshaw, L., and Palmucci, J. (1993). Coping with ambiguity and unknown words through probabilistic models. *Computational Linguistics*, 19(2):359–382.