USE OF PROSODY IN SYNTACTIC DISAMBIGUATION: AN ANALYSIS-BY-SYNTHESIS APPROACH

C. W. Wightman, N. M. Veilleux, M. Ostendorf

Boston University
44 Cummington St.
Boston, MA 02215

ABSTRACT

Experiments have shown that prosody is used by human listeners to disambiguate spoken language and, in particular, that the relative size and location of prosodic phrase boundaries provides a cue for resolving syntactic ambiguity. Therefore, automatically detected prosodic phrase boundaries can provide information useful in speech understanding for choosing among several candidate parses. Here, we propose a scoring algorithm to rank candidate parses based on an analysis-by-synthesis method which compares the observed prosodic phrase structure with the predicted structure for each candidate parse. In experiments with a small corpus of ambiguous sentences spoken by FM radio announcers, we have achieved disambiguation performance close to the performance of human subjects in perceptual experiments.

INTRODUCTION

Spoken language processing is a difficult problem, in part because of the many ambiguities inherent in natural language. Syntactic ambiguity arises when a given expression can be described by more than one syntactic structure, and represents an important problem in natural language processing. In particular, attachment ambiguities occur frequently in language, e.g.,

"Show the fares for [the cheapest flights on the screen]."
"Show [the fares for the cheapest flights] on the screen."

Several factors may be involved in resolving such ambiguities, including semantics, discourse and syntactic bias. In spoken language, prosody, or the suprasegmental information in an utterance, is an important additional cue.

Experimental evidence has shown that listeners can resolve several types of syntactic ambiguities by using prosodic information [13,8]. In [13], ambiguous sentences were read in contexts in which only one interpretation was reasonable, and the recordings edited to remove the context. Human subjects then listened to the ambiguous sentences, and were asked to select the intended meaning, which they were able to do reliably (86% correct identification) for six out of seven types of structural ambiguity. In addition, it appeared that the relative size and location of prosodic phrase boundaries provided the principal prosodic clue for resolving ambiguities. Thus, it seems likely that automatically detected prosodic phrase breaks could be used by speech understanding systems to reduce syntactic ambiguity.

In previous work [10], a hierarchical set of break indices was proposed as a representation of prosodic phrase boundaries and automatically detected break indices were used in a parser to provide constraints on rules that would prevent prosodically inconsistent parses. Here, we propose a scoring algorithm to rank candidate parses based on an analysis-by-synthesis method which involves: (1) using an algorithm to predict prosodic break locations for each candidate syntactic structure; (2) automatically detecting prosodic breaks in the spoken utterance; and (3) ranking the parses according to a similarity score between predicted and observed prosodic structure. Using the database of ambiguous sentences from [13], this approach achieves performance close to that of the human subjects.

The following section describes the speech corpus and prosodic break index representation. We then examine synthesis algorithms for predicting the phrase structure of a sentence and evaluate them in relation to the hand-labeled speech corpus. Next, we describe an automatic method of labeling the prosodic phrase structure and a measure of the similarity between the predicted and detected prosodic structures. We then present experimental results, based on the ambiguous sentence corpus, demonstrating the utility of this approach. In conclusion, we discuss future work suggested by these results.

CORPUS AND LABELING

As mentioned above, the experiments here are based on the corpus of ambiguous sentences described in [13]. An advantage of using this database is in the availability of perceptual experiment results, which provide an interesting performance baseline for comparison with results from our algorithm. The corpus and associated prosodic labeling is

described briefly here; readers are referred to [13] for further details.

Four professional FM radio announcers were asked to read 35 pairs of sentences, where members of a pair were phonetically similar but associated with different syntactic structures and therefore different meanings. The sentences included five examples of each of seven types of structural ambiguity: (1) parenthetical clauses vs. non-parenthetical subordinate clauses, (2) appositions vs. attached noun (or prepositional) phrases, (3) main clauses linked by coordinating conjunctions vs. a main clause and a subordinate clause, (4) tag questions vs. attached noun phrases, (5) far vs. near attachment of final phrase, (6) left vs. right attachment of middle phrase, and (7) particles vs. prepositions. In presentation, the target sentence was preceded by a disambiguating context of one or two sentences. The target sentence was edited out of context for analysis and for the perceptual experiments.

The utterances were phonetically labeled and segmented using the SRI Decipher system [15], given the sentence transcription, and the associated phoneme durations are used here for automatically detecting prosodic phrase breaks. In addition, the utterances have been hand-labeled with prosodic phrase break indices at each word boundary, where a break index corresponds to the amount of prosodic decoupling between words. We use a hierarchy of breaks, from 0 for word boundaries within clitic groups through 6 for sentence boundaries.

SYNTHESIS

Our goal in this work is to quantitatively measure the similarity between predicted and observed prosodic structures. Therefore, the prosodic phrase synthesis algorithm must not only predict the locations of prosodic phrase breaks, but also associate a numerical value to indicate hierarchical structure, as the perceptual labeling does. Below we describe algorithms that are appropriate for this application, together with some of our own modifications.

Previous Work

Gee and Grosjean [6] proposed the Phi algorithm to build a prosodic tree from syntactic structure. (Their goal was to predict psycholinguistic "performance structures;" however, we will only be interested in the prosodic tree.) The algorithm consists of a sequence of rules that progressively groups words and phrases based on syntactic structure and constituent length constraints. The rules are constrained to operate within, but not across, basic sentence clauses. First, function words and simple modifiers are grouped into ϕ-phrases using a right-branching structure. These ϕ-phrases are then grouped into I-phrases according to syntactic constituency, again using a right-branching structure. The exception is verb phrases which are grouped with either the subject or the verb's subcategorized complements, depending upon the size of these units, $N(\cdot)$, measured in number of branches (words):

- If $N(X) + N(V) \geq N(Y) \rightarrow X[VY]$
- otherwise $\rightarrow [XV]Y$

These constituents are then further bundled using a left-branching rule until all elements in the clause are included, and then clauses are bundled in a left branching structure. The degree of separation between two words, which we will refer to as a ϕ-break, is given by the number of nodes in the tree dominated by and including the node at this boundary.

A second performance structure algorithm, the Psy Algorithm, is proposed by van Wijk [14] as being more directly tied to linguistic notions of prosodic structure. The Psy algorithm requires knowledge of the location of intonational phrase boundaries and is based on a flatter prosodic structure. Unfortunately, prediction of intonational phrase boundary location is a difficult problem, so this approach was not investigated here. However, recent work in this area shows much promise [17,16], and the Psy algorithm might be interesting to pursue in the future.

Modifications to the Phi algorithm have been proposed by Bachenko and Fitzpatrick for speech synthesis applications [1]. The main difference lies in Bachenko and Fitzpatrick's claim that prosodic phrase boundaries can extend across syntactic boundaries, including clause boundaries, provided that balancing constituent length requires it. Specifically, they have modified Gee and Grosjean's verb balancing rule to include a wider range of syntactic constituents available for grouping in the verb phrase. In addition, constituent length is determined by the number of phonological words, rather than the number of words. (A phonological word is defined as a single content word or a content word combined with one or more function words that are orthographically distinct but are not separated by prosodic boundaries). In this paper, we will use "Bachenko/Fitzpatrick algorithm" to refer to the ϕ-break prediction algorithm which incorporates their modifications, noting that their work was not aimed at predicting numerical break indices.

Limitations and Modifications

An analysis of the ϕ-breaks predicted by the Phi algorithm and the Bachenko/Fitzpatrick algorithm for the ambiguous sentence corpus identified some weaknesses in the algorithms which we discuss below. In addition, we describe modifications to the Bachenko/Fitzpatrick algorithm, which we found to more often reflect observed prosodic structure.

The verb balancing rule, using either method of counting constituent length, did not always yield breaks that were

consistent with our data. We often observed the verb grouping with the subject when it was predicted by both algorithms to group with the following ϕ-group. For example, consider the labeling of the sentence

```
Marge would never deal in any guise.
    5     0     1    3 1   0
```

The largest break, after "Marge", was not perceived in any of the four spoken renditions. A more appropriate labeling would be

```
Marge would never deal in any guise.
    3     0     1    5 1   0
```

Based on this and other examples, we proposed the following revised verb balancing rule:

- If $N(X) + N(V) \geq N(V) + N(Y)$ $\rightarrow X[VY]$
- otherwise $\rightarrow [XV]Y$

Using this algorithm with a constituent counting function based on words rather than on phonological words seemed to be somewhat more consistent with our data, but this aspect should be confirmed through further study.

A second area where problems occured was in allowing prosodic units to contain clause boundaries. Although it is in general a positive feature of the Bachenko/Fitzpatrick algorithm, the predicted phrase breaks are not always consistent with the observed data, as in

```
. . . , only I knew my Dad would be angry.
      8     1 0    4  0   6     1  0
```

The larger break was perceived after "knew" rather than after "Dad" in our data. (Even though the predicted phrasing might be acceptable, for our purposes it is important that it be typical.) This particular problem could be handled by adding the rule:

$$\text{If } Y \text{ is NULL, } XV_1SV_2Y \rightarrow XV_1[SV_2]$$

where S is the subject corresponding to V_2. The resulting ϕ-break labels are then:

```
. . . , only I knew my Dad would be angry.
      8     1 0    6  0   4     1  0
```

Again, this rule needs further investigation because it may require an associated constituent length constraint.

A further limitation with both previous algorithms is the treatment of parenthetical phrases.

Algorithm	Correlation
Gee & Grosjean	0.74
Bachenko/Fitzpatrick	0.69
B-F + new verb rules	0.73
B-F + all modifications	0.73

Table 1: Correlation of predicted ϕ-breaks with hand-labeled perceived breaks for different synthesis algorithms.

```
They know, you realize, your goals.
    0     3  0        5     0
```

In our observations, parenthetical phrases are bracketed by nearly equal breaks. We therefore added a rule to increase the smaller ϕ-break at a parenthetical boundary to the size of the break at the other side of the parenthetical phrase, as in

```
They know, you realize, your goals.
    0     5  0        5     0
```

Evaluation

The different synthesis algorithms were evaluated by computing the correlation between the predicted ϕ-breaks and the hand-labeled break indices. A potential problem is that the hand-labeled indices are constrained to range from 0 to 6, while the ϕ-breaks are theoretically unbounded. However, there seemed to be a roughly linear association between the two labeling schemes in principle, apart from the specific rules for predicting groupings, and therefore it was felt that correlation would be a meaningful measure. The correlations given in Table 1 represent the average over seventy sentences from each of four speakers.

The original Phi algorithm actually had the highest performance, although average performance was similar for all four algorithms. The Phi algorithm predictions are more highly correlated to observed data in most syntactic categories in our database. Relative to the Phi algorithm, the Bachenko/Fitzpatrick algorithm offered slight improvements for parentheticals, main-main structures, and far attachments. Our modified algorithm was similar to the Phi algorithm, but having better performance for parentheticals and non-tags and somewhat worse performance for non-parentheticals and left attachments. Our algorithm was generally better than the Bachenko/Fitzpatrick algorithm, except for a significant performance degradation for left attachments.

Overall, results indicate that, while relaxation of clause boundary constraints is useful, a more conservative set of

rules may more accurately reflect observations. The verb association rule introduced here addresses one problem, that of verb attachment across a clause boundary. In addition, the length constraints that influence prosodic grouping become more important with more flexible syntactic constraints, which explains the improvement associated with the revised verb balancing rule.

ANALYSIS

After the synthesis component predicts the prosodic breaks of candidate parses, the analysis component uses a similarity measure to compare the match between the predicted and observed prosodic breaks for different possible interpretations. Clearly, in a speech understanding system, the observed prosodic breaks must be automatically detected and the algorithm used is described below. Given sequences of predicted and automatically detected breaks, many different similarity measures are possible. The results of [13], which suggest the importance of *relative* break size, motivate the correlation measure investigated here.

Automatic Labeling

Other work has reported an algorithm for automatically detecting prosodic break indices using a seven-state hidden Markov model [10], where each state represented a different break index. The feature used in that system was normalized duration in word-final syllable rhyme; a measure of the duration lengthening many researchers have observed at phrasal boundaries (e.g., [7,5]). Though pre-boundary lengthening is a particularly important cue, several other acoustic cues are also used to mark prosodic phrase boundaries, including breaths, pauses, boundary tones, and rhythm changes. In order to make use of these more diverse cues and increase the accuracy of our break detection algorithm, we have recently modified the algorithm to use a discrete HMM with a binary tree quantizer that can incorporate multiple non-homogeneous features. The algorithm is described briefly here; further details can be found in [18].

As in previous work, the first step of processing is to determine phoneme durations. These can be obtained from the output of the speech recognizer. Since inherent phone duration is the main contributor to variance in duration [7], segment durations are normalized according to phone-dependent means and variances. The means and variances are themselves adapted according to an estimate of the long-term speaking rate, using an algorithm motivated by the speaking rate differences given in the data in [5]. (This is somewhat different from the tracking algorithm reported in [18].)

The current system can combine several different features; we have thus far investigated the following:

- absolute duration of following pause;
- average normalized duration of the phonemes in the word-final syllable rhyme (pre-boundary lengthening);
- difference between average normalized duration of syllable rhyme and offset (to distinguish boundaries from phrasal prominence [4]);
- difference between the averages of normalized duration before and after the boundary (rhythm changes); and
- a flag indicating whether or not the word contains any stressed syllables (which was not included in [18]).

The use of a classification tree [3] provides a means of classifying feature vectors with non-homogeneous elements and, in fact, the quantizer can be designed jointly with the HMM [11]. Once the feature vectors for each word boundary are available, we uncover the sequence of break indices most likely to have produced them by using Viterbi decoding to recover the state sequence.

Scoring

In order to evaluate alternative interpretations of an utterance, we need to be able to compare the synthesised prosodic breaks with the automatically labeled break indices in some quantitative way. One measure might be a Hamming distance between binary sequences where a "1" indicates the location of a major prosodic phrase break. The difficulty with this approach is that it has been shown that major phrase breaks alone are often insufficient to disambiguate an utterance [13]. Thus we need to assign a score based on the agreement between the synthesisized break hierarchy and the automatic labels for an utterance.

The simplest method, and the one used here, is to compute the correlation between the two sets of labels. For example, consider the sentence *They may wear down the word.* The word *down* may be either a particle or a preposition in this sentence. The Gee and Grosjean ϕ-breaks for these two interpretations are $(1,1,0,3,0)$ and $(1,0,4,1,0)$, respectively. The break indices assigned to one reading of this sentence are $(1,1,4,1,0)$, and the correlations with the particle and preposition interpretations are -0.27 and 0.96, respectively. Thus, we select the parse in which *down* functions as a preposition as representing the speaker's intended meaning.

This scoring method is effectively a matched filter detection system, with the exception that we are not normalizing for "signal energy". Using this interpretation, it might be possible to incorporate the greater salience of intonational phrase boundaries (4,5) [13] through a weighted (as opposed to Euclidean) distance measure.

Maximum correlation can be used as a criterion for choosing among candidate parses. Occasionally, the correlations

for two candidates will be almost identical. In this case, we can either allow the algorithm to equivocate (assuming some other level of processing can resolve the ambiguity), or we can arbitrarily choose one parse as we do in the experiments described here. Another alternative would be to use the correlations to rank parses or sentence hypotheses. The rank or score might be used in combination with other knowledge sources, as in [9], to choose the correct sentence interpretation.

EXPERIMENTS

We have tested our analysis-by-synthesis approach by using it to perform the same task that the human subjects in [13] were asked to perform. Specifically, we attempt to select which of two interpretations was intended by the speaker. For each test utterance, we use the automatic labeling algorithm to label the break indices in the utterance and the synthesis algorithm to generate the prosodic breaks for the two candidate parses. We then compute the correlation between the labeled break indices and the synthesized prosodic breaks for each candidate parse and select the parse with the largest correlation. In the event of a tie, the first sentence in the pair is chosen.

The models used for the automatic labeling algorithm were speaker-independent models trained using data from three speakers. Rotation (train on 3 speakers, test on 1) was used to obtain results averaged over all four. The tree quantizer had a codebook size of 70.

To gain insight into the effect of break index labeling errors on the performance of our disambiguation scheme, we also conducted the experiment using the hand labeled break indices in the corpus. The results of these experiments are summarized in Table 2 for each of the 14 types (7 pairs) of syntactic ambiguity. For comparison, Table 2 also contains the results Price *et. al* [13] report for the human subjects.

The results based on the hand-labeled break indices again show that there is very little difference between the synthesis algorithms. As indicated by the correlation with hand labels (see Table 1), the Gee and Grosjean algorithm gave the best performance. The identification accuracy is comparable to humans in all but two cases: the non-parenthetical and non-apposition categories. This could be a weakness of either the synthesis algorithm, the similarity measure, or an artifact of the tie-breaking rule.

When we use automatically labeled break indices, there is a loss in performance. Even so, the algorithm correctly disambiguates 74% of the sentences, and this represents 88% of the human performance and 89% of the performance obtained with hand labels. Moreover, if we exclude the parentheticals and appositions, the automatic algorithm achieves 79% disambiguation as compared to human performance of 81% for the same categories.

DISCUSSION

In summary, we have demonstrated that automatically detected prosodic break indices contain enough information to achieve disambiguation close to the level of human performance. We have considered different synthesis algorithms which appear to be quite useful for this task. Little difference was observed between the synthesis algorithms, but evaluation on a larger task domain would probably yield more insight into this issue.

While these results demonstrate feasibility of the analysis-by-synthesis approach to disambiguation, the work needs to be extended in several ways. First, the current synthesis algorithm is not implemented automatically because we did not have access to machine parses for these sentences. Automatic implementation of the synthesis algorithm and integration with a parser is an important next step. As mentioned earlier, additional modifications to the synthesis algorithm or investigation of a variation based on the Psy algorithm might also be useful.

Second, the automatic break index labeling algorithm needs to be extended to achieve closer agreement with the hand labels. Although the correlation between the two is already 0.86, there is a loss of disambiguation performance. The principal reason for this loss can been seen by noting that the machine label differs from the hand label by no more than one 93% of the time for all the boundaries except those with hand labels of 3 and 4. These boundaries correspond to intermediate and intonational phrases [12] and in these cases, the current algorithm produces labels within 1 of the hand labels only 57% of the time. This is hardly surprising since intermediate and intonational phrases are marked by intonation [2] and our labeling algorithm currently has no pitch features. Thus a principal extension which needs to be investigated, is the inclusion of intonation features such as boundary tones. Since these are the principal cue for the larger breaks, we expect that their inclusion will improve performance considerably.

In addition, it might be useful to investigate other similarity measures. In particular, a measure which more highly weighted the larger break indices might be useful. Finally, it will be important to consider spontaneous speech domains, which may require an entirely different synthesis algorithm for predicting phrase breaks.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge Patti Price and Stefanie Shattuck-Hufnagel for their valuable suggestions and insights. Thanks also to John Butzberger and Hy Murviet at SRI for their help in obtaining the phonetic alignments. This research was jointly funded by NSF and DARPA under NSF grant number IRI-8905249.

Ambiguity	Hand Labels G-G	B-F	M-I	M-II	Machine & G-G	Human Perception
+ Parenthetical	50	80	80	75	50	77
- Parenthetical	90	40	35	45	65	96
+ Apposition	90	90	90	100	90	92
- Apposition	55	70	70	65	35	91
Main-Main	65	100	85	85	85	88
Main-Subordinate	85	45	55	55	95	54
+ Tag	90	100	100	100	90	95
- Tag	100	80	95	95	100	81
Far Attach	100	60	65	65	80	78
Near Attach	40	60	50	50	45	63
Left Attach	100	100	100	100	90	94
Right Attach	100	100	100	100	70	95
Particle	100	100	100	100	65	82
Preposition	95	95	95	95	70	81
Average	83	80	80	81	74	84

Table 2: Percent correct disambiguation as a function of different syntactic ambiguities for: different synthesis algorithms comparing to hand-labeled breaks (G-G: Gee/Grosjean, B-F: Bachenko/Fitzpatrick, M-I: B-F with verb rule modifications, M-II: B-F with all modifications); the best-case synthesis algorithm comparing to automatically labeled breaks; and human perceptual results.

REFERENCES

1. J. Bachenko and E. Fitzpatrick "A Computational Grammar of Discourse-Neutral Prosodic Phrasing in English", *Computational Linguistics,* Vol. 16, No. 3, pp. 155-170 (1990).
2. M. Beckman and J. Pierrehumbert (1986) "Intonational Structure in Japanese and English," *Phonology Yearbook 3,* ed. J. Ohala, pp. 255-309.
3. L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone (1984) *Classification and Regression Trees.* Wadsworth and Brooks/Cole Advanced Books and Software, Monterey, CA.
4. W. N. Campbell (1990). "Evidence for a Syllable-Based Model of Speech Timing," *Proceedings Int. Conf. Spkoken Language Processing,* pp. 9-12, Kobe, Japan.
5. T. H. Crystal and A. S. House (1988), "Segmental durations in connected-speech signals: Current results" *Journal of the Acoustical Society of America,* Vol. 83, No. 4, pp.1553-1573.
6. J. P. Gee and F. Grosjean (1983) "Performance Structures: A Psycholinguistic and Linguistic Appraisal," *Cognitive Psychology,* Vol. 15, pp. 411-458.
7. D. Klatt (1975) "Vowel Lengthening is Syntactically Determined in a Connected Discourse," *J. Phonetics 3,* 129-140.
8. I. Lehiste (1973) "Phonetic Disambiguation of Syntactic Ambiguity," *Glossa* 7:2.
9. M. Ostendorf, A. Kannan, S. Austin, O. Kimball, R. Schwartz and J. R. Rohlicek (1991), "Integration of Diverse Recognition Methodologies Through Reevaluation of N-Best Sentence Hypotheses," this proceedings.
10. M. Ostendorf, P. Price, J. Bear and C. W. Wightman (1990) "The Use of Relative Duration in Syntactic Disambiguation," *Proceedings of the 4th DARPA Workshop and Speech and Natural Language,* pp. 26–31. A shorter version appears in *Proceedings Int. Conf. Spkoken Language Processing,* pp. 13–16, Kobe, Japan.
11. M. Ostendorf and R. Rohlicek (1990) "Joint Quantizer Design and Parameter Estimation for Discrete Hidden Markov Models," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing,* pp. 705-708, Albuquerque, NM.
12. J. Pierrehumbert (1980) *The Phonology and Phonetics of English Intonation.* PhD Thesis, Massachusetts Institute of Technology.
13. P. Price, M. Ostendorf, S. Shattuck-Hufnagel, C. Fong "The Use of Prosody in Syntactic Disambiguation," manuscript submitted to the *Journal of the Acoustical Society of America.* A shorter version appears in this proceedings.
14. C. van Wijk (1987), "The PSY behind the PHI: A Psycholinguistic Model for Performance Structures," *Journal of Psycholinguistic Research* 16:2, pp. 185–199.
15. M. Weintraub, H. Murveit, M. Cohen, P. Price, J. Bernstein, G. Baldwin and D. Bell (1989) "Linguistic Constraints in Hidden Markov Model Based Speech Recognition," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing,* pp. 699–702, Glasgow, Scotland.
16. N. Veilleux and M. Ostendorf, "A Hierarchical Stochastic Structure for Automatic Prediction of Prosodic Boundary Location," manuscript.
17. Wang and J. Hirschberg (1991), "Predicting Intonational Boundaries Automatically from Text: the ATIS Domain",this proceedings.
18. C. W. Wightman and M. Ostendorf (1991), "Automatic Recognition of Prosodic Phrases," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing,* Toronto, Canada.