PARSE STRUCTURE AND SEGMENTATION FOR IMPROVING SPEECH RECOGNITION

William P. McNeill†, Jeremy G. Kahn†, Dustin L. Hillard‡, Mari Ostendorf†‡

University of Washington
†Department of Linguistics, ‡Department of Electrical Engineering
{billmcn,jgk,hillard,mo}@ssli.ee.washington.edu

ABSTRACT

Separate avenues of prior work have shown that parsing language models lead to improved recognition performance, and that segmentation of speech into sentence-like units has an impact on parser performance. This paper brings these two findings together, showing that segmentation also impacts the quality of a syntax-based language model, such that larger reductions in word error rate are possible when using sentence-like segmentations rather than simple paused-based strategies. Further, we show that the same types of syntactic features used in parse reranking can also be used to reduce word error rate in an N-best rescoring framework.

Index Terms— natural languages, speech recognition

1. INTRODUCTION

With their ability to model high-level syntactic structure and long-distance dependencies, parser-based language models can provide a complement to n-grams in speech recognition applications. Experiments evaluating PCFG-based parsing language models in terms of both perplexity and word error rate (WER) have shown performance gains, particularly when used in conjunction with n-grams [1, 2, 3]. Parsers work on entire sentences, however, and sentence boundaries are not always known (or well defined) for many speech tasks, including both broadcast news and conversational speech. Yet most of the experimental work with parsing language models for speech recognition have been based on known sentence boundaries, as in the read Wall Street Journal corpus. For corpora where sentence boundaries are not known, recognizers typically use pause-based segmentations. However, training and testing language models in mismatched segmentation conditions can lead to degraded performance, even for n-gram language models, as shown in [4].

The authors wish to thank Y. Liu and ICSI for use of the sentence segmentation system, and M. Johnson and E. Charniak for the use of their N-best parsing, parse-feature-extraction and reranking software This work was supported in part by NSF grant IIS-0326276, DARPA grant MDA972-02-C-0038 and Bosch. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

Sentence segmentation has been shown to have a dramatic impact on parser performance on conversational speech, both for the case when the human-transcribed words are given [5] (as judged by the standard Parseval measure [6]) and on speech recognition transcripts [7] (evaluating with the more recent SParseval measure [8]). Both studies show that using explicit sentence boundary detection, even though not highly accurate, yields much higher F-scores than when using pause-based segmentation. These results raise the question of how effective parsing language models are for speech recognition with pause-based vs. automatic sentence segmentation.

In addition to exploring the issue of segmentation, this paper also looks at new ways in which parsing language models can be used to improve word error rate in speech recognition. One parsing development that has not yet been leveraged in speech recognition is the use of discriminative methods that rerank parse hypotheses [9, 10] with models of their performance on an external metric. These reranking systems generate multiple candidates from a PCFG, extract syntactic feature vectors of the parse structure for each candidate, and incorporate a second-stage model of the relationship between these vectors and an external evaluation measure of parse quality (usually Parseval).

We have built a system that allows us to vary the segmentation of the output of a speech recognizer and then rescore the resulting N-best lists using a variety of features from a parsing-based language model. This system allows us to investigate the effectiveness of parser-based language models for speech recognition and their interaction with segmentation. In particular, the key questions investigated in this research are:

- Which combinations of parse scores and/or features are useful for improving recognition performance?
- How much does utterance segmentation impact the usefulness of a parsing language model in speech recognition?

In the sections to follow, we describe the system architecture for investigating these questions, together with details of the segmentation system and of the parse features, and present experimental result in conversational speech recognition.

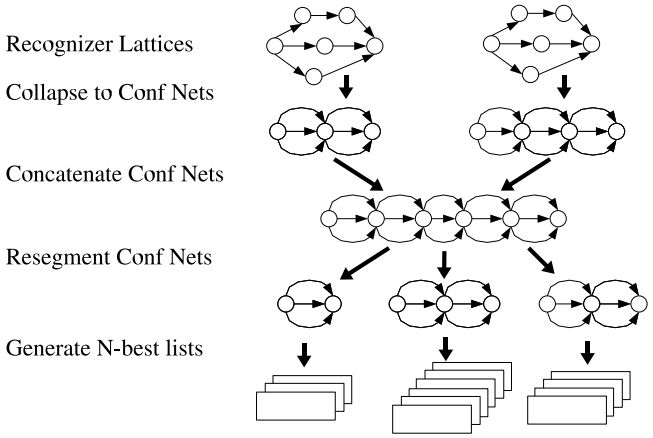

Fig. 1. N-best resegmentation using confusion networks.

2. METHODS

2.1. System Architecture

Our system includes the following steps:

1. Speech recognition using a pause-based segmentation and producing multiple hypotheses as output;
2. Automatic sentence boundary detection based on the 1-best recognition hypothesis;
3. Resegmentation of the recognizer output and generation of N-best hypotheses based on the new sentence segmentations;
4. Generation of the M-best parses for each sentence (word sequence) hypothesis, followed by extraction of parse features for each word-parse hypothesis; and
5. Reranking of the word sequence hypotheses based on ASR and parse scores, and optionally parse features.

For the recognition step (1), we use the SRI $5\times$RT Decipher speech recognition system [11], a state-of-the-art large vocabulary system. For the boundary detection step (2), we use the system in [12] which combines a hidden-event language model and bagging decision trees for modeling prosodic cues. The resegmentation and parse feature extraction steps (3) and (4) are key contributions of this work and are described further in the sections to follow. The reranker in step (5) is an average perceptron, a discriminative learner that works with a high-dimensional vector of possibly statistically correlated features. The top-ranking word-parse hypothesis returned by the reranker is evaluated in terms of word error rate.

2.2. Segmentation

Our experiments use three different segmentation conditions: the pause-based segmentation returned by the recognizer, automatic segmentation, and the hand-annotated segmentations. For all but the first case, the resegmentation strategy depicted in Figure (1) is used to produce multiple recognition hypotheses associated with the new segmentations without rerunning the recognition process on the new waveform segmentation. The recognition system output, which may be an N-best list or lattice, is converted into a word-level confusion network [13], which is comprised of a series of word slots, each of which has a list of alternative word hypotheses with associated posterior probabilities. The confusion networks are then concatenated to form a single network per speaker (conversation side), which is re-cut at the new segmentation points. In order to find the cut points for the reference segmentation, the confusion networks for a given speaker are aligned with the reference transcript using dynamic programming.

After resegmenting, the confusion networks are decoded to create N-best lists of word sequence hypotheses for each of the new segments. Each hypothesis w_i has an associated score, $\log p_{recog}(w_i)$, which is the sum of the log posteriors of the individual words in that hypothesis. The word posteriors are estimated from forward-backward lattice acoustic and n-gram language model scores as part of the confusion network generation process. For each w_i we also extract a word count C_i to enable the reranker to automatically incorporate a word insertion penalty.

2.3. Parsing and Parse Features

In step (4), we use a PCFG-based parser to parse the sentence hypotheses in the resegmented N-best lists. For the i-th word sequence hypothesis w_i in the N-best list for a given utterance, we produce up to M parses t_{ij} with associated parse probabilities $p(t_{ij}, w_i)$. From each parse we extract a vector of syntactic features $\vec{f_{ij}}$ that characterize various aspects of its structure. For example, one element of this vector might count the number of VPs in that parse of length 5 and headed by the word "think". Including only those features that separate individual hypotheses in more than 2000 examples in the training set, the resulting feature set is $\approx$ 130k features. For the PCFG-based parser, feature extractor, and the particular set of syntactic features that comprise $\vec{f_{ij}}$, we adapt the components used in [9]. For the sake of computational tractability, we restricted both M and N to be no larger than 20.

From the basic PCFG scores we derive several parse scores and features for use in reranking. For a given sentence hypothesis, the parse language model probability is the sum of all the individual parse probabilities:

$$p_{parser}(w_i) = \sum_{k=1}^{M} p(t_{ik}, w_i) \quad (1)$$

Two scores specific to the parse and word sequence are used: the conditional probability of the parse given the words

$$p(t_{ij}|w_i) = \frac{p(t_{ij}, w_i)}{p_{parser}(w_i)}, \quad (2)$$

and a normalized version for which the most probable parse gets a score of 1

$$p_0(t_{ij}|w_i) = \frac{p(t_{ij}|w_i)}{\max_{1 \le k \le M} p(t_{ij}|w_i)}, \quad (3)$$

which allows the use of individual parse probabilities without penalizing sentence hypotheses for which there may be many parses.

We also include the expectation of the $\vec{f_{ij}}$ values over the conditional parse probabilities, $p(t_{ij}|w_i)$, producing a single expected feature vector

$$E[\vec{F_i}] = \sum_{j=1}^{M} p(t_{ij}|w_i)\vec{f_{ij}} \quad (4)$$

for each word sequence hypothesis.

For utterances with some empty word sequence hypotheses (i.e. the recognizer output is noise, silence, laughter or other non-word events), the empty hypotheses are assumed to have a single parse with a very small probability $p(t_{i1}, w_i) = \epsilon$ and zero-count entries in the feature vector $\vec{f_{i1}}$. A boolean empty hypothesis flag B_i is included as a separate feature which effectively enables the reranker to learn an empty hypothesis penalty.

3. EXPERIMENTS

3.1. Corpus

For this experiment we used the Switchboard corpus of conversational speech [14], specifically the subset with Penn Treebank parses [15]. We resegmented the reference parses according to the later hand-annotation of sentence-like units [16]. We partitioned the data into a training set consisting of 1044 conversation sides and a test set consisting of 128 conversation sides. We also held out a development set consisting of 116 conversation sides for use in system development and debugging. For these experiments we used the original LDC transcriptions of the Switchboard audio rather than the more recent Mississippi State transcriptions [17] because we only have reference parses for the former.

Limited amounts of treebanked training data for the parser component constrained us to generate representative re-ranker training data by dividing the training set into ten parts and generating candidate parses for each part using a parser model trained on the remainder, as described in [10].

3.2. Reranker Feature Sets

We investigated the effect of different information sources on the effectiveness of parsing language models by varying the vector of features passed to the reranker. All the reranker vectors include the ASR probability $p_{recog}(w_i)$, word count C_i, and the empty hypothesis flag B_i. The $p_{recog}(w_i)$ score is the product of word posteriors from the confusion network, which incorporate both acoustic model and n-gram language model scores. Hence, the experiments here address how much a parser adds on top of n-grams. The different reranker feature sets contain the following additional features: **Parse** contains $p(t_{ij}|w_i)$ and $p_0(t_{ij}|w_i)$, **ParseLM** contains $p_{parser}(w_i)$, **Feat** contains $\vec{f_{ij}}$, and **E[Feat]** contains $E[\vec{F_i}]$. All probability scores are in log space.

Hand Segmentation

Features	WER
Baseline	22.9
Parse	22.8
Parse+ParseLM	**22.5**
ParseLM	22.4
ParseLM+Feat	**21.6**
ParseLM+E[Feat]	21.5

Table 1. Word error rate for different parse score combinations using hand-marked sentence segmentations.

3.3. Results

We reranked the word-parse hypotheses for all the resegmented N-best lists in each segmentation/feature-set pair. Because of memory limitations, we partitioned the training data and trained 15 separate models whose rank outputs were averaged together. All the top-ranked sentence hypotheses for a given speaker were concatenated together and scored against reference transcripts using the NIST `sclite` tool [18] to generate word error rates.

To address our first question from Section 2, we tried different combinations of information sources while holding the segmentation constant. Table 1 shows WER results for the hand-parsed reference segmentation. The trends are similar for the other segmentations. The best WER is obtained by using ParseLM+E[Feat], and aggregate features (ParseLM and E[Feat]) tend to be more effective than the corresponding individual values (Parse and Feat, respectively). WER values in bold are significantly different from the ones above them at a level of $p < 0.001$ for all NIST WER tests. All other adjacent WER differences are not statistically significant.

To address our second question, we applied the aggregate features across all three sentence segmentations. The results are shown in Table 2. Along the segmentation dimension, the pause-based segmentation serves as a baseline, while the hand segmentations correspond to the oracle case. All feature combinations perform better on the hand- than the pause-based segmentation. The performance on the pause-based and automatic segmentations are the same, except in the oracle case. Corresponding pause-based and automatic WER values are statistically indistinguishable. Corresponding pause/hand and automatic/hand WER values are all statistically different at $p < 0.001$, except ParseLM pause-based and hand which are

	Segmentation		
Features	Pause	Auto	Hand
ParseLM	22.6	22.7	22.4
ParseLM+E[Feat]	22.4	22.3	21.5
Oracle	17.5	16.6	16.0

Table 2. Word error rate for different sentence segmentations, compared to a baseline of 22.9% with no parsing.

different at $p < 0.05$.

3.4. Analysis

Approximately 20% of the segments in a given condition have WER changes with reranking, but generally only a few words change. Inspection of a number of the changed segments shows a few interesting patterns of error corrections, including correction of the determiner in a noun phrase, such as a deleted "the", and correction of headwords of major structural components, as in the example below, where the first line is the (correct) reranking output and the second is the n-gram-only result.

> so it's uh wound up that uh **we're** the old folks now
> so it's uh wound up that uh **where** the old folks now

In other examples the syntactic features help in overcoming the tendency to recognize frequent conversational words such as "yeah". We hypothesized that the parsing language model would have greater impact when the segment boundaries were correct. In looking at automatically-detected segments where parsing information selected an alternate word hypothesis, we found that 63% of segments with correctly-predicted boundaries improved vs. 56% for the incorrect-boundary case — a smaller difference than expected.

4. CONCLUSION

Our experiment with different information sources in ASR indicates that there is information to be obtained from the parse beyond the PCFG score, showing that parse features that have proved helpful for improving parse quality are also helpful for reducing WER. Future work includes determining which components of $\vec{f_{ij}}$ are best suited for the ASR task.

Our work also demonstrates that obtaining the correct segmentation can help improve WER. The oracle WER goes down as the segmentation condition improves. We suspect this improvement is because the better segmentation conditions generally have more (shorter) segments, and therefore allow for richer N-best lists for a fixed N. The parsing information sources are also more effective for reducing WER on the hand-parsed segmentation than the automatic segmentation. Our current automatic segmentation, however, does not do better than that returned by the recognizer. Another possibility for further investigation would be to tune the automatic segmentation operating point for parsing, as in [7]. Additionally, it may be that the segmentation-parsing interaction is of more importance in other speech domains, such as broadcast news.

5. REFERENCES

[1] B. Roark, "Probabilistic top-down parsing and language modeling," *Comp. Linguistics*, vol. 27, no. 2, pp. 249–276, 2001.

[2] E. Charniak, "Immediate-head parsing for language models," in *Proc. ACL*, 2001, pp. 124–131.

[3] C. Chelba and F. Jelinek, "Structured language modeling," *Computer Speech and Language*, vol. 14, no. 4, pp. 283–332, 2000.

[4] A. Stolcke, "Modeling linguistic segment and turn-boundaries for n-best rescoring of spontaneous speech," in *Proc. Eurospeech*, 1997, vol. 5, pp. 2779–2782.

[5] J. G. Kahn, "Moving beyond the lexical layer in parsing conversational speech," M.A. thesis, Univ. of Washington, 2005.

[6] E. Black *et al.*, "A procedure for quantitatively comparing syntactic coverage of English grammars," in *Proc. DARPA Speech & Natural Language Workshop*, 1991, pp. 306–311.

[7] M. Harper *et al.*, "Parsing speech and structural event detection," JHU Summer Workshop Final Report, 2005.

[8] B. Roark *et al.*, "SParseval: Evaluation metrics for parsing speech," in *Proc. LREC*, 2006.

[9] E. Charniak and M. Johnson, "Coarse-to-fi ne n-best parsing and MaxEnt discriminative reranking," in *Proc. ACL*, 2005, pp. 173–180.

[10] M. Collins and T. Koo, "Discriminative reranking for natural language parsing," *Comp. Linguistics*, vol. 31, no. 1, pp. 25–69, 2005.

[11] A. Stolcke *et al.*, "Recent innovations in speech-to-text transcript at SRI-ICSI-UW," *IEEE Trans. Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1729–1744, 2006.

[12] Y. Liu *et al.*, "Enriching speech recognition with sentence boundaries and disfluencies," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1526–1540, 2006.

[13] L. Mangu *et al.*, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech and Language*, vol. 14, no. 4, pp. 373–400, 2000.

[14] J. J. Godfrey *et al.*, "Switchboard: Telephone speech corpus for research and development," in *Proc. ICASSP*, 1992, vol. I, pp. 517–520.

[15] M. P. Marcus *et al.*, "Building a large annotated corpus of English: the Penn treebank," *Comp. Linguistics*, vol. 19, no. 1, 1993.

[16] M. Meteer *et al.*, "Dysfluency annotation stylebook for the switchboard corpus," Tech. Rep., Linguistic Data Consortium (LDC), 1995.

[17] ISIP, "Mississippi State transcriptions of SWITCHBOARD," http://www.isip.msstate.edu/projects/switchboard/, 1997.

[18] NIST, "NIST speech recognition scoring toolkit (SCTK)," Tech. Rep., NIST, http://www.nist.gov/speech/tools/, 2005.