

# Parsing Speech Repair without Specialized Grammar Symbols\*

**Tim Miller**  
University of Minnesota  
tmill@cs.umn.edu

**Luan Nguyen**  
University of Minnesota  
lnguyen@cs.umn.edu

**William Schuler**  
University of Minnesota  
schuler@cs.umn.edu

## Abstract

This paper describes a parsing model for speech with repairs that makes a clear separation between linguistically meaningful symbols in the grammar and operations specific to speech repair in the operation of the parser. This system builds a model of how unfinished constituents in speech repairs are likely to finish, and finishes them probabilistically with placeholder structure. These modified repair constituents and the restarted replacement constituent are then recognized together in the same way that two coordinated phrases of the same type are recognized.

## 1 Introduction

Speech repair is a phenomenon in spontaneous spoken language in which a speaker decides to interrupt the flow of speech, replace some of the utterance (the “reparandum”), and continues on (with the “alteration”) in a way that makes the whole sentence as transcribed grammatical only if the reparandum is ignored. As Ferreira et al. (2004) note, speech repairs<sup>1</sup> are the most disruptive type of disfluency, as they seem to require that a listener first incrementally build up syntactic and semantic structure, then subsequently remove it and rebuild when the repair is made. This difficulty combines with their frequent occurrence to make speech repair a pressing problem for machine recognition of spontaneous speech.

This paper introduces a model for dealing with one part of this problem, constructing a syntactic analysis based on a transcript of spontaneous spoken language. The model introduced here differs from other models attempting to solve the

same problem, by completely separating the fluent grammar from the operations of the parser. The grammar thus has no representation of disfluency or speech repair, such as the “EDITED” category used to represent a reparandum in the Switchboard corpus, as such categories are seemingly at odds with the typical nature of a linguistic constituent.

Rather, the approach presented here uses a grammar that explicitly represents incomplete constituents being processed, and repair is represented by rules which allow incomplete constituents to be prematurely merged with existing structure. While this model is interesting for its elegance in representation, there is also reason to hypothesize improved performance, since this processing model requires no additional grammar symbols, and only one additional operation to account for speech repair, and thus makes better use of limited data resources.

## 2 Background

Previous work on parsing of speech with repairs has shown that syntactic cues can be used to increase accuracy of detection of reparanda, which can increase overall parsing accuracy. The first source of structure used to recognize repair is what Levelt (1983) called the “Well-formedness Rule.” This rule essentially states that a speech repair acts like a conjunction; that is, the reparandum and the alteration must be of the same syntactic category. Of course, the reparandum is often unfinished, so the Well-formedness Rule allows for the reparandum category to be inferred.

This source of structure has been used by two related approaches, that of Hale et al. (2006) and Miller (2009). Hale and colleagues exploit this structure by adding contextual information to the standard reparandum label “EDITED”. In their terminology, *daughter annotation* takes the (possibly unfinished) constituent label of the reparandum and appends it to the EDITED label. This

This research was supported by NSF CAREER award 0447685. The views expressed are not necessarily endorsed by the sponsors.

<sup>1</sup>Ferreira et al. use the term ‘revisions’.

allows a learned probabilistic context-free grammar to represent the likelihood of a reparandum of a certain type being a sibling with a finished constituent of the same type.

Miller’s approach exploited the same source of structure, but changed the representation to use a REPAIRED label for alterations instead of an EDITED label for reparanda. The rationale for that change is the fact that a speech repair does not really begin until the interruption point, at which point the alteration is started and the reparandum is retroactively labelled as such. Thus, the argument goes, no special syntactic rules or symbols should be necessary until the alteration begins.

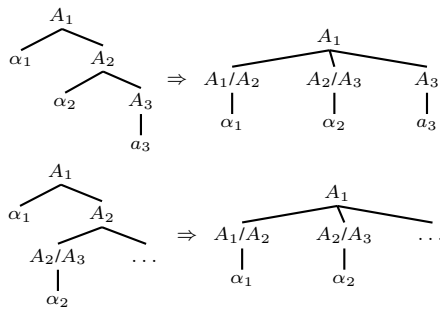
### 3 Model Description

#### 3.1 Right-corner transform

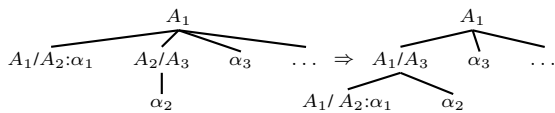
This work first uses a right-corner transform, which turns right-branching structure into left-branching structure, using category labels that use a “slash” notation  $\alpha/\gamma$  to represent an incomplete constituent of type  $\alpha$  “looking for” a constituent of type  $\gamma$  in order to complete itself.

This transform first requires that trees be binarized. This binarization is done in a similar way to Johnson (1998) and Klein and Manning (2003).

Rewrite rules for the right-corner transform are as follows, first flattening right-branching structure:<sup>2</sup>



then replacing it with left-branching structure:



One problem with this notation is the representation given to unfinished constituents, as seen in Figures 1 and 2. The standard representation of

<sup>2</sup>Here, all  $A_i$  denote nonterminal symbols, and  $\alpha_i$  denote subtrees; the notation  $A_1:\alpha_0$  indicates a subtree  $\alpha_0$  with label  $A_1$ ; and all rewrites are applied recursively, from leaves to root.

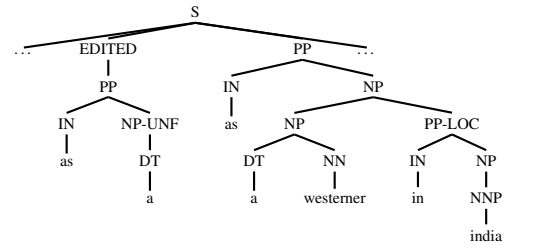


Figure 1: Section of interest of a standard phrase structure tree containing speech repair with unfinished noun phrase (NP).

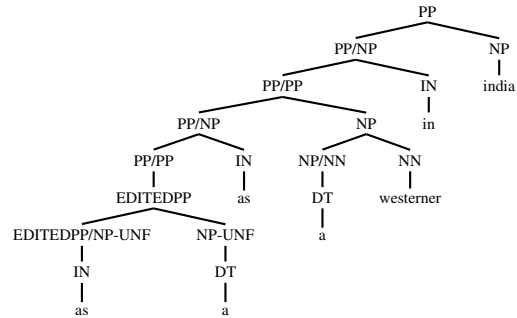


Figure 2: Right-corner transformed version of the fragment above. This tree requires several special symbols to represent the reparandum that starts this fragment.

an unfinished constituent in the Switchboard corpus is to append the -UNF label to the lowest unfinished constituent (see Figure 1). Since one goal of this work is separation of linguistic knowledge from language processing mechanisms, the -UNF tag should not be an explicit part of the grammar. In theory, the incomplete category notation induced by the right-corner transform is perfectly suited to this purpose. For instance, the category NP-UNF is a stand in category for several incomplete constituents, for example NP/NN, NP/NNS, etc. However, since the sub-trees with -UNF labels in the original corpus are by definition unfinished, the label to the right of the slash (NN in this case) is not defined. As a result, transformed trees with unfinished structure have the representation of Figure 2, which gives away the positive benefits of the right-corner transform in representing repair by propagating a special repair symbol (EDITED) through the grammar.

#### 3.2 Approximating unfinished constituents

It is possible to represent -UNF categories as standard unfinished constituents, and account for unfinished constituents by having the parser prema-

turally end the processing of a given constituent. However, in the example given above, this would require predicting ahead of time that the NP-UNF was only missing a common noun – NN (for example). This problem is addressed in this work by probabilistically filling in placeholder final categories of unfinished constituents in the standard phrase structure trees, before applying the right-corner transform.

In order to fill in the placeholder with realistic items, phrase completions are learned from corpus statistics. First, this algorithm identifies an unfinished constituent to be finished as well as its existing children (in the continuing example, NP-UNF with child labelled DT). Next, the corpus is searched for fluent subtrees with matching root labels and child labels (NP and DT), and a distribution is computed of the actual completions of those subtrees. In the model used in this work, the most common completions are NN, NNS, and NNP. The original NP-UNF subtree is then given a placeholder completion by sampling from the distribution of completions computed above.

After this addition is complete, the UNF and EDITED labels are removed from the reparandum subtree, and if a restarted constituent of the same type is a sibling of the reparandum (e.g. another NP), the two subtrees are made siblings under a new subtree with the same category label (NP). See Figure 3 for a simple visual example of how this works.

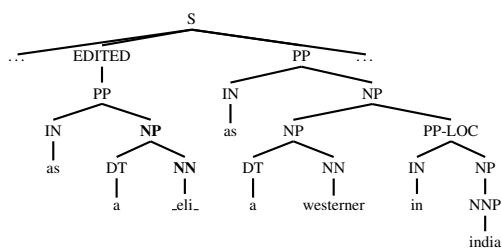


Figure 3: Same tree as in Figure 1, with the unfinished noun phrase now given a placeholder NN completion (both bolded).

Next, these trees are modified using the right-corner transform as shown in Figure 4. This tree still contains placeholder words that will not be in the text stream of an observed input sentence. Thus, in the final step of the preprocessing algorithm, the finished category label and the placeholder right child are removed where found in a right-corner tree. This results in a right-corner transformed tree in which a unary child or right

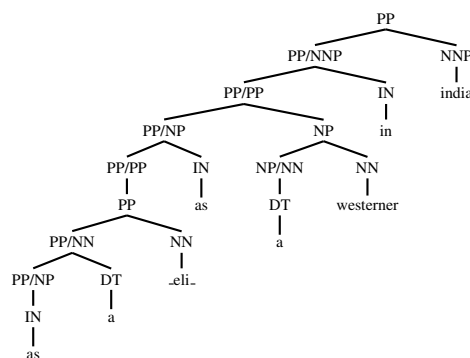


Figure 4: Right-corner transformed tree with placeholder finished phrase.

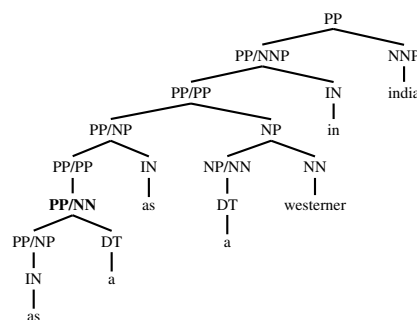


Figure 5: Final right-corner transformed state after excising placeholder completions to unfinished constituents. The bolded label indicates the signal of an unfinished category reparandum.

child subtree having an unfinished constituent type (a slash category, e.g. PP/NN in Figure 5) at its root represents a reparandum with an unfinished category. The tree then represents and processes the rest of the repair in the same way as a coordination.

## 4 Evaluation

This model was evaluated on the Switchboard corpus (Godfrey et al., 1992) of conversational telephone speech between two human interlocutors. The input to this system is the gold standard word transcriptions, segmented into individual utterances. For comparison to other similar systems, the system was given the gold standard part of speech for each input word as well. The standard train/test breakdown was used, with sections 2 and 3 used for training, and subsections 0 and 1 of section 4 used for testing. Several sentences from the end of section 4 were used during development.

For training, the data set was first standardized by removing punctuation, empty categories, typos, all categories representing repair structure,

and partial words – anything that would be difficult or impossible to obtain reliably with a speech recognizer.

The two metrics used here are the standard Parseval F-measure, and Edit-finding F. The first takes the F-score of labeled precision and recall of the non-terminals in a hypothesized tree relative to the gold standard tree. The second measure marks words in the gold standard as edited if they are dominated by a node labeled EDITED, and measures the F-score of the hypothesized edited words relative to the gold standard.

<i>System Configuration</i>	<i>Parseval-F</i>	<i>Edited-F</i>
Baseline CYK	71.05	18.03
Hale et al.	68.48	37.94
Plain RC Trees	69.07	30.89
<b>Elided RC Trees</b>	<b>67.91</b>	<b>24.80</b>
<b>Merged RC Trees</b>	<b>68.88</b>	<b>27.63</b>

Table 1: Results

Results of the testing can be seen in Table 1. The first line (“Baseline CYK”) indicates the results using a standard probabilistic CYK parser, trained on the standardized input trees. The following two lines are results from re-implementations of the systems from Hale et al. (2006) and Miller (2009). The line marked ‘Elided trees’ gives current results. Surprisingly, this result proves to be lower than the previous results. Two observations in the output of the parser on the development set gave hints as to the reasons for this performance loss.

First, repairs using the slash categories (for unfinished reparanda) were rare (relative to finished reparanda). This led to the suspicion that there was a state-splitting phenomenon, where categories previously lumped together as EDITED-NP were divided into several unfinished categories (NP/NN, NP/NNS, etc.). To test this suspicion, another experiment was performed where all unary child and right child subtrees with unfinished category labels X/Y were replaced with EDITED-X. This result is shown in line five of Table 1. This result improves on the elided version, and suggests that the state-splitting effect is most likely one cause of decreased performance.

The second effect in the parser output was the presence of several very long reparanda (more than ten words), which are highly unlikely in normal speech. This phenomenon does not occur

in the ‘Plain RC Trees’ condition. One explanation for this effect is that plain RC trees use the EDITED label in each rule of the reparandum (see Figure 2 for a short real-world example). This essentially creates a reparandum rule set, making expansion of a reparandum difficult due to the likelihood of a long chain eventually requiring a reparandum rule that was not found in the training data, or was not learned correctly in the much smaller set of reparandum-specific training data.

## 5 Conclusion and Future Work

In conclusion, this paper has presented a new model for speech containing repairs that enforces a clean separation between linguistic categories and parsing operations. Performance was below expectations, but analysis of the interesting reasons for these results suggests future directions. A model which explicitly represents the distance that a speaker backtracks when making a repair would prevent the parser from hypothesizing the unlikely reparanda of great length.

## References

- Fernanda Ferreira, Ellen F. Lau, and Karl G.D. Bailey. 2004. Disfluencies, language comprehension, and Tree Adjoining Grammars. *Cognitive Science*, 28:721–749.
- John J. Godfrey, Edward C. Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *Proc. ICASSP*, pages 517–520.
- John Hale, Izhak Shafran, Lisa Yung, Bonnie Dorr, Mary Harper, Anna Krasnyanskaya, Matthew Lease, Yang Liu, Brian Roark, Matthew Snover, and Robin Stewart. 2006. PCFGs with syntactic and prosodic indicators of speech repairs. In *Proceedings of the 45th Annual Conference of the Association for Computational Linguistics (COLING-ACL)*.
- Mark Johnson. 1998. PCFG models of linguistic tree representation. *Computational Linguistics*, 24:613–632.
- Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430.
- Willem J.M. Levelt. 1983. Monitoring and self-repair in speech. *Cognition*, 14:41–104.
- Tim Miller. 2009. Improved syntactic models for parsing speech with repairs. In *Proceedings of the North American Association for Computational Linguistics*, Boulder, CO.