A Unified Syntactic Model for Parsing Fluent and Disfluent Speech*

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

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

Abstract

This paper describes a syntactic representation for modeling speech repairs. This representation makes use of a right corner transform of syntax trees to produce a tree representation in which speech repairs require very few special syntax rules, making better use of training data. PCFGs trained on syntax trees using this model achieve high accuracy on the standard Switchboard parsing task.

1 Introduction

Speech repairs occur when a speaker makes a mistake and decides to partially retrace an utterance in order to correct it. Speech repairs are common in spontaneous speech – one study found 30% of dialogue turns contained repairs (Carletta et al., 1993) and another study found one repair every 4.8 seconds (Blackmer and Mitton, 1991). Because of the relatively high frequency of this phenomenon, spontaneous speech recognition systems will need to be able to deal with repairs to achieve high levels of accuracy.

The speech repair terminology used here follows that of Shriberg (1994). A speech repair consists of a *reparandum*, an *interruption point*, and the *alteration*. The reparandum contains the words that the speaker means to replace, including both words that are in error and words that will be retraced. The interruption point is the point in time where the stream of speech is actually stopped, and the repairing of the mistake can begin. The alteration contains the words that are meant to replace the words in the reparandum.

Recent advances in recognizing spontaneous speech with repairs (Hale et al., 2006; Johnson and Charniak, 2004) have used parsing approaches on transcribed speech to account for the structure inherent in speech repairs at the word level and above. One salient aspect of structure is the fact that there is often a good deal of overlap in words between the reparandum and the alteration, as speakers may trace back several words when restarting after an error. For instance, in the repair *. . . a flight to Boston, uh, I mean, to Denver on Friday . . .*, there is an exact match of the word 'to' between reparandum and repair, and a part of speech match between the words 'Boston' and 'Denver'.

Another sort of structure in repair is what Levelt (1983) called the well-formedness rule. This rule states that the constituent started in the reparandum and repair are ultimately of syntactic types that *could* be grammatically joined by a conjunction. For example, in the repair above, the well-formedness rule says that the repair is well formed if the fragment *. . . a flight to Boston and to Denver. . .* is grammatical. In this case the repair is well formed since the conjunction is grammatical, if not meaningful.

The approach described here makes use of a transform on a tree-annotated corpus to build a syntactic model of speech repair which takes advantage of the structure of speech repairs as described above, while also providing a representation of repair structure that more closely adheres to intuitions about what happens when speakers make repairs.

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

Proceedings of ACL-08: HLT, Short Papers (Companion Volume), pages 105–108,
Columbus, Ohio, USA, June 2008.

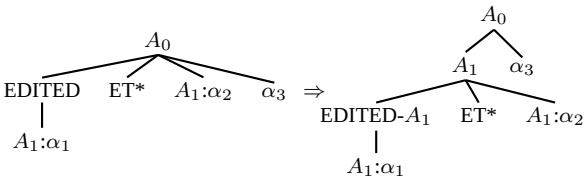
2 Speech repair representation

The representational scheme used for this work makes use of a *right-corner transform*, a way of rewriting syntax trees that turns all right recursion into left recursion, and leaves left recursion as is. As a result, constituent structure is built up during recognition in a left-to-right fashion, as words are read in. This arrangement is well-suited to recognition of speech with repairs, because it allows for constituent structure to be built up using fluent speech rules up until the moment of interruption, at which point a special repair rule may be applied. This property will be examined further in section 2.3, following a technical description of the representation scheme.

2.1 Binary branching structure

In order to obtain a linguistically plausible right-corner transform representation of incomplete constituents, the Switchboard corpus is subjected to a pre-process transform to introduce binary-branching nonterminal projections, and fold empty categories into nonterminal symbols in a manner similar to that proposed by Johnson (1998b) and Klein and Manning (2003). This binarization is done in in such a way as to preserve linguistic intuitions of head projection, so that the depth requirements of right-corner transformed trees will be reasonable approximations to the working memory requirements of a human reader or listener.

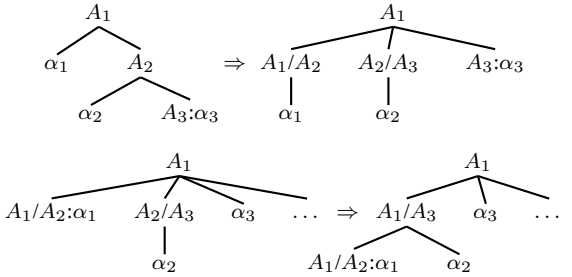
Trees containing speech repairs are reduced in arity by merging repair structure lower in the tree, when possible. As seen in the left tree below, [1] repair structure is annotated in a flat manner, which can lead to high-arity rules which are sparsely represented in the data set, and thus difficult to learn. This problem can be mitigated by using the rewrite rule shown below, which turns an EDITED-X constituent into the leftmost child of a tree of type X, as long as the original flat tree had X following an EDITED-X constituent and possibly some editing term (ET) categories. The INTJ category ('uh','um',etc.) and the PRN category ('I mean', 'that is', etc.) are considered to be editing term categories when they lie between EDITED-X and X constituents.



2.2 Right-corner transform

Binarized trees[2] are then transformed into *right-corner* trees using transform rules similar to those described by Johnson(1998a). This right-corner transform is simply the left-right dual of a left-corner transform. It transforms all right recursive sequences in each tree into left recursive sequences of symbols of the form A_1/A_2, denoting an incomplete instance of category A_1 lacking an instance of category A_2 to the right.

Rewrite rules for the right-corner transform are shown below:



Here, the first rewrite rule is applied iteratively (bottom-up on the tree) to flatten all right recursion, using incomplete constituents to record the original nonterminal ordering. The second rule is then applied to generate left recursive structure, preserving this ordering.

The incomplete constituent categories created by the right corner transform are similar in form and meaning to non-constituent categories used in Combinatorial Categorial Grammars (CCGs) (Steedman, 2000). Unlike CCGs, however, a right corner transformed grammar does not allow backward function application, composition, or raising. As a result, it does not introduce spurious ambiguity between forward and backward operations, but cannot be taken to explicitly encode argument structure, as CCGs can.

[1] Here, all A_i denote nonterminal symbols, and all α_i denote subtrees; the notation A_1:α_1 indicates a subtree α_1 with label A_1; and all rewrites are applied recursively, from leaves to root.

[2] All super-binary branches remaining after the above preprocess are 'nominally' decomposed into right-branching structures by introducing intermediate nodes with labels concatenated from the labels of its children, delimited by underscores

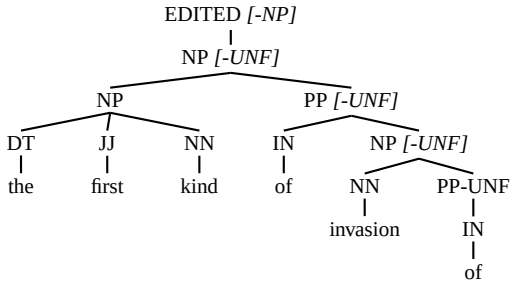


Figure 1: Standard tree repair structure, with -UNF propagation as in (Hale et al., 2006) shown in brackets.

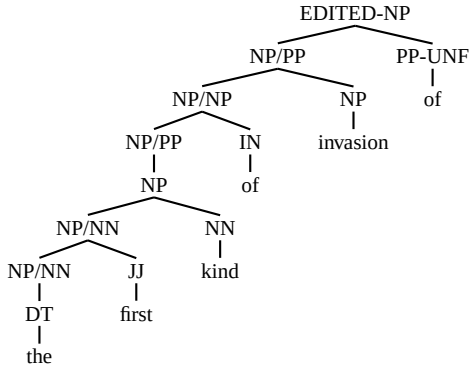


Figure 2: Right-corner transformed tree with repair structure

2.3 Application to speech repair

An example speech repair from the Switchboard corpus can be seen in Figures 1 and 2, in which the same repair fragment is shown in a standard state such as might be used to train a probabilistic context free grammar, and after the right-corner transform. Figure 1 also shows, in brackets, the augmented annotation used by Hale et al.(2006). This scheme consisted of adding -X to an EDITED label which produced a category X, as well as propagating the -UNF label at the right corner of the tree up through every parent below the EDITED root.

The standard annotation (without -UNF propagation) is deficient because even if an unfinished constituent like PP-UNF is correctly recognized, and the speaker is essentially in an error state, there may be several partially completed constituents above – in Figure 1, the NP, PP, and NP above the PP-UNF. These constituents need to be completed, but using the standard annotation there is only one chance to make use of the information about the error that has occurred – the NP → NP PP-UNF rule. Thus, by the time the error section is completed, there is no information by which a parsing algorithm could choose to reduce the topmost NP to EDITED other than independent rule probabilities.

The approach used by (Hale et al., 2006) works because the information about the transition to an error state is propagated up the tree, in the form of the -UNF tags. As the parsing chart is filled in bottom up, each rule applied is essentially coming out of a special repair rule set, and so at the top of the tree the EDITED hypothesis is much more likely. However, this requires that several fluent speech rules from the data set be modified for use in a special repair grammar, which not only reduces the amount of available training data, but violates our intuition that most reparanda are fluent up until the actual edit occurs.

The right corner transform model works in a different way, by building up constituent structure from left to right. In Figure 2, the same fragment is shown as it appears in the training data for this system. With this representation, the problem noticed by Hale and colleagues (2006) has been solved in a different way, by incrementally building up *left-branching* rather than right-branching structure, so that only a single special error rule is required at the end of the constituent. Whereas the -UNF propagation scheme often requires the entire reparandum to be generated from a speech repair rule set, this scheme only requires one special rule, where the moment of interruption actually occurred.

This is not only a pleasing parsimony, but it reduces the number of special speech repair rules that need to be learned and saves more potential examples of fluent speech rules, and therefore potentially makes better use of limited data.

3 Evaluation

The evaluation of this system was performed on the Switchboard corpus, using the *mrg* annotations in directories 2 and 3 for training, and the files sw4004.mrg to sw4153.mrg in directory 4 for evaluation, following Johnson and Charniak(2004).

The input to the system consists of the terminal symbols from the trees in the corpus section mentioned above. The terminal symbol strings are first pre-processed by stripping punctuation and other

System	Parseval F	EDIT F
Baseline	**60.86**	**42.39**
CYK (H06)	71.16	41.7
RCT	**68.36**	**64.41**
TAG-based model (JC04)	–	79.7

Table 1: Baseline results are from a standard CYK parser with binarized grammar. We were unable to find the correct configuration to match the baseline results from Hale et al. RCT results are on the right-corner transformed grammar (transformed back to flat treebank-style trees for scoring purposes). CYK and TAG lines show relevant results from related work.

non-vocalized terminal symbols, which could not be expected from the output of a speech recognizer. Crucially, any information about repair is stripped from the input, including partial words, repair symbols [3], and interruption point information. While an integrated system for processing and parsing speech may use both acoustic and syntactic information to find repairs, and thus may have access to some of this information about where interruptions occur, this experiment is intended to evaluate the use of the right corner transform and syntactic information on parsing speech repair. To make a fair comparison to the CYK baseline of (Hale et al., 2006), the recognizer was given correct part-of-speech tags as input along with words.

The results presented here use two standard metrics for assessing accuracy of transcribed speech with repairs. The first metric, Parseval F-measure, takes into account precision and recall of all non-terminal (and non pre-terminal) constituents in a hypothesized tree relative to the gold standard. The second metric, EDIT-finding F, measures precision and recall of the words tagged as EDITED in the hypothesized tree relative to those tagged EDITED in the gold standard. F score is defined as usual, $2pr/(p+r)$ for precision p and recall r.

The results in Table 1 show that this system performs comparably to the state of the art in overall parsing accuracy and reasonably well in edit detection. The TAG system (Johnson and Charniak, 2004) achieves a higher EDIT-F score, largely as a result of its explicit tracking of overlapping words between reparanda and alterations. A hybrid system using the right corner transform and keeping information about how a repair started may be able to improve EDIT-F accuracy over this system.

[3]The Switchboard corpus has special terminal symbols indicating e.g. the start and end of the reparandum.

4 Conclusion

This paper has described a novel method for parsing speech that contains speech repairs. This system achieves high accuracy in both parsing and detecting reparanda in text, by making use of transformations that create incomplete categories, which model the reparanda of speech repair well.

References

Elizabeth R. Blackmer and Janet L. Mitton. 1991. Theories of monitoring and the timing of repairs in spontaneous speech. *Cognition*, 39:173–194.

Jean Carletta, Richard Caley, and Stephen Isard. 1993. A collection of self-repairs from the map task corpus. Technical report, Human Communication Research Centre, University of Edinburgh.

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 and Eugene Charniak. 2004. A tag-based noisy channel model of speech repairs. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL '04)*, pages 33–39, Barcelona, Spain.

Mark Johnson. 1998a. Finite state approximation of constraint-based grammars using left-corner grammar transforms. In *Proceedings of COLING/ACL*, pages 619–623.

Mark Johnson. 1998b. 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.

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

Elizabeth Shriberg. 1994. *Preliminaries to a Theory of Speech Disfluencies*. Ph.D. thesis, University of California at Berkeley.

Mark Steedman. 2000. *The syntactic process*. MIT Press/Bradford Books, Cambridge, MA.