Intonational Boundaries, Speech Repairs and Discourse Markers: Modeling Spoken Dialog*

Peter A. Heeman and James F. Allen
Department of Computer Science
University of Rochester
Rochester NY 14627, USA
{heeman,james}@cs.rochester.edu

Abstract

To understand a speaker's turn of a conversation, one needs to segment it into intonational phrases, clean up any speech repairs that might have occurred, and identify discourse markers. In this paper, we argue that these problems must be resolved together, and that they must be resolved early in the processing stream. We put forward a statistical language model that resolves these problems, does POS tagging, and can be used as the language model of a speech recognizer. We find that by accounting for the interactions between these tasks that the performance on each task improves, as does POS tagging and perplexity.

1 Introduction

Interactive spoken dialog provides many new challenges for natural language understanding systems. One of the most critical challenges is simply determining the speaker's intended utterances: both segmenting the speaker's turn into utterances and determining the intended words in each utterance. Since there is no well-agreed to definition of what an utterance is, we instead focus on intonational phrases (Silverman et al., 1992), which end with an acoustically signaled *boundary tone*. Even assuming perfect word recognition, the problem of determining the intended words is complicated due to the occurrence of speech repairs, which occur where the speaker goes back and changes (or repeats) something she just said. The words that are replaced or repeated are no longer part of the intended utterance, and so need to be identified. The following example, from the Trains corpus (Heeman and Allen, 1995), gives an example of a speech repair with the words that the speaker intends to be replaced marked by *reparandum*, the words that are the intended replacement marked as *alteration*, and the cue phrases and filled pauses that tend to occur in between marked as the *editing term*.

Example 1 (d92a-5.2 utt34)
we'll pick up a tank of uh the tanker of oranges
reparandum | *editing term* *alteration*
interruption point

Much work has been done on both detecting boundary tones (e.g. (Wang and Hirschberg, 1992; Wightman and Ostendorf, 1994; Stolcke and Shriberg, 1996a; Kompe et al., 1994; Mast et al., 1996)) and on speech repair detection and correction (e.g. (Hindle, 1983; Bear, Dowding, and Shriberg, 1992; Nakatani and Hirschberg, 1994; Heeman and Allen, 1994; Stolcke and Shriberg, 1996b)). This work has focused on one of the issues in isolation of the other. However, these two issues are intertwined. Cues such as the presence of silence, final syllable lengthening, and presence of filled pauses tend to mark both events. Even the presence of word correspondences, a tradition cue for detecting and correcting speech repairs, sometimes marks boundary tones as well, as illustrated by the following example where the intonational phrase boundary is marked with the ToBI symbol %.

Example 2 (d93-83.3 utt73)
that's all you need % you only need one boxcar

Intonational phrases and speech repairs also interact with the identification of discourse markers. Discourse markers (Schiffrin, 1987; Hirschberg and Litman, 1993; Byron and Heeman, 1997) are used to relate new speech to the current discourse state. Lexical items that can function as discourse markers, such as "well" and "okay," are ambiguous as to whether they are being used as discourse markers or not. The complication is that discourse markers tend to be used to introduce a new utterance, or can be an utterance all to themselves (such as the acknowledgment "okay" or "alright"), or can be used as part of the editing term of a speech repair, or to begin the alteration. Hence, the problem of identifying discourse markers also needs to be addressed with the segmentation and speech repair problems.

*In Proceedings of ACL/EACL'97

These three phenomena of spoken dialog, however, cannot be resolved without recourse to syntactic information. Speech repairs, for example, are often signaled by syntactic anomalies. Furthermore, in order to determine the extent of the reparandum, one needs to take into account the parallel structure that typically exists between the reparandum and alteration, which relies on at identifying the syntactic roles, or part-of-speech (POS) tags, of the words involved (Bear, Dowding, and Shriberg, 1992; Heeman and Allen, 1994). However, speech repairs disrupt the context that is needed to determine the POS tags (Hindle, 1983). Hence, speech repairs, as well as boundary tones and discourse markers, must be resolved during syntactic disambiguation.

Of course when dealing with spoken dialogue, one cannot forget the initial problem of determining the actual words that the speaker is saying. Speech recognizers rely on being able to predict the probability of what word will be said next. Just as intonational phrases and speech repairs disrupt the local context that is needed for syntactic disambiguation, the same holds for predicting what word will come next. If a speech repair or intonational phrase occurs, this will alter the probability estimate. But more importantly, speech repairs and intonational phrases have acoustic correlates such as the presence of silence. Current speech recognition language models cannot account for the presence of silence, and tend to simply ignore it. By modeling speech repairs and intonational boundaries, we can take into account the acoustic correlates and hence use more of the available information.

From the above discussion, it is clear that we need to model these dialogue phenomena together and very early on in the speech processing stream, in fact, during speech recognition. Currently, the approaches that work best in speech recognition are statistical approaches that are able to assign probability estimates for what word will occur next given the previous words. Hence, in this paper, we introduce a statistical language model that can detect speech repairs, boundary tones, and discourse markers, and can assign POS tags, and can use this information to better predict what word will occur next.

In the rest of the paper, we first introduce the Trains corpus. We then introduce a statistical language model that incorporates POS tagging and the identification of discourse markers. We then augment this model with speech repair detection and correction and intonational boundary tone detection. We then present the results of this model on the Trains corpus and show that it can better account for these discourse events than can be achieved by modeling them individually. We also show that by modeling these two phenomena that we can increase our POS tagging performance by 8.6%, and improve our ability to predict the next word.

Dialogs	98
Speakers	34
Words	58298
Turns	6163
Discourse Markers	8278
Boundary Tones	10947
Turn-Internal Boundary Tones	5535
Abridged Repairs	423
Modification Repairs	1302
Fresh Starts	671
Editing Terms	1128

Table 1: Frequency of Tones, Repairs and Editing Terms in the Trains Corpus

2 Trains Corpus

As part of the TRAINS project (Allen et al., 1995), which is a long term research project to build a conversationally proficient planning assistant, we have collected a corpus of problem solving dialogs (Heeman and Allen, 1995). The dialogs involve two human 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. The collection methodology was designed to make the setting as close to human-computer interaction as possible, but was not a *wizard* scenario, where one person pretends to be a computer. Rather, the user knows that he is talking to another person.

The TRAINS corpus consists of about six and half hours of speech. Table 1 gives some general statistics about the corpus, including the number of dialogs, speakers, words, speaker turns, and occurrences of discourse markers, boundary tones and speech repairs.

The speech repairs in the Trains corpus have been hand-annotated. We have divided the repairs into three types: *fresh starts*, *modification repairs*, and *abridged repairs*.[1] A fresh start is where the speaker abandons the current utterance and starts again, where the abandonment seems acoustically signaled.

Example 3 (d93-12.1 utt30)

so it'll take *reparandum* | *interruption point* um *editing term* so you want to do what *alteration*

The second type of repairs are the modification repairs. These include all other repairs in which the reparandum is not empty.

Example 4 (d92a-1.3 utt65)

so that will total *reparandum* | will take *alteration* *interruption point* seven hours to do that

[1] This classification is similar to that of Hindle (1983) and Levelt (1983).

The third type of repairs are the abridged repairs, which consist solely of an editing term. Note that utterance initial filled pauses are not treated as abridged repairs.

Example 5 (d93-14.3 utt42)

we need to $\overbrace{\text{um}}^{\textit{editing term}}$ manage to get the bananas to Dansville

interruption point

There is typically a correspondence between the reparandum and the alteration, and following Bear *et al.* (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 reparandum and alteration are annotated with an **x**. Also, editing terms (filled pauses and clue words) are labeled with **et**, and the interruption point with **ip**, which will occur before any editing terms associated with the repair, and after a word fragment, if present. The interruption point is also marked as to whether the repair is a fresh start, modification repair, or abridged repair, in which cases, we use **ip:can**, **ip:mod** and **ip:abr**, respectively. The example below illustrates how a repair is annotated in this scheme.

Example 6 (d93-15.2 utt42)

engine two from Elmi(ra)- or engine three from Elmira

m1 r2 m3 m4 ↑ **et m1 r2 m3 m4**

ip:mod

3 A POS-Based Language Model

The goal of a speech recognizer is to find the sequence of words $\hat{W}$ that is maximal given the acoustic signal A. However, for detecting and correcting speech repairs, and identifying boundary tones and discourse markers, we need to augment the model so that it incorporates shallow statistical analysis, in the form of POS tagging. The POS tagset, based on the Penn Treebank tagset (Marcus, Santorini, and Marcinkiewicz, 1993), includes special tags for denoting when a word is being used as a discourse marker. In this section, we give an overview of our basic language model that incorporates POS tagging. Full details can be found in (Heeman and Allen, 1997; Heeman, 1997).

To add in POS tagging, we change the goal of the speech recognition process to find the best word and POS tags given the acoustic signal. The derivation of the acoustic model and language model is now as follows.

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

The first term $\Pr(A|WP)$ is the factor due to the acoustic model, which we can approximate by $\Pr(A|W)$. The second term $\Pr(WP)$ is the factor due to the language model. We rewrite $\Pr(WP)$ as $\Pr(W_{1,N}P_{1,N})$, where N is the number of words in the sequence. We now rewrite the language model probability as follows.

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

We now have two probability distributions that we need to estimate, which we do using decision trees (Breiman et al., 1984; Bahl et al., 1989). The decision tree algorithm has the advantage that it uses information theoretic measures to construct equivalence classes of the context in order to cope with sparseness of data. The decision tree algorithm starts with all of the training data in a single leaf node. For each leaf node, it looks for the question to ask of the context such that splitting the node into two leaf nodes results in the biggest decrease in *impurity*, where the impurity measures how well each leaf predicts the events in the node. After the tree is grown, a heldout dataset is used to smooth the probabilities of each node with its parent (Bahl et al., 1989).

To allow the decision tree to ask about the words and POS tags in the context, we cluster the words and POS tags using the algorithm of Brown *et al.* (1992) into a binary classification tree. This gives an implicit binary encoding for each word and POS tag, thus allowing the decision tree to ask about the words and POS tags using simple binary questions, such as 'is the third bit of the POS tag encoding equal to one?' Figure 1 shows a POS classification tree. The binary encoding for a POS tag is determined by the sequence of top and bottom edges that leads from the root node to the node for the POS tag.

Unlike other work (e.g. (Black et al., 1992; Magerman, 1995)), we treat the word identities as a further refinement of the POS tags; thus we build a word classification tree for each POS tag. This has the advantage of avoiding unnecessary data fragmentation, since the POS tags and word identities are no longer separate sources of information. As well, it constrains the task of building the word classification trees since the major distinctions are captured by the POS classification tree.

4 Augmenting the Model

Just as we redefined the speech recognition problem so as to account for POS tagging and identifying discourse markers, we do the same for modeling

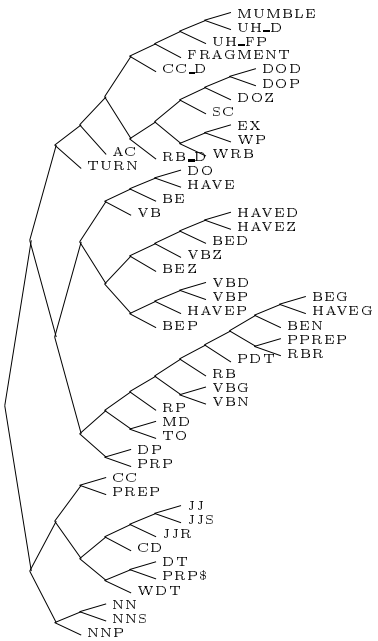

Figure 1: POS Classification Tree

boundary tones and speech repairs. We introduce null tokens between each pair of consecutive words $W_{i\text{-}1}$ and W_i (Heeman and Allen, 1994), which will be tagged as to the occurrence of these events. The boundary tone tag T_i indicates if word $w_{i\text{-}1}$ ends an intonational boundary (T_i=**T**), or not (T_i=**null**).

For detecting speech repairs, we have the problem that repairs are often accompanied by an editing term, such as "um", "uh", "okay", or "well", and these must be identified as such. Furthermore, an editing term might be composed of a number of words, such as "let's see" or "uh well". Hence we use two tags: an editing term tag E_i and a repair tag R_i. The editing term tag indicates if W_i starts an editing term (E_i=**Push**), if W_i continues an editing term (E_i=**ET**), if $w_{i\text{-}1}$ ends an editing term (E_i=**Pop**), or otherwise (E_i=**null**). The repair tag R_i indicates whether word W_i is the onset of the alteration of a fresh start (R_i=**C**), a modification repair (R_i=**M**), or an abridged repair (R_i=**A**), or there is not a repair (R_i=**null**). Note that for repairs with an editing term, the repair is tagged after the extent of the editing term has been determined. Below we give an example showing all non-null tone, editing term and repair tags.

Example 7 (d93-18.1 utt47)
it takes one **Push** you **ET** know **Pop** **M** two hours **T**

If a modification repair or fresh start occurs, we need to determine the extent (or the onset) of the reparandum, which we refer to as *correcting* the speech repair. Often, speech repairs have strong word correspondences between the reparandum and alteration, involving word matches and word replacements. Hence, knowing the extent of the reparandum means that we can use the reparandum to predict the words (and their POS tags) that make up the alteration. For $R_i \in \{\mathbf{Mod}, \mathbf{Can}\}$, we define O_i to indicate the onset of the reparandum.[2]

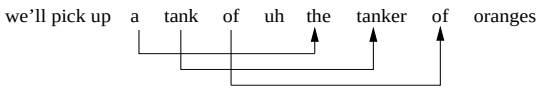

Figure 2: Cross Serial Correspondences

If we are in the midst of processing a repair, we need to determine if there is a word correspondence from the reparandum to the current word W_i. The tag L_i is used to indicate which word in the reparandum is licensing the correspondence. Word correspondences tend to exhibit a cross serial dependency; 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, as illustrated in Figure 2. This means that if W_i involves a word correspondence, it will most likely be with a word that follows the last word in the reparandum that has a word correspondence. Hence, we restrict L_i to only those words that are after the last word in the reparandum that has a correspondence (or from the reparandum onset if there is not yet a correspondence). If there is no word correspondence for W_i, we set L_i to the first word after the last correspondence.

The second tag involved in the correspondences is C_i, which indicates the type of correspondence between the word indicated by L_i and the current word W_i. We focus on word correspondences that involve either a word match (C_i=**m**), a word replacement (C_i=**r**), where both words are of the same POS tag, or no correspondence (C_i=**x**).

Now that we have defined these six additional tags for modeling boundary tones and speech repairs, we redefine the speech recognition problem so that its goal is to find the maximal assignment for the words as well as the POS, boundary tone, and speech repair tags.

$$\hat{W}\hat{P}\hat{C}\hat{L}\hat{O}\hat{R}\hat{E}\hat{T} = \arg\max_{WPCLORET} \Pr(WCLORET|A)$$

The result is that we now have eight probability distributions that we need to estimate.

$$\Pr(T_i|W_{1,i\text{-}1}P_{1,i\text{-}1}C_{1,i\text{-}1}L_{1,i\text{-}1}O_{1,i\text{-}1}R_{1,i\text{-}1}E_{1,i\text{-}1}T_{1,i\text{-}1})$$

$$\Pr(E_i|W_{1,i\text{-}1}P_{1,i\text{-}1}C_{1,i\text{-}1}L_{1,i\text{-}1}O_{1,i\text{-}1}R_{1,i\text{-}1}E_{1,i\text{-}1}T_{1,i})$$

$$\Pr(R_i|W_{1,i\text{-}1}P_{1,i\text{-}1}C_{1,i\text{-}1}L_{1,i\text{-}1}O_{1,i\text{-}1}R_{1,i\text{-}1}E_{1,i}T_{1,i})$$

$$\Pr(O_i|W_{1,i\text{-}1}P_{1,i\text{-}1}C_{1,i\text{-}1}L_{1,i\text{-}1}O_{1,i\text{-}1}R_{1,i}E_{1,i}T_{1,i})$$

[2]Rather than estimate O_i directly, we instead query each potential onset to see how likely it is to be the actual onset of the reparandum.

$$\Pr(L_i|W_{1,i\text{-}1}P_{1,i\text{-}1}C_{1,i\text{-}1}L_{1,i\text{-}1}O_{1,i}R_{1,i}E_{1,i}T_{1,i})$$

$$\Pr(C_i|W_{1,i\text{-}1}P_{1,i\text{-}1}C_{1,i\text{-}1}L_{1,i}O_{1,i}R_{1,i}E_{1,i}T_{1,i})$$

$$\Pr(P_i|W_{1,i\text{-}1}P_{1,i\text{-}1}C_{1,i}L_{1,i}O_{1,i}R_{1,i}E_{1,i}T_{1,i})$$

$$\Pr(W_i|W_{1,i\text{-}1}P_{1,i}C_{1,i}L_{1,i}O_{1,i}R_{1,i}E_{1,i}T_{1,i})$$

The context for each of the probability distributions includes all of the previous context. In principal, we could give all of this context to the decision tree algorithm and let it decide what information is relevant in constructing equivalence classes of the contexts. However, the amount of training data is limited (as are the learning techniques) and so we need to encode the context in order to simplify the task of constructing meaningful equivalence classes. We start with the words and their POS tags that are in the context and for each non-null tone, editing term (we also skip over E=**ET**), and repair tag, we insert it into the appropriate place, just as Kompe *et al.* (1994) do for boundary tones in their language model. Below we give the encoded context for the word "know" from Example 7

Example 8 (d93-18.1 utt47)
it/**PRP** takes/**VBP** one/**CD Push** you/**PRP**

The result of this is that the non-null tag values are treated just as if they were lexical items.[3] Furthermore, if an editing term is completed, or the extent of a repair is known, we can also clean up the editing term or reparandum, respectively, in the same way that Stolcke and Shriberg (1996b) clean up filled pauses, and simple repair patterns. This means that we can then generalize between fluent speech and instances that have a repair. For instance, in the two examples below, the context for the word "get" and its POS tag will be the same for both, namely "so/**CC_D** we/**PRP** need/**VBP** to/**TO**".

Example 9 (d93-11.1 utt46)
so we need to get the three tankers

Example 10 (d92a-2.2 utt6)
so we need to **Push** um **Pop A** get a tanker of OJ

We also include other features of the context. For instance, we include a variable to indicate if we are currently processing an editing term, and whether a non-filled pause editing term was seen. For estimating R_i, we include the editing terms as well. For estimating O_i, we include whether the proposed reparandum includes discourse markers, filled pauses that are not part of an editing term, boundary terms, and whether the proposed reparandum overlaps with any previous repair.

[3]Since we treat the non-null tags as lexical items, we associate a unique POS tag with each value.

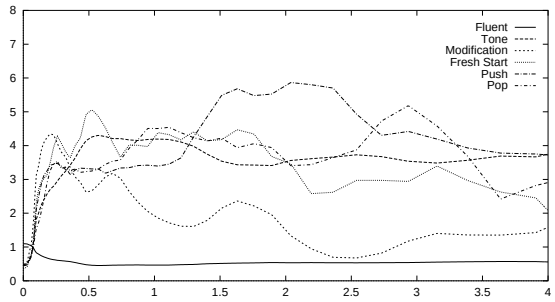

Figure 3: Preference for tone, editing term, and repair tags given the length of silence

5 Silences

Silence, as well as other acoustic information, can also give evidence as to whether an intonational phrase, speech repair, or editing term occurred. We include S_i, the silence duration between word $w_{i\text{-}1}$ and W_i, as part of the context for conditioning the probability distributions for the tone T_i, editing term E_i, and repair R_i tags. Due to sparseness of data, we make several the independence assumptions so that we can separate the silence information from the rest of the context. For example, for the tone tag, let $Rest_i$ represent the rest of the context that is used to condition T_i. By assuming that $Rest_i$ and S_i are independent, and are independent given T_i, we can rewrite $\Pr(T_i|S_i Rest_i)$ as follows.

$$\Pr(T_i|S_i Rest_i) = \Pr(T_i|Rest_i)\frac{\Pr(T_i|S_{i\text{-}1})}{\Pr(T_i)}$$

We can now use $\frac{\Pr(T_i|S_i)}{\Pr(T_i)}$ as a factor to modify the tone probability in order to take into account the silence duration. In Figure 3, we give the factors by which we adjust the tag probabilities given the amount of silence. Again, due to sparse of data, we collapse the values of the tone, editing term and repair tag into six classes: boundary tones, editing term pushes, editing term pops, modification repairs and fresh starts (without an editing term). From the figure, we see that if there is no silence between W_{i-1} and W_i, the null interpretation for the tone, repair and editing term tags is preferred. Since the independence assumptions that we have to make are too strong, we normalize the adjusted tone, editing term and repair tag probabilities to ensure that they sum to one over all of the values of the tags.

6 Example

To demonstrate how the model works, consider the following example.

Example 11 (d92a-2.1 utt95)
will take a total of um let's see total of s- of 7 hours
reparandum *ip* *et* *reparandum* *ip*

The language model considers all possible interpretations (at least those that do not get pruned) and assigns a probability to each. Below, we give the probabilities for the correct interpretation of the word "um", given the correct interpretation of the words "will take a total of". For reference, we give a simplified view of the context that is used for each probability.

Pr(T_6=**null**|a total of)=0.98
Pr(E_6=**Push**|a total of)=0.28
Pr(R_6=**null**|a total of **Push**)=1.00
Pr(P_6=**UH_FP**|a total of **Push**)=0.75
Pr(W_6=um|a total of **Push UH_FP**)=0.33

Given the correct interpretation of the previous words, the probability of the filled pause "um" along with the correct POS tag, boundary tone tag, and repair tags is 0.0665.

Now lets consider predicting the second instance of "total", which is the first word of the alteration of the first repair, whose editing term "um let's see", which ends with a boundary tone, has just finished.

Pr(T_{10}=**T**|**Push** let's see)=0.93
Pr(E_{10}=**Pop**|**Push** let's see **Tone**)=0.79
Pr(R_{10}=**M**|a total of **Push** let's see **Pop**) = 0.26
Pr(O_{10}=total|will take a total of R_{10}=**M**)=0.07
Pr(L_{10}=total|total of R_{10}=**M**)=0.94
Pr(C_{10}=**m**|will take a L_{10}=total/**NN**) = 0.87 4
Pr(P_{10}=**NN**|will take a L_{10}=total/**NN** C_{10}=**m**)=1
Pr(W_{10}=total|will take a **NN** L_{10}=total C_{10}=**m**)=1

Given the correct interpretation of the previous words, the probability of the word "total" along with the correct POS tag, boundary tone tag, and repair tags is 0.011.

7 Results

To demonstrate our model, we use a 6-fold cross validation procedure, in which we use each sixth of the corpus for testing data, and the rest for training data. We start with the word transcriptions of the Trains corpus, thus allowing us to get a clearer indication of the performance of our model without having to take into account the poor performance of speech recognizers on spontaneous speech. All silence durations are automatically obtained from a word aligner (Ent, 1994).

Table 2 shows how POS tagging, discourse marker identification and perplexity benefit by modeling the speaker's utterance. The POS tagging results are reported as the percentage of words that were assigned the wrong tag. The detection of discourse markers is reported using recall and precision. The recall rate of X is the number of X events that were correctly determined by the algorithm over the number of occurrences of X. The precision rate is the number of X events that were correctly determined over the number of times that the algorithm guessed X. The error rate is the number of X events that the algorithm missed plus the number of X events that it incorrectly guessed as occurring over the number of X events. The last measure is *perplexity*, which is a way of measuring how well the language model is able to predict the next word. The perplexity of a test set of N words $w_{1,N}$ is calculated as follows.

$$2^{-\frac{1}{N}\sum_{i=1}^{N}\log_2 \Pr(w_i|w_{1,i-1})}$$

	Base Model	Tones Repairs Corrections	Tones Repairs Corrections Silences
POS Tagging			
Error Rate	2.95	2.86	2.69
Discourse Markers			
Recall	96.60	96.60	97.14
Precision	95.76	95.86	96.31
Error Rate	7.67	7.56	6.57
Perplexity	24.35	23.05	22.45

Table 2: POS Tagging and Perplexity Results

The second column of Table 2 gives the results of the POS-based model, the third column gives the results of incorporating the detection and correction of speech repairs and detection of intonational phrase boundary tones, and the fourth column gives the results of adding in silence information. As can be seen, modeling the user's utterances improves POS tagging, identification of discourse markers, and word perplexity; with the POS error rate decreasing by 3.1% and perplexity by 5.3%. Furthermore, adding in silence information to help detect the boundary tones and speech repairs results in a further improvement, with the overall POS tagging error rate decreasing by 8.6% and reducing perplexity by 7.8%. In contrast, a word-based trigram backoff model (Katz, 1987) built with the CMU statistical language modeling toolkit (Rosenfeld, 1995) achieved a perplexity of 26.13. Thus our full language model results in 14.1% reduction in perplexity.

Table 3 gives the results of detecting intonational boundaries. The second column gives the results of adding the boundary tone detection to the POS model, the third column adds silence information, and the fourth column adds speech repair detection and correction. We see that adding in silence information gives a noticeable improvement in detecting boundary tones. Furthermore, adding in the speech repair detection and correction further improves the results of identifying boundary tones. Hence to detect intonational phrase boundaries in spontaneous speech, one should also model speech repairs.

Table 4 gives the results of detecting and correcting speech repairs. The detection results report the number of repairs that were detected, regardless of

	Tones	Tones Silences	Tones Repairs Corrections Silences
Within Turn			
Recall	64.9	70.2	70.5
Precision	67.4	68.7	69.4
Error Rate	66.5	61.9	60.5
All Tones			
Recall	80.9	83.5	83.9
Precision	81.0	81.3	81.8
Error Rate	38.0	35.7	34.8
Perplexity	24.12	23.78	22.45

Table 3: Detecting Intonational Phrases

	Repairs	Repairs Silences	Repairs Corrections Silences	Tones Repairs Corrections Silences
Detection				
Recall	67.9	72.7	75.7	77.0
Precision	80.6	77.9	80.8	84.8
Error Rate	48.5	47.9	42.4	36.8
Correction				
Recall			62.4	65.0
Precision			66.6	71.5
Error Rate			68.9	60.9
Perplexity	24.11	23.72	23.04	22.45

Table 4: Detecting and Correcting Speech Repairs

whether the type of repair (e.g. modification repair versus abridged repair) was properly determined. The second column gives the results of adding speech repair detection to the POS model. The third column adds in silence information. Unlike the case for boundary tones, adding silence does not have much of an effect.[4] The fourth column adds in speech repair correction, and shows that taking into account the correction, gives better detection rates (Heeman, Loken-Kim, and Allen, 1996). The fifth column adds in boundary tone detection, which improves both the detection and correction of speech repairs.

8 Comparison to Other Work

Comparing the performance of this model to others that have been proposed in the literature is very difficult, due to differences in corpora, and different input assumptions. However, it is useful to compare the different techniques that are used.

Bear *et al.* (1992) used a simple pattern matching approach on ATIS word transcriptions. They exclude all turns that have a repair that just consists of a filled pause or word fragment. On this subset they obtained a correction recall rate of 43% and a precision of 50%.

Nakatani and Hirschberg (1994) examined how speech repairs can be detected using a variety of information, including acoustic, presence of word matchings, and POS tags. Using these clues they were able to train a decision tree which achieved a recall rate of 86.1% and a precision of 92.1% on a set of turns in which each turn contained at least one speech repair.

Stolcke and Shriberg (1996b) examined whether perplexity can be improved by modeling simple types of speech repairs in a language model. They find that doing so actually makes perplexity worse, and they attribute this to not having a linguistic segmentation available, which would help in modeling filled pauses. We feel that speech repair modeling must be combined with detecting utterance boundaries and discourse markers, and should take advantage of acoustic information.

For detecting boundary tones, the model of Wightman and Ostendorf (1994) achieves a recall rate of 78.1% and a precision of 76.8%. Their better performance is partly attributed to richer (speaker dependent) acoustic modeling, including phoneme duration, energy, and pitch. However, their model was trained and tested on professionally read speech, rather than spontaneous speech.

Wang and Hirschberg (1992) did employ spontaneous speech, namely, the ATIS corpus. For turn-internal boundary tones, they achieved a recall rate of 38.5% and a precision of 72.9% using a decision tree approach that combined both textual features, such as POS tags, and syntactic constituents with intonational features. One explanation for the difference in performance was that our model was trained on approximately ten times as much data. Secondly, their decision trees are used to classify each data point independently of the next, whereas we find the best interpretation over the entire turn, and incorporate speech repairs.

The models of Kompe *et al.* (1994) and Mast *et al.* (1996) are the most similar to our model in terms of incorporating a language model. Mast *et al.* achieve a recall rate of 85.0% and a precision of 53.1% on identifying dialog acts in a German corpus. Their model employs richer acoustic modeling, however, it does not account for other aspects of utterance modeling, such as speech repairs.

9 Conclusion

In this paper, we have shown that the problems of identifying intonational boundaries and discourse markers, and resolving speech repairs can be tackled by a statistical language model, which uses local context. We have also shown that these tasks, along with POS tagging, should be resolved together. Since our model can give a probability estimate for the next word, it can be used as the lan-

[4] Silence has a bigger effect on detection and correction if boundary tones are modeled.

guage model for a speech recognizer. In terms of perplexity, our model gives a 14% improvement over word-based language models. Part of this improvement is due to being able to exploit silence durations, which traditional word-based language models tend to ignore. Our next step is to incorporate this model into a speech recognizer in order to validate that the improved perplexity does in fact lead to a better word recognition rate.

10 Acknowledgments

This material is based upon work supported by the NSF under grant IRI-9623665 and by ONR under grant N00014-95-1-1088. Final preparation of this paper was done while the first author was visiting CNET, France Télécom.

References

Allen, J. F., L. Schubert, G. Ferguson, P. Heeman, C. Hwang, T. Kato, M. Light, N. Martin, B. Miller, M. Poesio, and D. Traum. 1995. The Trains project: A case study in building a conversational planning agent. *Journal of Experimental and Theoretical AI*, 7:7–48.

Bahl, L. R., P. F. Brown, P. V. deSouza, and R. L. Mercer. 1989. A tree-based statistical language model for natural lnaguage speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(7):1001–1008.

Bear, J., J. Dowding, and E. Shriberg. 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.

Black, E., F. Jelinek, J. Lafferty, R. Mercer, and S. Roukos. 1992. Decision tree models applied to the labeling of text with parts-of-speech. In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 117–121. Morgan Kaufmann.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and Regression Trees.* Monterrey, CA: Wadsworth & Brooks.

Brown, P. F., V. J. Della Pietra, P. V. deSouza, J. C. Lai, and R. L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Byron, D. K. and P. A. Heeman. 1997. Discourse marker use in task-oriented spoken dialog. In *Proceedings of the 5th European Conference on Speech Communication and Technology (Eurospeech)*, Rhodes, Greece.

Entropic Research Laboratory, Inc., 1994. *Aligner Reference Manual.* Version 1.3.

Heeman, P. and J. Allen. 1994. Detecting and correcting speech repairs. In *Proceedings of the 32th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, Las Cruces, New Mexico, June.

Heeman, P. A. 1997. Speech repairs, intonational boundaries and discourse markers: Modeling speakers' utterances in spoken dialog. Doctoral dissertation.

Heeman, P. A. and J. F. Allen. 1995. The Trains spoken dialog corpus. CD-ROM, Linguistics Data Consortium.

Heeman, P. A. and J. F. Allen. 1997. Incorporating POS tagging into language modeling. In *Proceedings of the 5th European Conference on Speech Communication and Technology (Eurospeech)*, Rhodes, Greece.

Heeman, P. A., K. Loken-Kim, and J. F. Allen. 1996. Combining the detection and correction of speech repairs. In *Proceedings of the 4rd International Conference on Spoken Language Processing (ICSLP-96)*, pages 358–361, Philadephia, October.

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

Hirschberg, J. and D. Litman. 1993. Empirical studies on the disambiguation of cue phrases. *Computational Linguistics*, 19(3):501–530.

Katz, S. M. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, pages 400–401, March.

Kompe, R., A. Batliner, A. Kießing, U. Kilian, H. Niemann, E. Nöth, and P. Regel-Brietzmann. 1994. Automatic classification of prosodically marked phrase boundaries in german. In *Proceedings of the International Conference on Audio, Speech and Signal Processing (ICASSP)*, pages 173–176, Adelaide.

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

Magerman, D. M. 1995. Statistical decision trees for parsing. In *Proceedings of the 33th Annual Meeting of the Association for Computational Linguistics*, pages 7–14, Cambridge, MA, June.

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

Mast, M., R. Kompe, S. Harbeck, A. Kießling, H. Niemann, E. Nöth, E. G. Schukat-Talamazzini, and V. Warnke. 1996. Dialog act classification with the help of prosody. In *Proceedings of the 4rd International Conference on Spoken Language Processing (ICSLP-96)*, Philadephia, October.

Nakatani, C. H. and J. Hirschberg. 1994. A corpus-based study of repair cues in spontaneous speech. *Journal of the Acoustical Society of America*, 95(3):1603–1616.

Rosenfeld, R. 1995. The CMU statistical language modeling toolkit and its use in the 1994 ARPA CSR evaluation. In *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, San Mateo, California, 1995. Morgan Kaufmann.

Schiffrin, D. 1987. *Discourse Markers.* New York: Cambridge University Press.

Silverman, K., M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg. 1992. ToBI: A standard for labelling English prosody. In *Proceedings of the 2nd International Conference on Spoken Language Processing (ICSLP-92)*, pages 867–870.

Stolcke, A. and E. Shriberg. 1996a. Automatic linguistic segmentation of conversational speech. In *Proceedings of the 4rd International Conference on Spoken Language Processing (ICSLP-96)*, October.

Stolcke, A. and E. Shriberg. 1996b. Statistical language modeling for speech disfluencies. In *Proceedings of the International Conference on Audio, Speech and Signal Processing (ICASSP)*, May.

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

Wightman, C. W. and M. Ostendorf. 1994. Automatic labeling of prosodic patterns. *IEEE Transactions on speech and audio processing*, October.