Learning Prosodic Features using a Tree Representation

Julia Hirschberg, Owen Rambow

AT&T Labs — Research
Florham Park, NJ, USA

julia@research.att.com, rambow@research.att.com

Abstract

We describe experiments designed to learn associations between two types of intonational features, pitch accent and phrasing, from a tree-based corpus annotated with various intonational and syntactic features, for a concept-to-speech system. We show that using novel tree-based features improves the quality of boundary prediction over using only the linear order-based features normally used in text-to-speech.

1. Introduction

Assigning intonational features such as phrasing and prominence in text-to-speech (TTS) and concept-to-speech (CTS) systems is a much-explored but still perplexing problem. In TTS systems, the difficulties revolve around the need to interpret an input text sufficiently to make plausible decisions about how a human reader might produce the text orally. CTS systems are widely believed to enjoy major advantages over TTS systems in intonational assignment, since they "know" the meanings they wish to convey. Nonetheless, there are still serious problems involved in reliably associating intonational features with other (syntactic, semantic, discourse) linguistic features of the message to be realized. Also, unlike TTS systems, CTS systems cannot make use of most punctuation. And how to select among multiple plausible configurations of intonational, syntactic, and lexical features, e.g., without giving one feature type unwarranted prior status, represents an open question. Corpus-based approaches are appealing since they allow us to experimentally evaluate the contribution of different features; in practical terms, they allow us to port systems to new domains, genres, or speaking styles. However, assembling a sufficiently rich corpus is a major task, as is determining which features might be good predictors of intonational properties.

In this paper we describe experiments designed to learn associations between two types of intonational features, pitch accent and phrasing, for a CTS system from a tree-based corpus annotated with various intonational and syntactic features. The novel contribution of the paper is that we use a dependency representation of syntax from which we derive a large set of "deep" features. Using the same set of data, we address two different issues:

- In a TTS system, how much can parsing contribute? We show that syntax-based features can predict phrase boundaries significantly better than features derivable simply from the linear order of words and superficial processing (as is done in most TTS systems).
- In a CTS system, how much does the linear order contribute? Put differently, can intonational features be assigned on the syntactic tree prior to the determination of linear order, or do we need to wait until the linear order is determined? We show that phrase boundaries can be predicted significantly better if linear order is also taken into account.

In Section 2 we describe previous research on the assignment of intonational features in TTS and CTS. In Section 3 we discuss the corpus used to train and test our assignment procedures. Section 4 describes the machine learning algorithm we employed and the features we used. In Section 5 we discuss our results. Finally, in Section 6 we sum up the work to date and describe future research plans.

2. Previous Research

Most recent research on predicting prosodic assignment for text-to-speech (TTS) systems has largely focused on predicting phrasing and prominence from simple analysis of input text [1, 2, 3]. The best performing of these techniques employ automatically generated information such as part-of-speech labels, inferred constituency information, syllabification, length of sentence and other distance measures, punctuation, and inferred information about other prosodic features of the input text. Such predictions have been trained on prosodically labeled corpora, and employ statistical and machine learning techniques. These approaches have achieved accuracy rates of 80-85% for prominence prediction and up to 95% for phrasing, on test sets drawn from prosodically labeled read news stories or elicited speech corpora such as the ATIS corpus. However, there also is a long tradition of efforts to associate phrasing decisions, in particular, with more sophisticated syntactic analyses [4, 5, 6, 7, 8]. While most of these proposals have had to assume that parsing technology will improve to provide the requisite level of accuracy for their inputs, [8] has in fact been able to demonstrate performance better than simpler techniques using the uncorrected output of the Collins parser [9] to provide syntactic information for a corpus-based study of intonational phrasing. In CTS-oriented work, Pan and McKeown [10] have investigated how features such as deep syntactic/semantic structure and word informativeness correlate with accent placement. Pan and Hirschberg [11] have found effects of word collocation on accent placement.

3. Corpus

The corpus used in our experiments consists of 11,074 words of *Wall Street Journal* text from the Penn Tree Bank (PTB) [12] (507 sentences, average sentence length of 21.8 words).[1] The hand-annotated phrase structure tree from the PTB was converted to a dependency tree using the head percolation

[1] We thank the AT&T TTS group, in particular Alistair Conkie, Volker Strom, and Ann Syrdal, as well as Srinivas Bangalore, for help in assembling the corpus.

technique first used by Magerman [13], and the tokenization changed to that used in voice transcriptions. A sample dependency tree is shown in Figure 1. The text was read by a female professional speaker. The recordings were transcribed and annotated with ToBI labels [14] by two trained and experienced labelers. Subsequently, the ToBI labels from the annotated transcription were added to the dependency tree as follows: accent labels were added to the node of the lexical item with which they are associated, while boundary labels were added to the lexical item preceding the boundary. For our current experiments, we collapsed ToBI accent labels to form a binary distinction accented vs. deaccented. We also collaprsed the ToBI break index label in two ways: intonational phrase (level 4) vs. all other break indices and intermediate phrase (level 3 or 4) versus all other break indices. The data used for our experiments consisted in the 11,074 words from the corpus annotated with the ToBI labels as just described, which provide the classifications to be learned, and with syntactic features which are described in the next section, providing the independent features used in the learning.

4. Machine Learnings Experiments

This section describes experiments using the machine learning program Ripper [15] to automatically induce prediction models, using features derivable from the syntactic tree and from the linear order. Like many learning programs, Ripper takes as input the classes to be learned, a set of feature names and possible values, and training data specifying the class and feature values for each training example. In our case, the training examples are the words from the training corpus as described in Section 3. Ripper outputs a classification model for predicting the class of future examples. The model is learned using greedy search guided by an information gain metric, and is expressed as an ordered set of if-then rules.

In the following, we summarize the features we used in determining rules for prosody. In a generation system, the syntactic structure of a sentence is fully (and correctly) determined; in addition, the generation system must of course also determine the linear order of the lexemes. Accordingly, the features we use fall into two classes: "surfacy" features that can be deduced from the linear order alone ("linear order features"), and "deep" features that can only be deduced from the syntactic structure. The latter group we divide further into two groups: those syntactic features that can be determined with some reliability from a linear sequence without the need for full parsing ("linear syntactic features"), and those that require full parsing ("tree-based syntactic features"). For reasons of space limitations, we omit some useful features in the presentation.

4.1. Linear Order Features (LIN)

The following features can be deduced from the linear order alone. We refer to these features as LIN. Note that we are assuming that part-of-speech (POS) taggers of sufficient quality are available.

• Part-of-speech (**POS**). This is taken from a small set (12 tags) which distinguishes only the main word classes, including all function word classes.

• Length of sentence in words (**LEN**).

• Normalized relative position of that node in the sentence (**RPS**), a number between 0 (beginning of sentence) and 1000 (end of sentence).

We use these features for a five-word window around the current word. Subscripts denote features applying to words in the window (e.g., $\mathbf{POS}_{-2}$ is the part-of-speech of the word two to the left of the current word).

4.2. Linear Syntactic Features (LIN+)

This is one feature which can be determined with some reliability from a linearly ordered sentence without the need for a full parser. We refer to the linear features plus this one as LIN+.

• Supertag (**STAG**). A supertag is like a part-of-speech tag, but it contains more information, specifically the lexeme's active valency (what arguments it requires), passive valency (to what lexemes it can attach), and the manner in which the arguments are realized (for example, whether the verb is in active or passive voice). The supertags are names of trees in a Tree Adjoining Grammar of English [16]. Bangalore and Joshi [17] discuss trigram-based models for automatic supertagging, which achieve accuracy figures of 91%.

4.3. Tree-Based Syntactic Features (TREE)

We use the syntactic dependency trees to represent syntactic structure. In a syntactic dependency tree, each node is labeled with a lexeme of the target sentence; there are no nodes that represent intermediate phrasal projections such as VP. The daughters of a node are the lexical arguments and adjuncts of that node; in addition, we assume function words depend on their major-class lexeme, i.e., auxiliaries depend on their verb, determiners on their noun, and so on. The arcs are labeled with a small set of grammatical functions (**FUNC**): 0 for the subject, 1 for the direct object or object of a preposition, 2 for the indirect or prepositional object, ADJ for all types of adjuncts, and FUNC for the arcs that relate function words to their major-class lexeme. We use the following features; they are illustrated using the sample tree in Figure 1. We refer to the features presented in this subsection as TREE. We will refer to the union of all features as LIN+TREE.

• Whether or not the word is at the right boundary of a subsuming major constituent, i.e., one headed by a noun or a verb other than itself (**RMC**). Note that this feature requires more than just "noun chunking", since we also need to know whether a post-nominal PP attaches to the NP or not.

• Whether or not the word is at the right edge of a coordinated constituent (**RCC**).

• The size of the subtree headed by the current word, including the node (**STZ**). For example, *board* in Figure 1 has a value of 2.

• The number of siblings of the current word, including the current word (**SIB**). For example, for this feature *board* has a value of 1.

• The distance in arcs in the tree from the current word to the next word in the surface string (**TRD**). For example, the node labeled *board* has a value of 2: from *board* to *as* one must traverse two arcs.

In addition, we use the **POS**, **STG**, and **LEX** features described above for the mother and grandmother nodes in the syntax tree (denoted by subscript m and g).

5. Results

Results of our experiments are summarized in Figure 2. We report the average error rate obtained during a five-fold cross-validation, as well as recall, precision, and the F-measure[2] ob-

[2] $F = 2RP/(R + P)$, where R is recall and P precision.

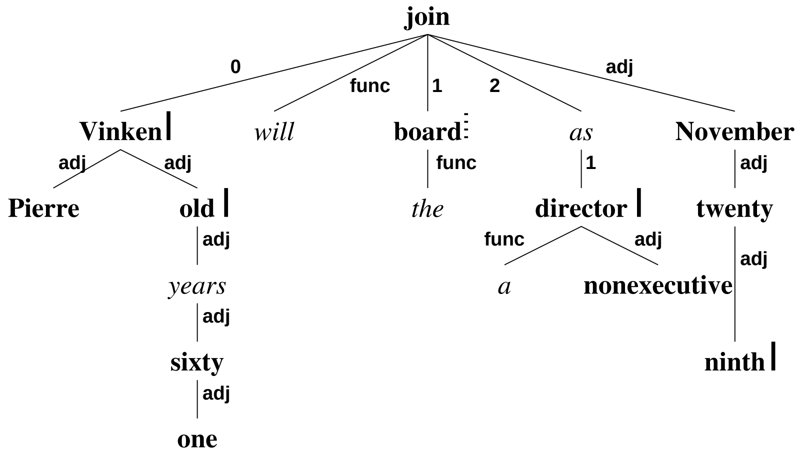

Figure 1: Dependency syntax tree for *Pierre Vinken, sixty one years old, will join the board as a nonexecutive director November twenty ninth.* Words in boldface have a pitch accent, while words in italics do not. A solid line following a word indicates an intonational phrase boundary following that word, while a dotted line indicates an intermediate phrase boundary.

Class	BL	TREE			LIN			LIN+			LIN+TREE		
		Acc	Rec/Prec	F	Acc	Rec/Prec	F	Acc	Rec/Prec	F	Acc	Rec/Prec	F
Pitch accent	40.7 (yes)	11.8 (.30)	96.6/84.7	90.3	13.0 (.35)	97.2/82.0	89.0	12.0 (.27)	96.8/84.3	90.1	12.0 (.21)	96.4/84.4	90.0
Interm. phrase	33.6 (no)	16.6 (.21)	65.1/81.6	72.4	17.0 (.57)	68.4/75.9	71.9	14.9 (.20)	76.6/80.2	78.4	13.9 (.57)	77.2/79.5	78.3
Inton. phrase	20.0 (no)	14.7 (.39)	63.6/61.8	62.7	14.2 (.64)	56.2/67.8	61.5	12.5 (.33)	56.2/75.2	64.3	10.2 (.17)	64.1/77.5	70.1

Figure 2: Error rates (Acc), recall (Rec), precision (Prec), and F-measure (F) for different feature sets. The baseline (BL) is the error rate on the majority choice (given in parentheses).

tained when training on a trainig corpus of approximately 8,800 words, tested on a test corpus of 2,200 words. Significance is assessed on the accuracy figures using the confidence intervals supplied by Ripper; these are given in parentheses after the figures. If the difference between two figures is greater than twice the sum of their intervals, then the difference is significant ($p < 0.05$). We see that for both phrase boundary types, LIN+TREE performs significantly better than either TREE or LIN alone. With respect to the finding that the performance of linear features can be improved upon by using more sophisticated syntatcic features, our findings support those of [10, 8]; our results that linear and syntactic features together improve on syntactic features are new. Furthermore, we see that for intermediate phrase boundaries, LIN+ performs significantly better than LIN, while for intonational phrase boundaries, LIN+TREE performs significantly better than LIN+. Despite the fact that the f-measure does not increase for intermediate phrase boundaries, we expect that more data will make both the comparison of LIN and LIN+, and of LIN+ and LIN+TREE significant for both boundary types, showing the power of the supertags (the only feature in LIN+ not in LIN). LIN and TREE do not differ significantly. We can also see that pitch accent placement is not significantly affected by the choice of feature set. The low contribution of the syntactic tree features can be explained by the high level of accenting that our speaker performed and the resulting regularity: if we use only two features – **POS** and **LEX** (with no window at all) – we obtain the same results as shown in Figure 2.

Comparing these results to prior research on the prediction of prosodic features from text is not straightforward, due to differences in corpus domain, style, and size; the description of dependent variables; and the methods used to evaluate predictive accuracy. The most comparable study is [8] which improved on [3]. Both used an 89,103-word training corpus (including punctuation) labeled with break indices. [8] achieved precision, recall and F-ratios predicting intonational phrase boundaries (ToBI level 4 vs. other) ranging from 86.6%, 52.7%, and 65.5% when trained on a 10,000-word subset of the corpus (comparable in size to our training set) and tested on a held-out test set, to 90.1%, 80.0% and 84.8% when trained on a 60,456-word subset. Earlier researchers obtained recall scores in the 75-85% range (with false alarm rates ranging from 1-11%) on smaller training and test sets of read and spontaneous speech [5, 18, 19].

1	yes if **STG**=A_NXN, **RMC**=Y (521/43)
2	yes if **STG**=A_NXN, **TRD**>=2, **SIB**<=1 (184/55)
3	yes if **STG**=A_NXN, **TRD**>=3, **LEX**$_g$≠adj (19/10)
4	yes if **STG**=A_NXN, **POS**$_{-1}$ = N, **ROL**$_{+2}$=adj, **STZ**>=4, **STG**$_1$ ≠B_nxPnx, **LEX**$_g$ ≠adj (73/11)
5	yes if **STG**=A_NXN, **TRD**>=2, **ROL**$_{-1}$=adj, **TRI**<=0 (26/8)
6	yes if **RMC**=Y, **STG**≠B_ABB (166/13)
7	yes if **STG**=A_NXN, **STZ**>=8, **RCC**=Y (21/2)
8	yes if **STG**=A_NXN, yps=V, **STZ**>=3 (47/30)
9	default no (6932/699)

Figure 3: Sample rule set for intonational phrase boundaries, using LIN+TREE features.

We now discuss a sample rule set for predicting intonational phrase boundaries, using the TREE features (Figure 3). The Ripper rule sets contain an ordered list of rules. Each rule contains a conjunction of conditions and a consequent classification. Each rule (except the first) only applies if the preceding ones do not. The rules are annotated in parentheses with the number of examples from the test corpus of that particular fold which they classify correctly and incorrectly, separated by a slash. The supertag condition **STG**=A_NXN means that the word is a noun which serves as an argument (the nouns at the end of numbered arcs), as opposed to a noun modifying a noun (*Pierre*), a verb (*November*), or an adjective (*years*). The first, very productive rule states that an argument noun which is at the right edge of its containing major constituent (headed by a different noun or a verb) is followed by an intonational phrase boundary. In our example (Figure 1), it does not apply to any nodes – the only nodes at the right edge of a subsuming major constituent are *old* and *ninth*, neither of which are argument nouns. The second rule states that an argument node which has no other siblings and whose tree distance to the next word is at least two arcs is followed by an intonational phrase boundary. This applies correctly to *director*, but incorrectly to *board* as well. Note the fairly high error rate on this rule in the test corpus. The third rule inserts a boundary after an argument noun if the tree distance is at least three and the grandmother is not an adjunct; this applies to *Vinken*: the next word, *sixty*, is exactly three arcs away. (The grandmother condition is met because there is no grandmother.) Rule 6 applies, correctly, to *old* and to *ninth*. None of the other rules apply (since there are no other argument nouns in the sentence), so this rule set gets one prediction wrong on this sentence of 17 words (6% error rate).

6. Conclusion

Using a read corpus of newspaper text which was annotated for intonational and for novel syntactic features, we have automatically learned rules for placing pitch accents, intermediate phrase boundaries, and intonational phrase boundaries. We have shown that for our corpus, pitch accent can be predicted simply from the word and its part of speech, while for both boundary types, using both "deep" syntactic features derived from the unordered tree and "surfacy" features derived from the linear sequence of words improves over the use of just one of these feature sets. We conclude that for TTS systems, deeper parsing will be valuable, and that in CTS systems, the decisions on intonational features should only be taken once linear order has been determined (i.e., nearly at the end of the generation process). We leave to future work the incorporation of frequency-based features, and features reflecting the discourse context.

7. References

[1] M. Ostendorf, P. Price, J. Bear, and C. W. Wightman, "The use of relative duration in syntactic disambiguation," in *Proceedings of the Speech and Natural Language Workshop*, Hidden Valley PA, June 1990, DARPA, pp. 26–31, Morgan Kaufmann.

[2] Michelle Q. Wang and J. Hirschberg, "Automatic classification of intonational phrase boundaries," *Computer Speech and Language*, vol. 6, pp. 175–196, 1992.

[3] Julia Hirschberg and Pilar Prieto, "Training intonational phrasing rules automatically for English and Spanish text-to-speech," *Speech Communication*, vol. 18, pp. 281–290, 1996.

[4] Bengt Altenberg, *Prosodic Patterns in Spoken English: Studies in the Correlation between Prosody and Grammar for Text-to-Speech Conversion*, vol. 76 of *Lund Studies in English*, Lund University Press, Lund, 1987.

[5] Joan Bachenko and Eileen Fitzpatrick, "A computational grammar of discourse-neutral prosodic phrasing in English," *Computational Linguistics*, vol. 16, no. 3, pp. 155–170, 1990.

[6] Arthur Dirksen and Hugo Quene, "Prosodic analysis: The next generation," in *Analysis and Synthesis of Speech: Strategic Research towards High-Quality Text-to-Speech Generation*, Vincent J. van Hueven and Louis C. W. Pols, Eds., pp. 131–144. Mouton de Gruyter, 1993.

[7] William Croft, "Intonation units and grammatical structure," *Linguistics*, vol. 33, pp. 839–882, 1995.

[8] P. Koehn, S. Abney, J. Hirschberg, and M. Collins, "Improving intonational phrasing with syntactic information," in *Proceedings of ICASSP-00*, Istanbul, 2000.

[9] Michael Collins, "Three generative, lexicalised models for statistical parsing," in *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, Madrid, Spain, July 1997.

[10] Shimei Pan and Kathleen R. McKeown, "Learning intonation rules for concept to speech generation," in *Proceedings of ACL-98*, Montreal, 1998.

[11] Shimei Pan and Julia Hirschberg, "Modeling local context for pitch accent prediction," in *Proceedings of ACL-2000*, Hong Kong, 2000.

[12] Mitchell M. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz, "Building a Large Annotated Corpus of English: The Penn Treebank," *Computational Linguistics*, vol. 19.2, pp. 313–330, June 1993.

[13] David Magerman, "Statistical decision-tree models for parsing," in *33rd Meeting of the Association for Computational Linguistics (ACL'95)*, 1995.

[14] John Pitrelli, Mary Beckman, and Julia Hirschberg, "Evaluation of prosodic transcription labeling reliability in the ToBI framework," in *Proceedings of the Third International Conference on Spoken Language Processing*, Yokohama, 1994, ICSLP, vol. 2, pp. 123–126.

[15] William Cohen, "Learning trees and rules with set-valued features," in *Fourteenth Conference of the American Association of Artificial Intelligence*. AAAI, 1996.

[16] The XTAG-Group, "A lexicalized Tree Adjoining Grammar for English," Tech. Rep., Institute for Research in Cognitive Science, University of Pennsylvania, 1999.

[17] Srinivas Bangalore and Aravind Joshi, "Supertagging: An approach to almost parsing," *Computational Linguistics*, vol. 25, no. 2, pp. 237–266, 1999.

[18] Hugo Quené and René Kager, "Automatic prosodic sentence analysis, accentuation and phrasing for Dutch text-to-speech conversion," Final Report 17, Research Institute for Language and Speech, Rijksuniversiteit Utrecht, Utrecht, The Netherlands, January 1990.

[19] Nanette Veilleux, *Computational Models of the Prosody/Syntax Mapping for Spoken Language Systems*, Ph.D. thesis, Boston University, 1994.