Early Deletion of Fillers In Processing Conversational Speech

Matthew Lease and Mark Johnson
Brown Laboratory for Linguistic Information Processing (BLLIP)
Brown University
Providence, RI 02912
{mlease,mj}@cs.brown.edu

Abstract

This paper evaluates the benefit of deleting fillers (e.g. *you know, like*) early in parsing conversational speech. Readability studies have shown that disfluencies (fillers and speech repairs) may be deleted from transcripts without compromising meaning (Jones et al., 2003), and deleting repairs prior to parsing has been shown to improve its accuracy (Charniak and Johnson, 2001). We explore whether this strategy of early deletion is also beneficial with regard to fillers. Reported experiments measure the effect of early deletion under in-domain and out-of-domain parser training conditions using a state-of-the-art parser (Charniak, 2000). While early deletion is found to yield only modest benefit for in-domain parsing, significant improvement is achieved for out-of-domain adaptation. This suggests a potentially broader role for disfluency modeling in adapting text-based tools for processing conversational speech.

1 Introduction

This paper evaluates the benefit of deleting fillers early in parsing conversational speech. We follow LDC (2004) conventions in using the term *filler* to encompass a broad set of vocalized space-fillers that can introduce syntactic (and semantic) ambiguity. For example, in the questions

```
Did you know I do that?
Is it like that one?
```

colloquial use of fillers, indicated below through use of commas, can yield alternative readings

```
Did, you know, I do that?
Is it, like, that one?
```

Readings of the first example differ in querying listener knowledge versus speaker action, while readings of the second differ in querying similarity versus exact match. Though an engaged listener rarely has difficulty distinguishing between such alternatives, studies show that deleting disfluencies from transcripts improves readability with no reduction in reading comprehension (Jones et al., 2003).

The fact that disfluencies can be completely removed without compromising meaning is important. Earlier work had already made this claim regarding speech repairs[1] and argued that there was consequently little value in syntactically analyzing repairs or evaluating our ability to do so (Charniak and Johnson, 2001). Moreover, this work showed that collateral damage to parse accuracy caused by repairs could be averted by deleting them prior to parsing, and this finding has been confirmed in subsequent studies (Kahn et al., 2005; Harper et al., 2005). But whereas speech repairs have received significant attention in the parsing literature, fillers have been relatively neglected. While one study has shown that the presence of interjection and parenthetical constituents in conversational speech reduces parse accuracy (Engel et al., 2002), these constituent types are defined to cover both fluent and disfluent speech phenomena (Taylor, 1996), leaving the impact of fillers alone unclear.

In our study, disfluency annotations (Taylor, 1995) are leveraged to identify fillers precisely, and these annotations are merged with treebank syntax. Extending the arguments of Charniak and Johnson with regard to repairs (2001), we argue there is little value in recovering the syntactic structure

[1] See (Core and Schubert, 1999) for a prototypical counter-example that rarely occurs in practice.

of fillers, and we relax evaluation metrics accordingly (§3.2). Experiments performed (§3.3) use a state-of-the-art parser (Charniak, 2000) to study the impact of early filler deletion under in-domain and out-of-domain (i.e. adaptation) training conditions. In terms of adaptation, there is tremendous potential in applying textual tools and training data to processing transcribed speech (e.g. machine translation, information extraction, etc.), and *bleaching* speech data to more closely resemble text has been shown to improve accuracy with some text-based processing tasks (Rosenfeld et al., 1995). For our study, a state-of-the-art filler detector (Johnson et al., 2004) is employed to delete fillers prior to parsing. Results show parse accuracy improves significantly, suggesting disfluency filtering may have a broad role in enabling text-based processing of speech data.

2 Disfluency in Brief

In this section we give a brief introduction to disfluency, providing an excerpt from Switchboard (Graff and Bird, 2000) that demonstrates typical production of repairs and fillers in conversational speech.

We follow previous work (Shriberg, 1994) in describing a repair in terms of three parts: the *reparandum* (the material repaired), the corrected *alteration*, and between these an optional *interregnum* (or editing term) consisting of one or more fillers. Our notion of fillers encompasses filled pauses (e.g. `uh, um, ah`) as well as other vocalized space-fillers annotated by LDC (Taylor, 1995), such as `you know, i mean, like, so, well`, etc. Annotations shown here are typeset with the following conventions: **fillers** are bold, [reparanda] are square-bracketed, and <u>alterations</u> are underlined.

> S1: **Uh** first **um** i need to know **uh** how do you feel [about] **uh** <u>about</u> sending **uh** an elderly **uh** family member to a nursing home
>
> S2: **Well** of course [it's] **you know** <u>it's</u> one of the last few things in the world you'd ever want to do **you know** unless it's just **you know** really **you know uh** [for their] **uh you know** <u>for their</u> own good

Though disfluencies rarely complicate understanding for an engaged listener, deleting them from transcripts improves readability with no reduction in reading comprehension (Jones et al., 2003). For automated analysis of speech data, this means we may freely explore processing alternatives which delete disfluencies without compromising meaning.

3 Experiments

This section reports parsing experiments studying the effect of early deletion under in-domain and out-of-domain parser training conditions using the August 2005 release of the Charniak parser (2000). We describe data and evaluation metrics used, then proceed to describe the experiments.

3.1 Data

Conversational speech data was drawn from the Switchboard corpus (Graff and Bird, 2000), which annotates disfluency (Taylor, 1995) as well as syntax. Our division of the corpus follows that used in (Charniak and Johnson, 2001). Speech recognizer (ASR) output is approximated by removing punctuation, partial words, and capitalization, but we do use reference words, representing an upperbound condition of perfect ASR. Likewise, annotated sentence boundaries are taken to represent oracle boundary detection. Because fillers are annotated only in disfluency markup, we perform an automatic tree transform to merge these two levels of annotation: each span of contiguous filler words were pruned from their corresponding tree and then reinserted at the same position under a flat FILLER constituent, attached as highly as possible. Transforms were achieved using TSurgeon[2] and Lingua::Treebank[3].

For our out-of-domain training condition, the parser was trained on sections 2-21 of the Wall Street Journal (WSJ) corpus (Marcus et al., 1993). Punctuation and capitalization were removed to bleach our our textual training data to more closely resemble speech (Rosenfeld et al., 1995). We also tried automatically changing numbers, symbols, and abbreviations in the training text to match how they would be read (Roark, 2002), but this did not improve accuracy and so is not discussed further.

3.2 Evaluation Metrics

As discussed earlier (§1), Charniak and Johnson (2001) have argued that speech repairs do not

[2] http://nlp.stanford.edu/software/tsurgeon.shtml

[3] http://www.cpan.org

contribute to meaning and so there is little value in syntactically analyzing repairs or evaluating our ability to do so. Consequently, they *relaxed* standard PARSEVAL (Black et al., 1991) to treat EDITED constituents like punctuation: adjacent EDITED constituents are merged, and the internal structure and attachment of EDITED constituents is not evaluated. We propose generalizing this approach to disfluency at large, i.e. fillers as well as repairs. Note that the details of appropriate evaluation metrics for parsed speech data is orthogonal to the parsing methods proposed here: however parsing is performed, we should avoid wasting metric attention evaluating syntax of words that do not contribute toward meaning and instead evaluate only how well such words can be identified.

Relaxed metric treatment of disfluency was achieved via simple parameterization of the SParseval tool (Harper et al., 2005). SParseval also has the added benefit of calculating a dependency-based evaluation alongside PARSEVAL's bracket-based measure. The dependency metric performs syntactic head-matching for each word using a set of given head percolation rules (derived from Charniak's parser (2000)), and its relaxed formulation ignores terminals spanned by FILLER and EDITED constituents. We found this metric offered additional insights in analyzing some of our results.

3.3 Results

In the first set of experiments, we train the parser on Switchboard and contrast early deletion of disfluencies (identified by an oracle) versus parsing in the more usual fashion. Our method for early deletion generalizes the approach used with repairs in (Charniak and Johnson, 2001): contiguous filler and edit words are deleted from the input strings, the strings are parsed, and the removed words are reinserted into the output trees under the appropriate flat constituent, FILLER or EDITED.

Results in Table 1 give F-scores for PARSEVAL and dependency-based parse accuracy (§3.2), as well as per-word edit and filler detection accuracy (i.e. how well the parser does in identifying which terminals should be spanned by EDITED and FILLER constituents when early deletion is not performed). We see that the parser correctly identifies filler words with 93.1% f-score, and that early deletion of fillers (via oracle knowledge) yields only a modest improvement in parsing accuracy (87.8% to 88.9% bracket-based, 87.9% to 88.5% dependency-based). We conclude from this that for in-domain training, early deletion of fillers has limited potential to improve parsing accuracy relative to what has been seen with repairs. It is still worth noting, however, that the parser does perform better when fillers are absent, consistent with Engel et al.'s findings (2002). While fillers have been reported to often occur at major clause boundaries (Shriberg, 1994), suggesting their presence may benefit parsing, we do not find this to be the case. Results shown for repair detection accuracy and its impact on parsing are consistent with previous work (Charniak and Johnson, 2001; Kahn et al., 2005; Harper et al., 2005).

Table 1: F-scores on Switchboard when trained in-domain. LB and Dep refer to relaxed labelled-bracket and dependency parse metrics (§3.2). Edit and filler word detection f-scores are also shown.

Edits	Fillers	Edit F	Filler F	LB	Dep
oracle	oracle	100.0	100.0	88.9	88.5
oracle	parser	100.0	93.1	87.8	87.9
parser	oracle	64.3	100.0	85.0	85.6
parser	parser	62.4	94.1	83.9	85.0

Our second set of experiments reports the effect of deleting fillers early when the parser is trained on text only (WSJ, §3.1). Our motivation here is to see if disfluency modeling, particularly filler detection, can help bleach speech data to more closely resemble text, thereby improving our ability to process it using text-based methods and training data (Rosenfeld et al., 1995). Again we contrast standard parsing with deleting disfluencies early (via oracle knowledge). Given our particular interest in fillers, we also report the effect of detecting them via a state-of-the-art system (Johnson et al., 2004).

Results appear in Table 2. It is worth noting that since our text-trained parser never produces FILLER or EDITED constituents, the bracket-based metric penalizes it for each such constituent appearing in the gold trees. Similarly, since the dependency metric ignores terminals occurring under these constituents in the gold trees, the metric penalizes the parser for producing dependencies for these termi-

Table 2: F-scores parsing Switchboard when trained on WSJ. Edit word detection varies between parser and oracle, and filler word detection varies between none, system (Johnson et al., 2004), and oracle. Filler F, LB, and Dep are defined as in Table 1.

Edits	Fillers	Filler F	LB	Dep
oracle	oracle	100.0	83.6	81.4
oracle	detect	89.3	81.6	80.5
oracle	none	-	71.8	75.4
none	oracle	100.0	76.3	76.7
none	detect	74.6	75.9	91.3
none	none	-	66.8	71.5

nals. Taken together, the two metrics provide a complementary perspective in interpreting results.

The trend observed across metrics and edit detection conditions shows that early deletion of system-detected fillers improves parsing accuracy 5-10%. As seen with in-domain training, early deletion of repairs is again seen to have a significant effect. Given that state-of-the-art edit detection performs at about 80% f-measure (Johnson and Charniak, 2004), much of the benefit derived here from oracle repair detection should be realizable in practice. The broader conclusion we draw from these results is that disfluency modeling has significant potential to improve text-based processing of speech data.

4 Conclusion

While early deletion of fillers has limited benefit for in-domain parsing of speech data, it can play an important role in *bleaching* speech data for more accurate text-based processing. Alternative methods of integrating detected filler information, such as parse reranking (Kahn et al., 2005), also merit investigation. It will also be important to evaluate the interaction with ASR error and sentence boundary detection error. In terms of bleaching, we saw that even with oracle detection of disfluency, our text-trained model still significantly under-performed the in-domain model, indicating additional methods for bleaching are still needed. We also plan to evaluating the benefit of disfluency modeling in bleaching speech data for text-based machine translation.

Acknowledgments

This work was supported by NSF grants 0121285, LIS9720368, and IIS0095940, and DARPA GALE contract HR0011-06-2-0001. We would like to thank Brian Roark, Mary Harper, and the rest of the JHU PASSED team for its support of this work.

References

E. Charniak and M. Johnson. 2001. Edit detection and parsing for transcribed speech. In *Proc. NAACL*, pages 118–126.

E. Charniak. 2000. A maximum-entropy-inspired parser. In *Proc. NAACL*, pages 132–139.

M.G. Core and L.K. Schubert. 1999. A syntactic framework for speech repairs and other disruptions. In *Proc. ACL*, pages 413–420.

E. Black et al. 1991. Procedure for quantitatively comparing the syntactic coverage of English grammars. In *Proc. Workshop on Speech and Natural Language*, pages 306–311.

D. Engel, E. Charniak, and M. Johnson. 2002. Parsing and disfluency placement. In *Proc. EMNLP*, pages 49–54.

D. Graff and S. Bird. 2000. Many uses, many annotations for large speech corpora: Switchboard and TDT as case studies. In *Proc. LREC*, pages 427–433.

M. Harper et al. *2005 Johns Hopkins Summer Workshop Final Report on Parsing and Spoken Structural Event Detection*.

J.G. Kahn et al. 2005. Effective use of prosody in parsing conversational speech. In *Proc. HLT/EMNLP*, 233–240.

M. Johnson and E. Charniak. 2004. A TAG-based noisy channel model of speech repairs. In *Proc. ACL*, pages 33–39.

M. Johnson, E. Charniak, and M. Lease. 2004. An improved model for recognizing disfluencies in conversational speech. In *Proc. Rich Text 2004 Fall Workshop (RT-04F)*.

D. Jones et al. 2003. Measuring the readability of automatic speech-to-text transcripts. In *Proc. Eurospeech*, 1585–1588.

Linguistic Data Consortium (LDC). 2004. Simple metadata annotation specification version 6.2.

M. Marcus et al. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2): 313–330.

B. Roark. 2002. Markov parsing: Lattice rescoring with a statistical parser. In *Proc. ACL*, pages 287–294.

R. Rosenfeld et al. 1995. Error analysis and disfluency modeling in the Swichboard domain: 1995 JHU Summer Workshop project team report.

E. Shriberg. 1994. *Preliminaries to a Theory of Speech Disfluencies*. Ph.D. thesis, UC Berkeley.

A. Taylor, 1995. *Revision of Meteer et al.'s Dysfluency Annotation Stylebook for the Switchboard Corpus*. LDC.

A. Taylor, 1996. *Bracketing Switchboard: An addendum to the Treebank II Bracketing Guidelines*. LDC.