Accenting and Deaccenting: a Declarative Approach

Arthur Dirksen
Institute for Perception Research/IPO
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
E-mail: dirksen@heiipo5.bitnet

1 Introduction

One of the problems that must be addressed by a text-to-speech system is the derivation of pitch accent, marking the distinction between "given" and "new" information in an utterance. This paper discusses a language-independent approach to this problem, which is based on focus–accent theory (e.g. Ladd 1978, Gussenhoven 1984, Baart 1987), and implemented in my program PROS-3. This program has been developed as part of the ESPRIT-project POLYGLOT, and provides an integrated environment for modelling the syntax-to-prosody interface of a multi-lingual text-to-speech system.

The program operates in the following manner. First, the input text is parsed using a variation of context-free phrase-structure rules, augmented with information about "argument" structure of phrases. Next, the syntactic representation is mapped onto a metrical tree. The metrical tree is then used to derive locations for pitch accents, as well as phonological and intonational phrase boundaries.

In this approach, differences between languages are modelled entirely by the syntactic rules. Also, the system is strictly declarative, in the sense that once a piece of information is added by a rule, it is never removed. In this respect, our approach differs radically from systems which make use of derivational rules (e.g. Quené & Kager 1992). Such systems tend to become extremely complex, hard to verify and almost impossible to maintain or extend (Quené & Dirksen 1990, Dirksen & Quené *in press*). By contrast, in PROS-3 there is a conspicuous relation between theory and implementation, and the program can be extended in a number of ways.[1]

Below, I will focus on two major rules from focus-accent theory: Default Accent and Rhythmic Deaccenting. The first rule is used to model deaccenting of "given" information, e.g. the pronouns *it*, *het* and *es* in the English, Dutch and German sentences of (1), (2) and (3), respectively.

(1)a I should have read a BOOK
   b I should have READ it

(2)a ik had een BOEK moeten lezen
   b ik had het moeten LEZEN

(3)a ich hatte ein BUCH lesen sollen
   b ich hatte es LESEN sollen

The second rule is used to provide rhythmical alternations between accented and deaccented material in certain well-defined contexts, as is illustrated by the sentences of (4).

(4)a she is a NICE GIRL

[1] One extension we are currently considering is the addition of some kind of discourse model (along the lines of Hirschberg 1990) to more adequately model the "given–new" distinction. Also, some preliminary work has been done on phonological parsing (e.g. Coleman 1990, 1991; see also his paper in this volume) to derive word stress and temporal structure of words.

b she is REALLY NICE
c she is a REALLY nice GIRL
d she is REALLY a NICE GIRL

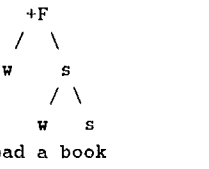
This paper is organized as follows. Section 2 briefly introduces focus–accent theory and how it may be implemented. Next, sections 3 and 4 discuss Default Accent and Rhythmic Deaccenting, respectively. In section 5, we make some concluding remarks.

2 From Focus to Accent

In focus–accent theory, metrical trees are used to represent relative prominence of nodes with respect to pitch accent. Whether a given node is accented or not is accounted for in terms of the focus/non-focus distinction.

For example, a pitch accent on *book* in the phrase *read a book* may be accounted for by assuming the metrical structure (5).

(5)



In (5), the entire phrase is marked +F(ocus), indicating that it is to be interpreted as a "new" or otherwise important addition to the discourse. The relation between the focus-marker and a pitch accent on *book* is mediated by the labels *w(eak)* and *s(trong)*, and formally accounted for by the following recursive rule:[2]

Accent Rule
For each node X, X is accented if
a. X is marked +F, or
b. X is strong, and the node immediately dominating X is accented.

[2]By convention, only weak or root nodes are marked +F, thus indicating the upper bound of what is sometimes called the "focus set".

Baart (1987) assumes that the metrical labeling of a structure is determined by syntactic/thematic properties of phrases such as specification and complementation. More generally, we assume that "arguments" which are not deaccented are strong. For example, in (1) the NP *a book* is an argument of the verb *read*. Also, a determiner takes a noun as an argument. In a PROS-3 grammar, one must make this explicit by writing rules such as those in (6).

```
(6)a  VP -> (V/NP)      (English)
   b  VP -> (NP\V)      (Dutch/German)
   c  NP -> (Det/N)
```

In such rules, (X/Y) or (Y\X) serves to indicate that Y is an argument of X. If we ignore deacccenting, argument structure directly determines the geometrical properties of the metrical tree, and we may read (X/Y) or (Y\X) as *weak-strong* or *strong-weak*, respectively.[3]

Also, a PROS-3 grammar must indicate which nodes are eligible for focus (normally, all major phrasal categories). If a node is eligible for focus, it must either be accented or deaccented. Words which are typically deaccented are specified as such in a lexicon.

In our implementation, a binary-branching metrical tree is used as the central data-structure, and the relation between focus and accent is defined by using sharing variables, which may become instantiated to a value "true" (=accented) or "false" (=deaccented), or remain unspecified (=not accented). The following definitions are used to implement accenting:[4]

```
accented(X) :-
  X:accent === true.
```

[3]Even though metrical trees are strictly binary-branching, multi-branching are accommodated by allowing rules such as S → (NP/(Infl/VP)).

[4]The notation has been borrowed from Gazdar & Mellish 1989; '===' is the unification operator, and Node:Attr indicates a path in a graph (or a field in a record). We assume negation by failure as in standard Prolog implementations.

```
strong(X, Y) :-
  X:accent === Y:accent.

deaccented(X) :-
  not accented(X).

focus(X) :-
  accented(X);
  deaccented(X).
```

The statement `accented(X)` may be used to assign accent to a node, or to verify that the node is acccented. The statement `strong(X, Y)`, which reads "the strong node of X is Y" implements condition b of the Accent Rule above by unifying the values for accent of X and Y. The statement `deaccented(X)` succeeds if the value for accent of X is instantiated to "false", and fails otherwise, so it may be used as a test. Similarly, the statement `not deaccented(X)` may be used to test whether it might be possible to assign accent to X, but will not instantiate any values. Finally, the statement `focus(X)` is used to assign accent to those nodes marked by the grammar writer as "eligible for focus", unless they have been deacccented.

3 Default Accent

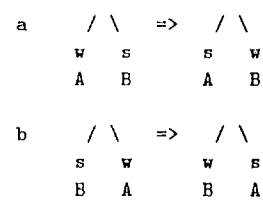
Consider again the sentences in (1), (2) and (3), and observe that when the NP *the book* is replaced by the pronoun *it*, pitch accent appears to "shift" from the NP to the most deeply embedded verb, *read*, of which it is an argument. Any differences between English, Dutch and German seem to be strictly a matter of syntax. Assuming appropriate phrase-structure rules, such as (6)a and b, this is reflected in the corresponding metrical tree. The metrical structure of the verb-phrase of (1)a, is a strictly right-branching structure which is uniformly labeled as weak-strong. The metrical trees corresponding to the verb phrases of (2)a and (3)a, shown in (7) and (8), are less uniform.

(7)

(8)

In order to account for the b-sentences of (1), (2) and (3), in which a (deaccented) pronoun replaces NP, it seems that all that is needed is a reversal of the weak-strong labeling of the VP-node. To this end, Baart (1987) assumes the following rule:

DEFAULT ACCENT



Condition: B is deaccented

In PROS-3, this rule is implemented as a filter, called STP, which takes as input a syntactic structure assigned by the parser, and produces as output a metrical tree. A typical invocation might be:

```
...
VP->(V/NP) => Prosody,
focus(VP).
```

Using the definitions of section 2, STP is defined by the following set of rules:[5]

[5]Take note that we are rather frivolous in using the slash-notation to encode both argument structure and metrical structure, though, of course, the two are distinct. That is, the metrical tree does not replace argument structure, but is merely its realization in the domain of sentence prosody.

STP

```
a  Z->(X/Y) => Z->(X\Y) :-
     deaccented(Y),
     strong(Z, X).

b  Z->(X/Y) => Z->(X/Y) :-
     not deaccented(Y),
     strong(Z, Y).

c  Z->(Y\X) => Z->(Y/X) :-
     deaccented(Y),
     strong(Z, X).

d  Z->(Y\X) => Z->(Y\X) :-
     not deaccented(Y),
     strong(Z, Y).
```

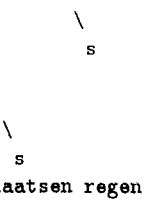
Cases a and c implement Default Accent, whereas b and d represent the "normal" case.

4 Rhythmic Deaccenting

Rhythmic factors provide a second source of deaccenting phenomena. They apply to structures such as (9), representing (4)c from section 1, and (10), representing the Dutch sentence "er is op VEEL plaatsen REGEN voorspeld" (there is in MANY places RAIN predicted), meaning: it has been predicted that it will rain in many places.

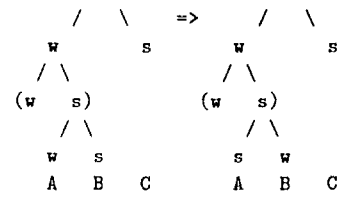
(9)

(10)



Although the pitch accent patterns implied by these structures are well-formed, there is a strong preference for deaccenting *nice* in (9) and *plaatsen* in (10). In order to account for these phenomena, we assume the following optional rule (adapted from Baart 1987):

RHYTHM RULE



In this rule, brackets indicate a substructure which may repeated zero or more times. A further requirement is that nodes A, B and C are not deaccented.

The Rhythm Rule differs from Default Accent in that it is not a local rule: its structural change, the weak-strong reversal of A and B, is dependent on the presence of a node C whose weak sister-node dominates A and B in a rather complex manner. One way to implement such context-sensitive rules in a declarative framework, is to use feature percolation. Space does not permit us to work out the implementation in full detail (there are also some additional requirements to be met), but the following should give the reader some idea.

First, we add a new case to the STP-filter above, implementing the structural change of the Rhythm Rule, and marking the resulting structure with a feature annotation indicating that the Rhythm Rule has "applied":

```
Z->(X/Y) => Z->(X\Y) :-
  not deaccented(X),
  not deacccented(Y),
  strong(Z, X),
  Z:rhythm_rule === true.
```

Next, we make sure that this feature is percolated upwards in weak-strong configurations, and blocked wherever necessary in order to filter out over-generation.

5 Conclusion

As emphasized above, PROS-3 is a language-independent system for deriving sentence prosody in a text-to-speech system. This is true, of course, only to the extent that focus–accent theory and its major rules are universals of linguistic theory. Clearly, the proof of the pudding is in the eating. At IPO, PROS-3 is currently being evaluated for Dutch, using a grammar of about 125 rules and a lexicon of some 80,000 word forms derived from the CELEX lexical database. Also, we are working on grammars and lexicons of comparable size and scope for English and German, and PROS-3 is used in the POLYGLOT-project for several European languages.

Although preliminary results are encouraging, there are also problems which need mention. First, the focus/non-focus distinction is modelled by rather crude heuristics (i.e. taking each major phrase as a candidate for focus, deaccenting of pronouns etc. by lexical specification). It would be nice if something more flexible and "discourse-aware" could be built in. Second, we have deliberately kept the PROS-3 grammar formalism rather simple (allowing only atomic syntactic categories), so we could guarantee fairly efficient processing. However, simple context-free rules do not disambiguate very well. Third, simple rules cannot fully take into account verb subcategorization. As a result, it is sometimes impossible to make the distinction between arguments and non-arguments, which is crucial to the metrical rules. So, what we need to do, is find an optimal compromise between sophistication of syntactic analysis and efficiency of processing. We think that PROS-3 is the right tool to do this.

6 Bibliography

Baart, J.L.G. (1987), *Focus, syntax and accent placement.* Diss. University of Leiden.

Coleman, J.S. (1990), Unification Phonology: another look at "synthesis-by-rule". *COLING 90*, Vol. 3, 79-84. ACL.

— (1991), Prosodic structure, parameter-setting and ID/LP grammar. S. Bird (ed.), *Declarative Perspectives on Phonology.* Edinburgh Working Papers in Cognitive Science, Vol. 7, 65-78.

Dirksen, A. & H. Quené (in press), Prosodic analysis: the next generation. V.J. van Heuven & L. Pols (eds.), *Analysis and synthesis of speech: strategic research towards high-quality text-to- speech generation.* Mouton de Gruyter, Berlin.

Gazdar, G. & C. Mellish (1989), *Natural language processing in prolog: an introduction to computational linguistics.* Addison-Wesley, Workingham.

Gussenhoven, C. (1984), *On the grammar and semantics of sentence accents.* Foris Publ., Dordrecht.

Hirschberg, J. (1990), Accent and discourse context: assigning pitch accent in synthetic speech. In *Proceedings of the IEEE*, 73-11, 1589-1601.

Ladd, D.R. (1978), *The structure of intonational meaning.* Indiana University press, Bloomington.

Quené, H. & A. Dirksen (1990), A comparison of natural, theoretical and automatically derived accentuations of Dutch texts. G. Bailly & C. Benoit (eds.), *Proceedings of the ESCA workshop on speech synthesis, Autrans*, 137-140.

Quené, H. & R. Kager (1992), The derivation of prosody for text- to-speech from prosodic sentence structure. *Computer, speech and language*, 6, 77-98.