

Contrast in concept-to-speech generation

MARIËT THEUNE

*Department of Computer Science,
University of Twente,
P.O. Box 217, 7500 AE Enschede, The Netherlands.
Email: theune@cs.utwente.nl*

Abstract

In concept-to-speech systems, spoken output is generated on the basis of a text that has been produced by the system itself. In such systems, linguistic information from the text generation component may be exploited to achieve a higher prosodic quality of the speech output than can be obtained in a plain text-to-speech system. In this paper we discuss how information from natural language generation can be used to compute prosody in a concept-to-speech system, focusing on the automatic marking of contrastive accents on the basis of information about the preceding discourse. We discuss and compare some formal approaches to this problem and present the results of a small perception experiment that was carried out to test which discourse contexts trigger a preference for contrastive accent, and which do not. Finally, we describe a method for marking contrastive accent in a generic concept-to-speech system called D2S. In D2S, contrastive accent is assigned to generated phrases expressing different aspects of similar events. Unlike in previous approaches, there is no restriction on the kind of entities that may be considered contrastive. This is in line with the observation that, given the ‘right’ context, any two items may stand in contrast to each other.

1. Introduction

To achieve high quality speech output in a spoken language generation system, close attention should be paid not only to acoustic/phonetic aspects of the generated speech, but also to its *prosodic* properties, i.e., the variations in pitch, loudness, tempo and rhythm within an utterance. These variations are determined mainly by accentuation and phrasing. Some of the words in an utterance are emphasised by pronouncing them with a pitch change; these words are said to be accented. In addition, most utterances are divided into (intonational) phrases, the boundaries of which are often marked by a pause, a rise in pitch and the lengthening of pre-boundary speech sounds. In natural speech, the prosody of an utterance varies depending on several factors such as syntactic structure and discourse context. For instance, the strength and placement of prosodic boundaries in an utterance is strongly influenced by syntax. This can be illustrated by example (1) from Sanderman [1996], who showed that PPs serving as adverbial adjuncts (as in (1)a) are generally set off by stronger prosodic boundaries than PPs that serve as nominal adjuncts within an NP (as in (1)b). In the latter case, a prosodic boundary may be absent altogether. Other factors such as phrase length and (to a lesser extent) focus information also play a role in the placement of prosodic boundaries, as was shown by Fitzpatrick [2001] for similar examples.

- (1)a The man hit [the DOG] / with the STICK
 b The man hit [the DOG with the STICK]

In example (1), as in later examples, accented words are indicated by small capital letters. In English and Dutch usually the rightmost word in a phrase is accented, but this default may be changed by contextual factors. For instance, words expressing a concept that has been previously mentioned generally do not receive a pitch accent. An example from Sproat [1995] is (2), where the word *dogs* will often be deaccented because of the earlier reference to the concept ‘dog’.

- (2) My SON wants a DOG, but I am ALLERGIC to dogs.

On the other hand, words expressing information that is contrastive are always accented, even if the information they express has been mentioned previously. A well-known example is (3). Here, both pronouns are accented, even though they refer to entities that have already been mentioned.

- (3) John insulted Mary and then SHE insulted HIM.
 (Lakoff [1971]:333)

We see that different kinds of linguistic information are relevant for establishing the prosody of a generated sentence. In text-to-speech systems, where the input is a text provided by some source outside the system, the required information must be obtained through linguistic analysis of the input text. Such an

analysis may yield unreliable and incomplete results, which has a negative impact on the prosodic quality of the speech output. In concept-to-speech systems, on the other hand, spoken output is generated on the basis of a text that has been produced by the system itself. In such systems, linguistic information from the text generation component may be exploited to achieve a higher prosodic quality than can be obtained in a plain text-to-speech system.

In this paper it is discussed how information from natural language generation can be used for the computation of prosody in a concept-to-speech system. The paper focuses on the automatic marking of ‘contrastive accents’: accents that are used to indicate contrastive information. The question addressed here is how the placement of contrastive accents can be predicted automatically during language generation, based on information about the preceding discourse. The paper does not go into the question how these accents should be realised in speech synthesis, or whether they are phonologically different from other types of pitch accent. (See Krahmer and Swerts [2001] for a discussion of the latter, controversial issue.)

The basic assumption underlying this research is that in English, Dutch and other Germanic languages, accent functions (among other things) as a marker of *information status*. The notion of information status used here is based on Chafe [1976], who distinguishes the statuses newness, givenness, and contrast. Like many others, Chafe observes that words expressing information which is supposed to be *new* to the hearer tend to be accented by the speaker. The opposite holds for words expressing *given* information, i.e., information which is in the hearer’s consciousness, for instance because it was recently mentioned. These words tend to be deaccented. Finally, words expressing *contrastive* information are always accented, even if they refer to something that has been previously mentioned.

Over the years, the distinction between new and given information has received a lot of attention. Various models of givenness have been proposed, including those of Prince [1981], Ariel [1990] and Gundel et al. [1993]. The effect of newness and givenness on accentuation has been empirically investigated by Bock and Mazzella [1983], Brown [1983], Terken and Nootboom [1987], Terken and Hirschberg [1994] and others. Strategies for determining givenness, mostly based on information about word class and previous mention, have been employed to guide accentuation decisions in text-to-speech systems [Hirschberg and Sproat, 1993, Horne and Filipsson, 1997] and in concept-to-speech generation [Davis and Hirschberg, 1988, Monaghan, 1994, Williams, 1998, Nakatani and Chu-Carroll, 2000].

On the other hand, contrast of information has not received much attention in spoken language generation systems, in spite of the fact that the relevance of contrast for accentuation has been pointed out by several researchers, including Halliday [1967], Ladd [1980], Bolinger [1986], Cruttenden [1986], and Pierrehumbert and Hirschberg [1990]. In addition, some formal approaches to the interpretation of contrast have been proposed, of which Rooth [1985, 1992] and Büring [1997] are probably the most well-known. Nevertheless, the accentuation

algorithms in most systems either do not take contrast into account at all, or limit their treatment of contrast to clearly recognizable cases such as corrections or explicitly contrastive constructions (*but ...*). This means that most cases of contrast are ignored, and this can have unwanted consequences. For instance, in speech generation systems that employ a deaccentuation strategy based on previous mention, a failure to detect the contrast in example (3) would cause the pronouns *she* and *he* to be deaccented, resulting in the generation of a quite unnatural intonation pattern.

In short, to achieve appropriate prosody in the output of a spoken language generation system, the presence of contrastive information should not be ignored. In this paper we discuss a few recent, computational approaches to the prediction of contrast, which are potential candidates for implementation in a concept-to-speech system. In addition, we present an alternative, practically oriented approach that has been adopted in a generic concept-to-speech system called D2S. Some of the assumptions underlying contrast prediction in D2S have been tested in a small experiment, which is also described.

The paper is structured as follows. First, Section 2 describes D2S, the concept-to-speech system that formed the practical framework for the research presented here. This provides the background for the remainder of the paper, which deals with contrast prediction in concept-to-speech generation. In Section 3, some computational approaches to contrast prediction are discussed and compared. Section 4 describes a small perception experiment that was carried out to test which discourse contexts trigger a preference for contrastive accent, and which do not. Finally, in Section 5 a practical, but theoretically motivated way of detecting the presence of contrastive information within D2S is presented. We end with an overview of future work and some concluding remarks.

2. Spoken language generation in D2S

D2S is a generic system for the development of concept-to-speech applications. It was originally developed as part of the *Dial Your Disc (DYD)* system, where it was used to generate English spoken monologues about Mozart compositions [van Deemter et al., 1994, Odijk, 1995, van Deemter and Odijk, 1997]. Since then, D2S has been used for spoken language generation in various systems including GoalGetter, a system that generates Dutch spoken football reports [Theune et al., 2001], and OVIS, a Dutch spoken dialogue system providing train information [Veldhuijzen van Zanten, 1998, van Noord et al., 1999, Klabbers, 2000]. In this section, examples from GoalGetter are used to illustrate D2S.

The general architecture of D2S is shown in Figure 1. The language generation module (LGM) of D2S takes data as input and produces *enriched text*, i.e., text which has been annotated with prosodic markers indicating the placement of accents and phrase boundaries. The enriched text forms the input for the speech generation module, which turns it into a speech signal. Below, the components of D2S are discussed in turn, focusing on the close link between them. Section 2.1

discusses text generation in D2S, Section 2.2 describes how the Prosody module, which is embedded in the LGM, places prosodic markers in the generated text, and finally, Section 2.3 sketches how speech generation is performed in D2S.

(Figure 1 approximately here.)

2.1. Language generation in D2S

The input to a D2S system typically consists of (tabular) data, which may be retrieved from a database or some other information source. Figure 2 shows an example input to the GoalGetter system: a table with information about a football match, taken from a teletext page. The table, shown here in English translation, specifies (i) the names of the two teams, (ii) the names of all players – listed below their team – who have scored a goal, followed by the minute in which this happened, (iii) the name of the referee, (iv) the number of spectators, and (v) a list of players who committed some offense and consequently received a yellow or a red card. In addition to these variable data, the LGM also uses a static domain knowledge base containing background information about the teams (home town and stadium name) and the players (position and nationality).

(Figure 2 approximately here.)

(Figure 3 approximately here.)

The first step of the D2S language generation module (from now on referred to as LGM) is to convert the system’s input data into a typed data structure that is suitable as a basis for generation. The LGM data structure corresponding to Figure 2 is shown in Figure 3. To create a natural language text, the LGM attempts to express all elements of the data structure using a collection of so-called *syntactic templates*, which contain syntactic tree structures with open slots for variable information. Figure 4 shows an example template from GoalGetter, which can be used to describe the scoring of a goal. Formally, a syntactic template σ is a quadruple $\langle S, E, C, T \rangle$, where S is a syntactic tree (typically for a sentence) with open slots in it, E is a set of links to additional syntactic structures which may be substituted in the gaps of S , C is a (possibly complex) condition on the applicability of σ and a T is a set of topics.

(Figure 4 approximately here.)

The syntactic trees S in the templates bear a certain resemblance to the initial trees of Tree Adjoining Grammar (TAG, [Joshi, 1987]). A difference with TAG trees is that the latter are generally ‘minimal’, i.e., only the head of the construction is lexicalised and the gaps coincide with the arguments of the head, whereas the syntactic trees in the LGM templates may contain more words. Often, this is done in order to express collocations, i.e., groups of words with a frozen meaning. In TAG, these are put into a common tree as well [Abeillé and Schabes, 1989]. Examples of collocations occurring in the GoalGetter templates are *een doelpunt*

laten aantekenen (“have a goal noted”) (as in Template Sent16) and *de leiding nemen* (“take the lead”). The syntactic information in the templates is used for prosody computation (see Section 2.2), and for checking certain grammatical conditions on sentences.

The second element of a syntactic template is E: *the slot fillers*. Each open slot in the tree S is associated with a call of a so-called **Express** function, which generates the set of possible slot fillers for the given gap. Typically, there are several possible slot fillings for each slot in a template. For instance, a person may be referred to using a proper name, a definite description or a pronoun.

The third ingredient is C: *the condition*. A template σ is applicable if and only if its associated condition is true. There are two kinds of conditions: (i) Knowledge State conditions and (ii) linguistic conditions. The Knowledge State records which parts of the input data structure have been expressed (these are assumed to be *known* to the user) and which parts have not (these are assumed to be *unknown*). Knowledge State conditions are used to restrict the ordering of information, stating things like “ X should not be conveyed to the user before Y has been conveyed”. The first two (sub)conditions of Template Sent16 are of this kind. They state that the template can only be used if the competing teams have been introduced to the user (i.e., are known) and the current goal is the first one which has not been conveyed (is unknown). ‘Linguistic’ conditions are related to the semantics/pragmatics of the sentence that can be generated from the template, restricting the kind of input data to which the template can be applied. The two final conditions on Template Sent16 are of this type. The first says that Sent16 is only applicable if the player of the current goal has scored more than once during the match; the second states that this syntactic template cannot be used to describe an own goal.

Finally, each syntactic template is associated with a *topic* T . This is a label that globally describes what the template is about. Topics are used to ensure a natural grouping of the generated sentences. Each paragraph in the generated text contains only sentences that have been generated from syntactic templates sharing the same topic.

The ordering of paragraphs in a generated text and sentences in a paragraph is determined by the conditions on the syntactic templates. A template can be used if it is associated with the topic of the current paragraph, and if its conditions evaluate to true given the current Knowledge State. If more than one template is applicable, one is chosen arbitrarily. After a sentence has been generated from the chosen template, the Knowledge State is updated and new templates become applicable. If there are no more applicable templates with the current topic, a new topic must be chosen. Whether a new paragraph can be started given a topic T depends on the applicability of the templates associated with that topic. If there are no templates associated with T whose conditions evaluate to true in the current Knowledge State, T is skipped until the Knowledge State has been sufficiently changed for some of its templates to be applicable. When a syntactic template has been selected for use, its variable slots are filled with

appropriate expressions. This is done relative to the *Context State*, which keeps track of (among other things) the discourse objects that have been mentioned. This information is relevant for the generation of referring expressions (e.g., the use of pronouns) and for prosody computation, as will be shown below. For a detailed discussion of the LGM's generation algorithm, see Theune [2000] and Theune et al. [2001].

To illustrate the generation process, we discuss the generation of (a part of) the example text shown in Figure 5. This is a text generated by GoalGetter to express the information in Figure 2. For ease of exposition, we only show the English translation of the original text. It should be noted, however, that the sixth sentence was originally based on Template Sent16 from Figure 4. The generation of this sentence is discussed in some detail below. First we briefly sketch the generation of the text up to the example sentence.

(Figure 5 approximately here.)

At the start of the generation process, the Context State is empty and the Knowledge State shows that all available information is still 'unknown' to the user. In this situation, the only GoalGetter topic with any applicable syntactic templates is the 'general' topic, which is associated with templates that express global match information, for instance introducing the competing teams. The sentences in the first paragraph of Figure 5 all belong to this topic; we do not discuss their generation here.

When the 'general' topic has been exhausted, a new paragraph is started. This time, the system picks the 'game_course' topic. In the current Knowledge State, several syntactic templates associated with that topic are applicable: these are templates describing the first goal of the match. One of them is picked at random and used to generate the fourth sentence of Figure 5: *The team from Sittard took the lead after seventeen minutes through a goal by Hamming*. Consequently, the Knowledge State is updated to reflect the fact that information about the first goal has been conveyed. In addition, the Context State is extended with entities corresponding to the phrases *the team from Sittard*, *Hamming* and *after seventeen minutes*. Now the system goes on and attempts to convey the second goal scoring event. It cannot use the same template as the one used for the fourth sentence, since the second goal scoring event of the current match does not make one team take the lead. Instead, a template is used that is applicable if the scores of two teams are equalised: *One minute later Schenning from Go Ahead Eagles equalised the score*.

Now a sentence must be generated to describe the last goal of the match. To do this, Template Sent16 (shown in Figure 4) is selected. As the reader can verify, all conditions associated with this syntactic template are met. After the template has been selected, the system creates the set of all trees that can be generated from it, using all possible combinations of slot fillers generated by the associated **Express** functions. These slot fillers are shown in Figure 6. Details on their generation can be found in Theune [2000] and Theune et al. [2001].

(Figure 6 approximately here.)

Combining all different slot fillings returns a set of 16 trees that can be generated from Template Sent16 in the current context. A test on syntactic well-formedness filters out the trees where the proper name *Hamming's* occupies the <playergen> slot, because these violate Principle C of the Binding Theory [Chomsky, 1981]. Next, a check on the proper use of anaphors filters out the trees where either the pronoun *he* or the definite description *the forward* occupies the <player> slot. Both expressions require an accessible antecedent, which is not available in the current discourse: the antecedent (*Hamming* in sentence four) is not accessible due to the intervening reference to the player Schenning. This means that only trees for the following sentences remain:

$$\left\{ \begin{array}{l} \textit{After forty-eight minutes Hamming had his second goal noted,} \\ \textit{After forty-eight minutes the forward Hamming had his second goal noted} \\ \textit{In the forty-eighth minute Hamming had his second goal noted,} \\ \textit{In the forty-eighth minute the forward Hamming had his second goal noted} \end{array} \right\}$$

From these trees, one is selected arbitrarily (in this case the second one), and sent to the Prosody module, where its prosodic properties are computed. This is discussed in Section 2.2. The fringe of the resulting tree, i.e., the sentence enriched with prosodic markers, is sent to speech generation to be pronounced, and the Knowledge State and the Context State are updated accordingly. This ends our illustration of the language generation process; more details can be found in Theune [2000] and Theune et al. [2001].

2.2. Prosody computation in D2S

The Prosody module of the LGM automatically determines the location of accents and phrase boundaries in a generated sentence, using both syntactic and discourse information. The prosodic rules that are used are independent of domain and language, within the class of Germanic languages (e.g., English, Dutch, and German). We illustrate them using our example sentence “After forty-eight minutes the forward Hamming had his second goal noted”, now in its original Dutch version: *Na achtenveertig minuten liet de aanvaller Hamming zijn tweede doelpunt aantekenen.*

The Prosody module computes the accentuation pattern of an incoming sentence tree using a version of Focus-Accent Theory [Baart, 1987] proposed by Dirksen [1992] and Dirksen and Quené [1993]. In Focus-Accent Theory, binary branching metrical trees represent the semantic and syntactic prominence of nodes with respect to pitch accent. In D2S, the metrical tree of a sentence is constructed by converting its syntactic tree to a tree that is at most binary-branching and then marking its nodes with *focus* markers and *weak* or *strong* labels. Informally speaking, the focus markers indicate which words and phrases should or should not receive an accent, while the w/s labels determine where in each constituent an accent may land.

The focus properties of the nodes in the metrical tree are determined as follows. First, the Prosody module adds a preliminary focus marking to the tree: all maximal projections of the form XP are assumed to be in focus and are marked [+F]; the other nodes are unspecified for focus. After the initial, default assignment of focus markers has taken place, the system tries to determine the *information status* of the words or phrases in the tree. The notion of information status used in D2S is based on Chafe [1976], who distinguishes newness, givenness, and contrast, as explained in Section 1. To determine information status, the D2S Prosody module first checks which words and phrases in the current sentence are *contrastive*. For a detailed discussion of how this is done, we refer to Section 5. Here, it suffices to say that in our example sentence, two contrastive phrases are detected: *na achtenveertig minuten* (“after forty-eight minutes”) and *de aanvaller Hamming* (“the forward Hamming”). These phrases are marked [+C] in the metrical tree, to indicate that they are in focus due to contrast. If a constituent is marked for contrast its focus marking cannot be changed. Next, the system uses information from the Context State to determine which words or phrases in the tree express *given* information. The focus value of their dominating node is changed to [-F], except if the node has a [+C] marking. The rules for determining givenness are based on van Deemter [1994b], who, like Chafe [1976], distinguishes *object-givenness* and *concept-givenness*. A word or phrase is object-given if it refers to a discourse entity (e.g., a player) that has already been mentioned, and it is concept-given if it expresses a concept (e.g., ‘scoring a goal’) which has been evoked earlier in the discourse. Following Davis and Hirschberg [1988] and Hirschberg and Sproat [1993], items are assumed to remain given within one discourse segment; in the case of D2S this corresponds to a paragraph. In the example sentence, the NPs *de aanvaller Hamming* (“the forward”) and *zijn* (“his”) are object-given, because their referent, the player Hamming, was referred to two sentences earlier (see Figure 5). The focus marking of the second NP (*zijn*) is therefore changed to [-F], but the marking of the first NP does not change because it has been marked for contrast. The example sentence contains two cases of concept-givenness. The word *minuten* (“minutes”) is defocused because the time concept it expresses has already been mentioned in the two previous sentences. The collocation *een doelpunt laten aantekenen* (“having a goal noted”) also refers to a concept that has been previously expressed: the concept of goal scoring. The words *doelpunt*, *liet* and *aantekenen* are therefore marked as [-F].

Finally, the weak/strong labelling is applied to the metrical tree nodes, based on the structure of the tree and the focus properties of its nodes. In Dutch, like in English, normally the left node of two sisters is weak and the right node is strong. However, if the structurally strong node is marked [-F] while the structurally weak node is not, the weak/strong labelling is switched due to the so-called Default Accent rule [Ladd, 1980], [Baart, 1987]. In Figure 7, showing the complete metrical tree of the example sentence, this has occurred in three cases: (i) for the AP *achtenveertig* and the defocused N⁰ *minuten*, (ii) for the AP

tweede and the defocused N^0 *doelpunt*, and (iii) for the NP *zijn tweede doelpunt* and the defocused V^0 *aantekenen*.

(Figure 7 approximately here.)

When the metrical tree is complete, the focus markers indicate which constituents should be accented, and the weak/strong labelling indicates on which words the accent may land. The actual accentuation algorithm can therefore be very simple: each node that is marked [+F] or [+C] launches an accent, which trickles down the tree along a path of strong nodes until it lands on a terminal node, dominating a word. In the example sentence, the accents launched by CP, IP and VP all coincide with the accent launched by the NP node of *zijn tweede doelpunt*, finally landing on the word *tweede*, and not on the defocused N^0 *doelpunt*. Since the nodes dominating *liet* and *aantekenen* are weak, no accent trickles down to them, and because they are marked [-F] they do not launch an accent themselves. The PP node dominating the phrase *na achtenveertig minuten* does launch an accent, which trickles down to the NP *achtenveertig minuten*, where it coincides with the accent launched by the NP itself. Within the NP, the right node dominating *minuten* has been defocused, so the accent goes leftward and lands on the word *achtenveertig*. Finally, the appositive, contrastive NP *de aanvaller Hamming* consists of two NPs (not shown in the tree due to space limitations), both of which launch an accent that trickles down to their head nouns.

It should be noted that at its final level, the accentuation algorithm no longer makes any distinction between accents arising from newness ([+F]) or contrast ([+C]). The main, practical reason for this is that the speech synthesis system employed in D2S (discussed in Section 2.3) does not make a distinction between contrastive and newness accents during the assignment of intonation contours. Since accent type is not taken into account by speech synthesis, information on this is currently not included in the output of language generation. However, it would be very easy to change this if the LGM were to be combined with a speech synthesizer that does associate contrastive and newness accents with different tunes, as distinguished by Pierrehumbert and Hirschberg [1990] and others.

After accentuation, phrase boundaries are assigned. Currently, three phrase boundary strengths are distinguished. The strongest is the *sentence-final* boundary (///). Next comes the *major* boundary (//), which follows words preceding a punctuation symbol other than a comma (e.g., “;”) and sentence-internal clauses (i.e., a CP or IP within a sentence). Finally, a *minor* boundary (/) follows words preceding a comma and constituents meeting the following conditions: (i) the constituent has sufficient length (more than four syllables), (ii) the constituent on its right is an \bar{I} , a \bar{C} or a maximal projection, and (iii) both constituents contain at least one accented word. This is a slightly modified version of a structural rule proposed by Dirksen and Quené [1993]. In the present example only the PP *na achtenveertig minuten* and the NP *de aanvaller Hamming* meet this

condition and are therefore followed by a minor phrase boundary. Since the example sentence contains no punctuation and consists of just one clause, the only other phrase boundary is the sentence-final one.

The accentuation algorithm of D2S was formally evaluated for Dutch in a small-scale experiment [Nachtegaal, 1997]. Non-professional speakers of Dutch read aloud the plain text versions of texts generated by GoalGetter, and ‘expert listeners’ indicated on which words they heard an accent. The accentuation patterns produced by the speakers were then compared to those generated by the system, which showed that the number of words on which the accentuation by GoalGetter deviated from the accentuation by the speakers was very small. The general prosodic quality of the output of D2S has been informally compared with that of two Dutch text-to-speech systems, one of which employs the same speech synthesis as used in GoalGetter. The two systems were used to pronounce some texts generated by the GoalGetter system (plain text version), and the prosodic quality of their speech output was compared to that produced by GoalGetter. This revealed two main flaws displayed by both text-to-speech systems. First, the placement of phrase boundaries by the two text-to-speech systems was less adequate than in D2S: several obvious phrase boundaries were missing (e.g., between conjugated clauses), or misplaced (e.g., between an adjective and the NP it modified). Second, both systems failed to perform deaccentuation even in simple cases involving the literal repetition of a word. These flaws made the output of the text-to-speech systems sound much less natural than that of GoalGetter.

2.3. Speech generation in D2S

The D2S system currently has two different speech output modes available: phonetics-to-speech synthesis and phrase concatenation. In phonetics-to-speech mode, the enriched text output from the LGM is first converted to a phonetic transcription. Because the LGM generates an orthographic representation with a unique phonetic representation, it is possible to do errorless grapheme-to-phoneme conversion by lexical lookup instead of rules. The resulting phonetic transcription is fed into the Calipso speech synthesis system [Gigi and Vogten, 1997, Klabbers, 2000], which generates speech output by concatenating diphones: small speech segments consisting of the transition between two adjacent phonemes. The intonation rules in Calipso are based on ’t Hart et al. [1990], who describe the intonation of an utterance in terms of pitch movements. Calipso assigns (combinations of) pitch movements on the basis of combinations of accents and boundaries. In D2S, information about these is given in the enriched text that is input to the speech synthesizer. In two anonymous tests concerning subjective evaluation under telephone conditions, Calipso was judged favourably on several aspects, including general quality, intelligibility and voice pleasantness [Rietveld et al., 1997, Sluiter et al., 1998].

Although diphone synthesis offers unlimited flexibility, its naturalness still leaves a great deal to be desired. In contrast, the concatenation of pre-recorded

phrases offers a speech quality that is close to that of natural speech. D2S uses an advanced phrase concatenation technique, where phrases are recorded in different prosodic versions [Klabbers, 2000]. The use of several prosodic variants is aimed in particular at the slots in the syntactic templates used by the LGM. The fixed parts of the syntactic templates can usually be recorded as a whole, and in only one version. The prosody of the slots in the templates however, is most crucial, because there the variable (and usually most important) information is inserted. These slot fillers were recorded in six prosodically different versions, one for each context described in terms of accent and phrase boundary markers. During speech generation in D2S, the appropriate versions are selected on the basis of the markings in the enriched text. Stylistations of the different prosodic realisations are depicted in Figure 8 and are briefly explained below.

(Figure 8 approximately here.)

1. An accented slot filler which does not occur before a phrase boundary is produced with a so-called (pointed) *hat pattern*, consisting of a rise and fall on the same syllable. This contour corresponds to the prosodically neutral version used in many other phrase concatenation techniques.
2. An accented slot filler occurring before a minor or a major phrase boundary is most often produced with a rise to mark the accent and an additional continuation rise to signal that there is a non-final boundary. It is followed by a pause of either 200 ms (minor boundary) or 300 ms (major boundary).
3. An accented slot filler occurring in final position receives a final fall and is followed by a pause of 500 ms.
4. Unaccented slot fillers are pronounced on the declination line without any pitch movement associated with them.
5. Unaccented slot fillers occurring before a minor or a major phrase boundary only receive a small continuation rise. Again, a 200-ms or 300-ms pause is inserted.
6. Unaccented slot fillers in a final position are produced with final lowering, i.e., a declination slope that is steeper than in other parts of the utterance. They are followed by a 500-ms pause.

The phrase concatenation method used in D2S has been evaluated in a formal listening experiment [Klabbers, 2000], in which it was compared to (i) natural speech output, (ii) a conventional concatenation approach, where words and phrases are recorded in one version only (as often used in commercial applications), and (iii) Calipso's diphone synthesis. The results show that the advanced form of phrase concatenation used in D2S compares well to natural speech on both intelligibility and fluency, and scores very well on overall quality and suitability. The conventional concatenation approach scores significantly less on all dimensions, indicating that it sounds less natural than is sometimes assumed

[Sluijter et al., 1998]. The evaluation results indicate that it is worth the extra effort to take a prosodically sophisticated approach to phrase concatenation. (Of course, this is useful only if, as in D2S, prosodic markers are available to guide phrase selection.) Diphone synthesis scores worst on all dimensions. However, in applications where the vocabulary is large, or changes frequently, phrase concatenation is infeasible and speech synthesis is the only option available.

2.4. Discussion

In the generic data-to-speech system D2S there is a tight coupling between the language generation and the speech generation modules. Language generation is done using a hybrid technique where the use of syntactically enriched templates is constrained by local conditions on the linguistic context, while for speech generation pre-recorded phrases are combined in a sophisticated manner, taking prosodic variations into account. Speech generation can also be achieved by phonetics-to-speech synthesis, which offers greater flexibility but a less natural speech quality. The coupling between the language and speech generation modules is brought about by the computation of prosody by the LGM. This ensures that the syntactic, semantic and discourse knowledge captured in the LGM can be used in speech generation, without requiring linguistic analysis of the generated texts. This makes it possible to achieve a better prosodic quality of the system's output than could be obtained by simply feeding the outcome of language generation into a text-to-speech system.

D2S is a practically useful system which can serve as a basis for a wide range of applications. Developing a new application mainly involves constructing a set of syntactic templates, designing a structure representing the input data and, optionally, adding a database with domain knowledge; all other parts of the LGM are application independent. The work to be done on speech generation depends on the chosen output mode. If phonetics-to-speech is chosen, an off-the-shelf speech synthesis program may be used and only an application-specific lexicon for grapheme-to-phoneme conversion has to be made. The use of phrase concatenation gives rise to more work, but this is compensated by a more natural sounding speech output [Klabbers, 2000].

Although D2S is presented here as one integrated system, the techniques it employs can also be used independently. A variant of the phrase concatenation method used in D2S has been employed in a new version of the German train information system described by Aust et al. [1995], while the LGM of D2S has been used for the generation of English and German route descriptions in VODIS, a European project aimed at the development of a speech interface for a car navigation system [Krahmer et al., 1997, Pouteau and Arévalo, 1998].

3. Some approaches to the prediction of contrast

In order to generate acceptable accentuation patterns, a spoken language generation system should be able to determine the information status (new or given,

contrastive or not contrastive) of the words and phrases in its output. So far, most research on information status has focused on the distinction between given and new information. However, a spoken language generation system should be able to distinguish contrastive information as well, because ignoring contrast may lead to improper deaccentuation of given but contrastive items. Only relatively recently, some approaches to the prediction of contrast in concept-to-speech generation have been put forward. These are discussed in the current section.

3.1. Prevost: sets of alternatives

The theory of contrast proposed by Prevost [1995] was inspired by the ‘alternative semantics’ of Rooth [1985, 1992]. In Prevost’s approach, the determination of contrast is based on the semantic representations of the generated sentences. Two propositions count as contrastive if they contain either two contrasting pairs of discourse entities or only one contrasting pair of discourse entities plus contrasting ‘functors’, e.g., verbs. A discourse entity x is contrastive if the preceding discourse contains a reference to another entity belonging to the ‘set of alternatives’ of x , which is a set of different entities of the same type as x . Of contrastive functors, no definition is given, but the verbs *love* and *hate* are given as an example. Prevost implemented his theory in two small generation systems, one of which can produce the responses to database queries, while the other produces monologues describing stereo systems and their components. A sequence of two contrastive utterances, generated by the monologue system is shown in (4). It should be noted that in the original example by Prevost, these utterances are not adjacent; they are the first sentences of two subsequent paragraphs describing the two amplifiers. The example is shortened here for expository reasons. In general, the discussion of contrast in this paper is restricted to cases of contrast between two subsequent sentences.

- (4) The X4 is a SOLID-state AMPLIFIER. The X5 is a TUBE amplifier.
(Prevost [1995]:142-143)

The generation of the second utterance, in the context of the first, is discussed in some detail by Prevost. In the utterance, two discourse entities are referred to: **e2**, an amplifier (expressed using the name $x5$), and **c2**, which is a specific *class* of amplifiers (expressed using the description *a tube amplifier*). These two entities are considered to stand in contrastive relationships to the entities referred to in the first utterance: **e1**, a different amplifier, and **c1**, which is a different *class* of amplifiers. This means that the requirement of two contrasting pairs of discourse entities is met. Once the contrastive entities are located, Prevost’s algorithm for contrastive accent assignment determines which of these objects’ properties are contrastive. These are the properties that help to distinguish an object from its alternatives that have been mentioned in the context. So, **c2**’s having a *tube* design is a contrastive property (because its alternative **c1** has a different design),

but its being an amplifier is not (because *c1* is also an amplifier). Consequently, the modifier *tube* receives contrastive accent, and the noun *amplifier* does not.

An important problem of Prevost’s approach, as Prevost himself notes, is that it is very difficult to define exactly which items count as being of the same type, and thus as contrastive. If the definition of ‘same type’ is too strict, not all cases of contrast will be accounted for. On the other hand, if it is too broad, then anything will be predicted to contrast with anything. Prevost discusses the following problematic example:

- (5) Mary took John to a hockey game for his birthday, but he didn’t seem very pleased. While HE intently watched the CLOCK, SHE watched the GAME.
(Prevost [1995]:147)

This is a clear case of contrast, but it does not seem appropriate to regard *clock* and *game* as being alternatives, since they are not obviously of the same type. Counting them as alternatives would mean an unwanted broadening of the notion of ‘set of alternatives’: such a set could then contain almost anything. Prevost has adopted a practical solution in the spoken language generation system he developed: in this system, alternative sets are fixed in the knowledge base, where such sets are inferred from **isa** links that define class hierarchies [Prevost, 1996]. Only entities with the same parent or grandparent class are considered alternatives. Thus, the system is unable to produce contrastive accent in examples like (5).

3.2. Pulman and others: HOU and parallelism

Another approach to the generation of contrastive accent is advocated by Pulman [1997], who proposes to use Higher Order Unification (HOU) for the interpretation and prediction of contrastive accent (as well as other cases of ‘narrow focus’). Higher Order Unification is a method for solving equations through substitution, where the sides of the equations are terms of higher order logic. Pulman makes use of equivalences like the one in (6), which can be used for both interpretation and prediction of contrast, and which operate at the level of semantic representation (or, more specifically, quasi-logical form or QLF; see Alshawi and Crouch [1992]).

- (6) $\text{assert}(F,S) \Leftrightarrow S$
if
 $B(F) = S$
& $\text{context}(C)$
& $P(A) = C$
& $\text{parallel}(B \bullet F, P \bullet A)$

In (6), *S* corresponds to the QLF of the target sentence (i.e., the sentence for which the focused parts must be found); *F* corresponds to the focused part of

S; B is the background part of S (i.e., the result of abstracting over F in S); C is the QLF of a ‘salient’ utterance in the context of the target sentence; and P and A are the parts of C that are parallel to B and F respectively. Pulman does not give an exact definition of parallelism, but states that “to be parallel, two items need to be at least of the same type and have the same sortal properties” (Pulman [1997]:90). This is similar to, but less specific than, Prevost’s characterisation of the elements of alternative sets, discussed in Section 3.1. Informally, the equivalence in (6) says that asserting QLF S with focus on F is equivalent to S if in its context an utterance can be found with QLF C, where C contains an item A that is parallel to F, while the background P of C is parallel to the background B of S. Using HOU, this equivalence can be resolved to predict the placement of focus markers in a generated sentence. As an illustration, assume that the focus of the second sentence in (7) must be computed.

- (7)A: John kissed Sue.
 B: (No,) John kissed Mary.

Given the sentences in (7), we know that S is equal to $kiss(j,m)$, being the semantic representation of the second sentence, and that C is equal to $kiss(j,s)$, the semantic representation of the preceding one. In order to determine F, the equation in (8) needs to be resolved.

$$(8) \quad \begin{aligned} B(F) &= S = kiss(j,m) \\ P(A) &= C = kiss(j,s) \\ \text{where} \\ \text{parallel}(B \bullet F, P \bullet A) \end{aligned}$$

Using HOU, the following solution can be found: $F = \lambda P.P(m)$ and $A = \lambda Q.Q(s)$, with $B = P = \lambda O.O(\lambda x.kiss(j,x))$. Informally, this means that Mary is the focus (i.e., contrastive element) of the second sentence, and that the background of both sentences is that John kissed someone. Note that when the ‘parallel’ operator takes several arguments, as in (6), at least one parallel pair is required to be distinct; the other(s) may be identical.

The main problem of Pulman’s approach is the same as that of Prevost’s theory, namely that of defining when two items are ‘of the same type’. In fact, Pulman consciously avoids giving a definition of this “notoriously slippery notion” (Pulman [1997]:93). However, Gardent and Kohlhase [1997] do propose a HOU-based way of determining which are the parallel elements in parallel constructions. They combine HOU with an abductive calculus using sorted type theory [Kohlhase, 1994] to model ‘contrastive parallelism’ or in short *c-parallelism*. In sorted type theory, objects of a certain logical type are subdivided in terms of sorts (which can be seen as unary predicates), ordered by a partial ordering relation ($=<$) in a sort hierarchy. According to Gardent and Kohlhase [1997], objects are considered as c-parallel (and thus as belonging to each other’s set of alternatives, in Prevost’s terms) if they are both *similar* (have a sort in common)

and *contrastive* (have a *distinguishing* (complementary) sort – for instance, one is ‘animate’ whereas the other is ‘inanimate’). These ideas on what it takes for items to be c-parallel are largely similar to those of Prevost [1995] concerning alternatives. As a consequence, the approach advocated by Gardent and Kohlhase seems equally unsuitable to deal with examples like (5), which proved problematic for Prevost.

3.3. Van Deemter: contrariety and equivalence

The theory of contrast put forward by van Deemter [1994a, 1998, 1999] is based on the determination of contrariety and equivalence. Van Deemter states that many cases of contrast can be attributed to parallelism (see e.g., Prüst [1992], Hobbs and Kehler [1997], Asher et al. [2001]), but he points out that there are also many examples of contrast which lack parallelism. Van Deemter uses the notion of *contrariety* to account for these cases. Informally defined, two sentences (or clauses) are contrary to each other if they cannot be true at the same time. According to van Deemter, two sentences stand in a contrast relationship if they contain two items (one in each sentence) which are ‘contrastible’ and whose substitution by the same constant will cause the two sentences to be contrary to each other. A contrastive relationship between two sentences triggers a contrastive accent in each sentence. The landing place of these accents depends on the location of the substituted items. If they are located in the subject of the sentence, the accent will land in the subject; if they are located in the predicate, the accent will land in the predicate. (The two items need not be located in corresponding parts of the two sentences.) Van Deemter gives (9) as an example. If we assume that being an organ mechanic implies knowing much about organs, then replacing Mozart by Bach produces a contrariety. This correctly predicts a contrastive accent on Bach and Mozart.

- (9) BACH was an organ mechanic; MOZART knew little about organs.
 After substitution:
 Bach was an organ mechanic; Bach knew little about organs.
 (van Deemter [1999]:11)

Two sentences also count as contrastive if they contain two contrastible items which, after replacement by the same constant, cause the sentences to be logically *equivalent* [van Deemter, 1999]. This is shown in (10).

- (10) SEVEN is a prime number and so is THIRTEEN.
 After substitution:
 Seven is a prime number and so is seven.
 (van Deemter [1999]:12)

Unlike Prevost’s and Pulman’s, van Deemter’s definition of ‘contrastible items’ is extremely permissive: the only constraint on contrastibility is inequality of

denotations. This permissiveness makes it necessary to restrict the number of allowed substitutions to one, because otherwise far too many cases of contrastive stress would be predicted: any pair of sentences of the form $(NP_1 VP_1)$, $(NP_2 \text{ Negation } VP_2)$ or $(NP_1 VP_1)$, $(NP_2 VP_2)$ would then count as contrastive, since substituting both the NPs and the VPs by a constant would lead to a contrariety or equivalence. This problem is discussed in van Deemter [1994a]. Unfortunately, many examples of contrast can only be explained in terms of contrariety if at least two pairs of items are substituted. For instance, the contrast in Prevost's example (5) (*While HE intently watched the CLOCK, SHE watched the GAME*) could be predicted easily by replacing the pairs {he, she} and {clock, game} with the same constant, which would result in an equivalence. Replacing only one pair of contrastible items does not give the desired results. The best prediction would be made by replacing the pair {he, she} by a constant, because if we assume that a person cannot watch two things at the same time, this would result in a contrariety (e.g., *While John₁ intently watched the clock, he₁ watched the game*). This solution is not entirely satisfactory, since it only predicts contrastive accent in the subject part of the contrastive sentences, which is where *he* and *she* are located. It does no justice to the observation that (5) contains *two* contrastive pairs, located both in the subject *and* the predicate part of the containing sentences: {he, she} and {clock, game}. Not counting *clock* and *game* as contrastive leaves open the possibility of deaccenting them in a context where they are given. Another problematic example, based on Prevost [1995], is (11). Here an equivalence can only be reached by substituting constants for the pairs {British, American} and {Stereofool, Audiofad}. Replacement of only one of these pairs does not result in an equivalence or contrariety, as shown below.

- (11) The BRITISH amplifier was praised by STEREOFUOL, an audio journal. The AMERICAN amplifier was praised by AUDIOFAD, another audio journal.

After substitution of two items (not allowed):

The British amplifier was praised by Stereofool.

The British amplifier was praised by Stereofool.

After substitution of only one item:

The British amplifier was praised by Stereofool.

The American amplifier was praised by Stereofool.

or:

The British amplifier was praised by Stereofool.

The British amplifier was praised by Audiofad.

In principle, the contrasts in the above examples may be attributed to syntactic parallelism. However, for most examples of this kind it is possible to come up with re-phrasings that do not show any parallelism, e.g., (12), and which therefore still lack an explanation.

- (12) STEREOFOL, an audio journal, printed a favourable review of the BRITISH amplifier. The AMERICAN amplifier was praised by AUDIOFAD, another audio journal.

3.4. Discussion

All approaches to contrast prediction discussed in the previous sections have in common that they regard the presence of a pair of ‘alternative items’ (called parallel, contrastive, or contrastible items) as a prerequisite for contrast. An important difference between the approaches is that both Prevost [1995] and Pulman [1997] claim that two alternatives should be at least ‘of the same type’, whereas in the theory of van Deemter [1994a, 1998, 1999], the only condition on contrastible items is inequality of denotations. A possible explanation for this difference is the following. Prevost’s approach is based on the alternative set semantics of Rooth [1985, 1992], who presents a unitary analysis of focus that includes not only contrast, but also other focus phenomena such as the association with focus of various adverbs (*only, even, too . . .*). The latter cases seem to require at least some constraints on alternative sets for a successful interpretation. Pulman also works in this tradition, and aims to cover largely the same cases as Rooth. Van Deemter’s approach, however, is aimed solely at the prediction of contrastive accent.

In addition to the presence of a pair of alternatives, all discussed approaches impose a further condition that must be met for two sentences to count as contrastive: Prevost [1995] requires the additional presence of a second pair of contrastive objects, or of a pair of contrastive functors; Pulman [1997] requires the ‘backgrounds’ of the two sentences to be parallel (where the backgrounds are the results of abstracting over the alternative items); and van Deemter [1998, 1999] requires that substitution of the two contrastible items by a constant results in a contrariety or equivalence.

(Table I approximately here.)

None of the three approaches correctly predicts all examples of contrast that have been presented in this section. These examples, and the corresponding predictions of Prevost, Pulman, and van Deemter are listed in Table I. The first example, (4) from Prevost, is a clear-cut case of contrast and can be handled by all three approaches. The sentences contain two pairs of alternatives, meeting the contrast condition of Prevost, and they have a parallel background, satisfying Pulman’s constraint. Van Deemter also predicts a contrast here, because replacing x_4 and x_5 with the same constant results in a contrariety. Note, however, that contrariety only predicts a contrastive accent on x_5 , and not on the equally contrastive *tube* (which is only accented due to newness in van Deemter’s approach).

As was discussed in Sections 3.1 and 3.2, in the problematic example (5) contrast is predicted by neither Prevost nor Pulman, because in their approaches

‘clock’ and ‘game’ are not counted as alternative items. (In Prevost’s case, this means that the sentences do not contain the two pairs of alternatives required for contrast, and in Pulman’s case, that the backgrounds are not parallel.) As we have seen in Section 3.3, van Deemter’s contrariety approach does predict the presence of contrast, but only between *he* and *she*.

Example (7) is cited by Pulman, who shows how his approach predicts the placement of contrastive accent in this example. At first sight, it appears that Prevost cannot predict contrast in this case, because there is only one pair of alternatives: {Sue, Mary}. However, it is possible that Prevost would regard the (explicit or implicit) *No* in the answer as a contrastive functor, in which case contrast would be predicted after all. (As mentioned in Section 3.1, Prevost provides little information as to what constitutes a contrastive functor.) Finally, van Deemter correctly predicts the contrast between *Sue* and *Mary*, because replacing these two items by the same constant gives rise to an equivalence.

The contrast between *Bach* and *Mozart* in example (9) from van Deemter is not predicted by the other two approaches. The two sentences only contain one pair of alternatives, {Bach, Mozart}, and their backgrounds are not parallel. Of course, van Deemter’s approach does successfully predict the contrast in his own example. A marginal note here is that van Deemter does not predict a contrastive accent on the adjective *little*, in spite of its apparent contrast to the implied *much* in the preceding sentence. (Being an organ mechanic is assumed to imply knowing *much* about organs, which is what causes the contrariety in the first place.)

Finally, in example (12) contrast is predicted only by Prevost, based on the presence of two pairs of alternatives (the two amplifiers and the two magazines). Pulman’s approach fails here, because the backgrounds of the sentences are not parallel. As discussed in Section 3.3, van Deemter cannot explain this case because the substitution of two items does not result in a contrariety or equivalence, and an alternative explanation in terms of parallelism is not available either.

On the whole, the approach of van Deemter gives the best results for the examples discussed here, in particular those that do not involve obvious alternatives, like example (5), or that exhibit no parallelism, like example (9). However, not all cases of contrast can be explained in his approach (see example (12)), and in addition, in several cases it seems that too few contrastive accents are predicted (see examples (4), (5) and (9)). The predictive problems of Prevost and Pulman have different causes. A few of the examples from Table I do not meet their respective conditions on contrast, imposed in addition to the presence of a pair of alternatives: example (7) falls short of the number of alternative pairs required by Prevost, and the sentences in example (12) do not have the parallel backgrounds required by Pulman. Still, the most important limitation of the approaches of Prevost and Pulman is their restriction that alternative items should have similar properties. Due to this restriction, they cannot handle cases of contrast involving items that *in isolation* would never be considered contrastive,

like ‘clock’ and ‘game’ in (5). Such cases are quite common, however; another example is given below.

- (13) Mary came out against John in the finals of a TV quiz. In the end, SHE won a CRUISE; HE won a TOASTER.

Examples like these indicate that the constraints on alternative items posed by Prevost and Pulman should be dropped. Apparently, any two items may be considered as contrastive if they have the same thematic roles (e.g., agent or patient) with respect to the same verb. In terms of Pulman’s approach, it seems to be only parallelism of the ‘background’ that counts, and not the sortal properties of the contrasted items. Based on this observation, a very simple rule for the placement of contrastive accents suggests itself: if two sentences have the same verb, contrastive accents are assigned to the distinct role fillers. Following this simple approach, the contrastive accents in examples (4), (5), (7), and (13) are all directly predicted. To account for examples (9) and (12), the rule should be extended to include verbs that are semantically equivalent.

Summing up, the approaches of Prevost and Pulman fail to predict several cases of contrastive accentuation, because of their restriction that alternative items should have similar properties. Van Deemter’s approach does not have this limitation, but suffers from other problems. As a practical alternative, a simple rule is proposed for the assignment of contrastive accents, based on the comparison of role fillers for the same verbs. The validity of this rule has been tested in an experiment, which is discussed below.

4. Experiment: parallelism, coherence and contrast

This section describes an experiment that was carried out to test which discourse contexts trigger a preference for contrastive accent. In particular, we wanted to test the assumptions that the mere presence of a pair of alternatives (in the sense of Prevost and Pulman) does not trigger a preference for contrastive accent, but that parallelism between sentences does. In the experiment, the subjects had to indicate for a number of texts which of two spoken versions of these texts they found the most natural-sounding. The two versions differed with respect to the accentuation of one target item.

4.1. Hypotheses and assumptions

The following hypotheses were tested in the experiment:

Hypothesis I: the presence of one pair of alternative items in two consecutive sentences does not trigger a preference for contrastive accent.

Hypothesis II: parallelism between two consecutive sentences does trigger a

preference for contrastive accent.

The first hypothesis corresponds to an assumption shared by all approaches discussed in the previous section, which is that the mere presence of one pair of alternative items is not a sufficient condition for contrast between sentences. As noted in Section 3.4, all approaches have an additional constraint on contrastiveness: the presence of a second pair of alternatives (Prevost), parallel ‘backgrounds’ (Pulman), or contrariety after substitution (van Deemter). In the alternative approach sketched at the end of Section 3.4, the detection of contrast is not based on the presence of alternative items at all, but on the presence of identical verbs with different role fillers.

The second hypothesis is directly related to the simple rule for contrast prediction suggested at the end of Section 3.4. Here, two sentences are assumed to be parallel if they have the same verb, regardless of the sortal properties of its arguments. In such parallel constructions, people are expected to prefer accentuation of the verb’s different role fillers. This is in line with the observation by Chafe [1976] that contrast is related to the presence of different candidates for the same role. The informal definition of parallelism used in the experiment simplifies that of Pulman [1997] in the following ways. First, Pulman’s restriction that ‘obvious’ alternatives should be involved is eliminated; second, the ‘background’ of a sentence, which in Pulman’s theory is obtained after abstracting over the alternative items, is simply equated to the verb; and third, only identical backgrounds (i.e., verbs) are regarded as being parallel. A (translated) example of a parallel construction used in the experiment is the following.

(14) The Pope₁ kissed the ground. The head of state kissed him₁.

In (14), our basic contrast rule predicts an accent on the pronoun *him* because its referent, the Pope, is contrasted with the ground, which was kissed in the preceding sentence. In non-parallel constructions like (15), the Pope and the ground are not considered to be contrastive.

(15) The Pope₁ kissed the ground. The head of state congratulated him₁.

To test Hypothesis I in the experiment, only alternative items are used which may be regarded as alternatives in isolation (that is, outside a parallel construction). Examples are {cat, dog}, {car, bike}, {relaxed, nervous}, etc. These are all entities that clearly belong to the same mother category. They correspond to the *c-parallel* items of Gardent and Kohlhase [1997], being entities (of all logical types) that have both a common and a distinguishing sortal property. To test Hypothesis II also a few non-obvious alternatives were used, like the Pope and the ground in example (14).

Concerning the relation between information status and accentuation we adopt the same assumptions that underly the D2S accentuation algorithm (see Section 2.2). These can be summed up as follows. Information is regarded as *given*

to the hearer if it has been expressed previously in the same discourse segment, and words expressing given information are deaccented. For details on how givenness can be determined, see van Deemter [1994b]. Information which is not given is regarded as *new*, and words expressing new information are accented. Finally, both new and given (previously mentioned) information can be *contrastive*. Words or phrases expressing information that is contrastive are always accented, even if they are given. The exact landing place of accents within a phrase is determined by Focus-Accent theory.

4.2. Method

The hypotheses presented in Section 4.1 were tested by means of a small perception experiment. In the experiment, twenty native speakers of Dutch (14 male, 6 female, with different ages and backgrounds) were presented with twenty short texts in Dutch, displayed on a computer screen. Displayed next to each text were two buttons, which upon clicking played two different spoken versions of each text. The subjects were instructed to first read each text, and then listen to its two spoken versions. They could listen to each version as often as they liked. The spoken versions of the texts differed with respect to the accentuation of a target word in their final sentence: in one version (the ‘accented’ version), this word was accented, in the other version (the ‘unaccented’ version) it was not.

For each text, the subjects had to indicate which of its two spoken versions they found the most natural sounding. The subjects knew that the two versions only differed with respect to the pronunciation of the last sentence, but they were not told the exact nature of this difference, in order to reduce the awareness of the variable being tested.

4.3. Materials

The texts used in the experiment were constructed to test Hypotheses I and II, and consisted of two or (in most cases) three sentences. The first sentence of each text introduced a target item X. The last sentence of each text also contained a reference to X, which used the same wording. X was assumed to constitute given information by then. This was important for the experiment because, as explained above, words expressing previously mentioned (‘given’) information are assumed to be deaccented, *except* if the information they convey is contrastive. (As we will see, this assumption turned out to be too simple.) Thus, a subject’s preference for the ‘accented version’ of a text could be interpreted as a preference for contrastive accent on the target word (= the final reference to X). For target words expressing new information this interpretation would not follow, because new information is always accented. The texts were divided into three categories, representing three different types of context for the target word. An example from each category is given in Figure 9.

(Figure 9 approximately here.)

The texts in Category I were used to test Hypothesis I. The texts contained a reference to an alternative item Y, preceding the final reference to X, but there was no parallelism between the sentence introducing Y and the sentence containing the final reference to X. (In one case, corresponding to example (15), the reference to Y preceded that to X within the same sentence.) Category I contained ten texts, two of which consisted of two sentences. For the texts in this category, neither of the approaches to contrast discussed in Section 3 would predict a contrastive accent on the target item X. According to the deaccenting strategy based on previous mention, employed in D2S and other spoken language generation systems, the target item X would be deaccented.

The texts in Category II were used to test Hypothesis II. These texts showed parallelism between the sentence containing the final reference to X and the preceding sentence. Category II contained seven texts, one of which consisted of two sentences. Most of the presumably contrastive role fillers in the parallel sentences of Category II would also be counted as alternatives in isolation (just like those used in Category I). As a consequence, for nearly all texts of this category Prevost and Pulman would also predict a contrastive accent on target item X. It seems that van Deemter would predict a contrast in all texts, although in most cases this would be based on parallelism rather than contrariety. (For instance, receiving a bouquet and receiving a bottle of wine, as in the middle text from Figure 9, are not contrary to each other.)

Finally, the texts in Category III contained no reference to an alternative item, and did not show any parallelism. This category was not directly related to the hypotheses, but was added to test the underlying assumption that the presence of an intervening sentence does not affect the givenness of an item. It contained only three texts, all of which consisted of three sentences. Neither of the approaches to contrast discussed in Section 3 would predict any contrast in these texts. Following the deaccenting strategy based on previous mention, the target item X would be deaccented.

Figure 9 shows a schematic representation of the texts in each category, together with an example text and its translation. In the schematic representations, X represents the target item and Y its alternative. In the texts of Category II, P and Q are a second pair of potentially contrastive role fillers. Finally, A, B and C represent the ‘backgrounds’ of the sentences. In the example texts from Figure 9, the phrase *de burgemeester* (“the mayor”) refers to the target object X, and the phrase *de beeldhouwer* (“the sculptor”) refers to the alternative item Y. In the text of Category II, the phrases *een bloemetje* (“a bouquet”) and *een fles wijn* (“a bottle of wine”) form the other pair of items {P, Q}.

The spoken versions of the texts were generated using the speech synthesis system Calipso [Gigi and Vogten, 1997, Klabbers, 2000], which is also used for speech generation in D2S. Accents were assigned according to Focus-Accent theory. For the accented text versions, the target word was assumed to be in focus, leading to its accentuation. As explained in Section 2.2, Calipso does not make a distinction between accents signaling newness or contrast. This means that in

our material, the presumably contrastive target words in Category II were *not* set apart from other accented words by assigning them a specific ‘contrastive’ type of pitch accent. For the unaccented text versions, the target word was assumed to be defocused due to givenness, and therefore it was not accented. In some of the unaccented cases, application of the Default Accent rule (see Section 2.2) caused an accent to land on a word that would otherwise have remained unaccented. This happened, for instance, to the verb *congratulated* in example (15).

4.4. Results

The results of the experiment are shown in Table II. For each category, the table indicates the average number of preferences for the unaccented and for the accented versions of the texts in that category, and whether the difference between the two is significant using the binomial test ($p < 0.05$). As Table II shows, there is a significant preference for the accented text versions in Category II. This is as predicted by Hypothesis II. For the other two categories, showing no parallelism, the difference between preferences for the accented or the unaccented versions is not significant. This is not entirely as expected, because assuming that the target items in these categories constitute given, non-contrastive information, a strong preference for the unaccented text versions would be predicted. An attempt to explain the results for Categories I and III is made in Section 4.6.

(Table II approximately here.)

4.5. Discussion

The results of the experiment indicate two things. First, the mere presence of a pair of alternative items in consecutive sentences does not trigger a significant preference for contrastive accent (shown by the results for Category I). This confirms Hypothesis I, and is in line with all approaches to contrast prediction that have been presented. Second, the presence of parallelism between sentences does trigger a preference for contrastive accent (shown by the results for Category II, as opposed to Categories I and III). This confirms Hypothesis II. The notion of parallelism used here is based on the ‘same verb’ rule informally proposed at the end of Section 3.4. In the texts used to test Hypothesis II, the predictions of this rule largely overlap with those of the formal approaches to contrast discussed in Section 3. This means that the results of the current experiment do not indicate which approach is the most valid; for this, an experiment that is especially geared to such a comparison will be required.

The experimental findings may be explained as follows. Because the events that are described in the parallel sentences of Category II are similar, people feel the need to emphasise the differences between them by accenting the words that refer to the different participants in the events (i.e., the alternative items). On the other hand, the events described in the non-parallel sentences of Category I

are not similar, and people feel no need to distinguish them by emphasising the alternative items.

Although the above explanation accounts for the significant preference for accentuation in Category II, it does not fully explain the results for the other two categories. According to the ‘event-based’ account of contrast, the target items in Categories I and III are not contrastive, and therefore a significant preference for the unaccented text versions in these categories would be expected, based on the assumption that givenness is directly linked to previous mention. Nevertheless, Table II does not show a significant preference for either the accented or the unaccented text versions in Categories I and III. It is important to note, however, that Table II shows the *average* preferences for accented and unaccented text versions, taken over all texts within Categories I and III. So, although the average preferences for the accented and the unaccented versions are nearly the same, this need not be the case for the preferences associated with the individual texts in these categories. In fact, there are large differences in preference within Categories I and III: for some texts in these categories, there is a strong (but not always significant) preference for the unaccented version, and for other texts there is a preference for the accented version. Since these differences within Categories I and III cannot be explained by the presence or absence of an alternative item or of parallelism, another factor must play a role here. In Section 4.6 it is argued that this factor is (local) coherence.

4.6. Coherence

Assuming that the target items from Categories I and III are indeed not contrastive, the only remaining explanation for their accentuation is that although they have been previously mentioned, they do not represent given information. As was noted by Chafe [1976], a previously mentioned item may lose its givenness if it is no longer in the listener’s consciousness, nor recoverable from the preceding discourse [Halliday, 1967]. Hirschberg [1992] relates givenness to global discourse structure, assuming that items only remain given within one discourse segment [Grosz and Sidner, 1986]. The texts used in our experiment seem too short to be divided into discourse segments, so the accentuation of supposedly given items in Categories I and III cannot be explained through the presence of a discourse segment boundary that limits their givenness. Still, it may be the case that *local* (i.e., intra-segmental) discourse structure plays a role here, with the local relations between sentences influencing the accentuation of discourse items.

A well-known theory of local discourse structure is Centering Theory [Grosz et al., 1995]. Centering Theory assigns to each utterance U_i a set of *forward looking centers* $Cf(U_i)$ containing representations of the entities referred to in U_i . This set is partially ordered to reflect the relative prominence of the referring expressions within the utterance. Additionally, each utterance U_i is (in principle) assigned a *backward looking center* $Cb(U_i)$. This is an entity which was

also mentioned in the preceding utterance U_{i-1} , and is therefore a member of $Cf(U_{i-1})$. The Cb functions as a kind of link between two utterances, establishing coherence between them. For example, in the sequence *The dog chased the cat. The cat chased a mouse*, the Cb of the second sentence is the cat. Centering Theory distinguishes different types of sentence transitions, which are more or less coherent depending on the relation of $Cb(U_i)$ to the Cb and Cf of the preceding sentence.

The relation between accentuation and local discourse structure, analysed in terms of Centering Theory, has been investigated by Cahn [1995], Nakatani [1997], and Kameyama [1999]. The most extensive of these studies is that of Nakatani [1997]. Based on an analysis of spontaneously spoken English monologues, Nakatani proposes an enriched taxonomy of the given/new information status, relating accentuation to both global and local discourse structure. When discussing the occurrence of accented pronouns in subject position, Nakatani observes that accent “signals reference to a previous center that was not realized in the immediately preceding utterance, and therefore was not a member of the Cf list of the immediately preceding utterance” (Nakatani [1997]:35). For non-pronominal expressions, Nakatani finds that accentuation is generally determined by the global discourse structure. Although Nakatani suggests that local discourse structure is mainly relevant for the accentuation of pronouns, her observations concerning accented pronouns also seem to hold for the target items used in Categories I and III of our experiment (none of which are pronouns). This is discussed in more detail below.

A Centering analysis of the texts used in our experiment reveals that they can be divided into two basic coherence classes, based on the presence or absence of a Cb in their final sentence. (Note that, like Kameyama [1986], we assume the presence of a Cb to be optional.) If none of the items mentioned in the final sentence of a text (including the target item) occurred in the preceding sentence, the final sentence has no Cb. In this case, the local coherence between the two sentences is obviously minimal, and the text is classified as ‘incoherent’ (IC). If the final sentence does have a Cb, the text is classified as ‘coherent’ (C), ignoring any differences in the level of coherence within the C class. In each of the Categories I to III, approximately half of the texts belongs to the C class, and half of the texts belongs to the IC class. In Category II, with the parallel texts, there appears to be no relationship between accentuation and coherence class: in this category, the accented version was preferred for all texts except one (an IC text, where the two spoken versions scored equal). This suggests that the effect of parallelism is strong enough to overrule any effects of local coherence. For this reason, only the texts of the combined Categories I and III are considered in our analysis. For these texts, there does seem to be a relation between coherence class and accentuation preference. The average accentuation preferences per coherence class are shown in Table III. Although the differences in preference are non-significant, a clear trend is visible: coherence of a text (i.e., the presence of a Cb in the final sentence) leads to a preference for its unaccented

version, and incoherence (i.e., the absence of a Cb) leads to a preference for the accented version.

(Table III approximately here.)

The status of the target items in the IC texts closely corresponds to that of the accented subject pronouns analysed by Nakatani [1997]: the items have been previously mentioned, but not in the immediately preceding utterance. According to Nakatani, if the referent of an accented subject pronoun is not a member of the preceding utterance’s Cf list, this indicates the start of a new discourse segment. In our IC texts, the target item is not a pronoun but a full NP and moreover, none of the other entities mentioned in the final sentence occur in the preceding Cf list either. The combination of these factors strongly suggests the presence of a discourse segment boundary between the penultimate and the final sentences of the IC texts. Since givenness of discourse items is generally not supposed to hold across discourse segments, this seems to explain the preference for accentuation of the target items in these texts. This effect of local incoherence is similar to that of ‘displacement’ found in an experiment by van Donzel [1999], where listeners judged the prosodic prominence of items in different ‘re-told’ or ‘re-read’ versions of a short story. In that experiment, displaced discourse items, i.e., items which had been previously mentioned but could no longer be referred to using a pronoun, were relatively often perceived as prominent by the listeners. In the IC texts used in our experiment, the target items are all displaced in the sense that they cannot be pronominalised. Of course, it should be kept in mind that the findings of Nakatani and van Donzel are based on a study of human speech production, whereas in our experiment we studied listener preferences. Although there seems to be a close correspondence between the two, they are not expected to be fully comparable.

There are insufficient data available here to justify strong conclusions concerning the link between local incoherence and the accentuation of given items, but as the current (sparse) data are in line with the findings of Nakatani [1997] and van Donzel [1999], it seems safe to assume that the slight preference for the accented versions of the IC texts in Categories I and III may be attributed to decreased accessibility (or givenness) of the target object, and not to contrast.

4.7. Conclusions

Summarising, the experiment has shown that parallelism between sentences causes a significant preference for contrastive accent, and that the mere presence of a pair of alternative items does not. Presumably, this is because people prefer to emphasise the points of difference between similar events by assigning contrastive accents to the different actors in these events. In addition, it has been observed that people show a slight preference for accenting previously mentioned items in ‘incoherent’ texts; these items appear to be regarded as ‘new’ again (at

least by some people). In texts that are coherent and show no parallelism, people tend to prefer deaccentuation of previously mentioned items. In texts that do show parallelism, there is no effect of (in)coherence: here, there is a general, significant preference for accentuation of given items.

5. Assigning contrastive accent in D2S

In this section we present a practical method for the generation of contrastive accent in D2S, the system described in Section 2. The proposed method, which has been successfully implemented in the GoalGetter system, is not based on any of the approaches to contrast discussed in Section 3, as these could not be straightforwardly applied in the LGM of D2S. Instead, the basic rule that was informally presented at the end of Section 3.4 is taken as the starting point. This rule was tested in a small experiment, which confirmed our expectation that people prefer to have a contrastive accent on different arguments of the same verb. This idea is taken one step further here: contrastive accent is assumed to be triggered by the subsequent mention of similar events, which may or may not be expressed using the same verb. To determine event similarity, the proposed method for contrast prediction in D2S compares the data structures that serve as the basis for generation within the LGM, instead of looking at the semantic representations of generated sentences.

5.1. Deriving contrast from data structures

In Section 3, several methods for contrast prediction have been discussed. As we have seen, none of these can predict all cases of contrast that were presented; but what is more important for their usability in D2S is that in some way or other, these methods all make use of semantic sentence representations to determine contrast. This makes it impossible to apply them in the LGM, because the syntactic templates used in the LGM are not associated with semantic representations. The only sentence representations that are available within LGM are the syntactic structures that are present in the templates. These are not quite suitable for determining contrast, however, since – as originally argued by Ladd [1980] – contrast appears to be a matter of meaning rather than form. This claim can be supported by examples like (9),(12) and (16) below, which is cited by Delin and Zacharski [1997].

- (16) When I arrived back in Croton I bumped into Laesus, though I had not expected to see him again and HE looked pretty surprised at seeing ME.
(Accents marked in original: Lindsey Davis 1991. *Shadow in bronze*. London: Pan Books. p. 98)

In this example, there is no parallelism at the surface level, but as argued by Delin and Zacharski, the sentences clearly stand in contrast to each other at the level of the *events* that are being described. Examples like these suggest

that what is relevant for contrast is not so much the form and wording of an utterance, but the kind of event that is being expressed. In the experiment discussed in Section 4, contrastive accent was consistently assigned to items that were involved in similar events. In the texts used in the experiment, similar events were expressed using identical verbs, causing parallelism at both the surface level and at the event level. The experimental results therefore do not give a definite answer as to which is the level at which contrast should be determined. However, on the basis of examples like those discussed above we assume that it is parallelism at the event level which is the primary source of contrast.

In D2S applications like GoalGetter, such ‘event-based’ contrasts occur very frequently. Since the generation method used in D2S is aimed at achieving variation in the generated texts [Odiijk, 1995, Theune et al., 2001], similar events can be expressed in many linguistically varied ways, e.g., by using different syntactic templates. As a consequence, two sentences that do not show any surface parallelism may still express the same kind of event, as illustrated by example (17) from GoalGetter (shown in translation).

- (17) The score was opened in the sixth minute by Koeman from Feyenoord.
EIGHT minutes LATER KLUIVERT scored a goal for AJAX.

In this example, both sentences describe the scoring of a goal. There is a clear contrast between the two events: the first goal was scored by Koeman for Feyenoord, whereas the second one was scored (eight minutes later) by Kluivert for Ajax. Contrastive sequences such as (17) are very common in the texts generated by GoalGetter; in Section 5.2, some others are discussed.

So, for the prediction of contrast in D2S we assume that contrast does not depend on the surface form of the generated sentences, nor on the presence of items that are ‘of the same type’, but on the kinds of events that are being expressed. These events may be regarded as the ‘deep’ semantics, or ‘conceptual representation’ of the sentence. A similar approach to contrastive accent assignment is advocated by Delin and Zacharski [1997], who discuss the generation of intonation contours in the BRIDGE spoken dialogue system. Inspired by Schmerling [1976], they say that “for a discourse entity that represents an eventuality, the speaker determines what distinguishes this eventuality from others” (and accents the associated expressions). They do not explain how this strategy is actually carried out in their system.

In D2S, the data structures expressed by the generated sentences (see Section 2.1) may be seen as the conceptual representations of these sentences. The presence of contrastive information should therefore be predictable by looking at these data structures, so that contrast detection in D2S can be based on a simple principle: two consecutive sentences that express the same type of data structure (and therefore express similar information) are assumed to be contrastive.

DEFINITION 1: *Contrast*

1. *Two consecutive sentences are contrastive if they express contrastive data structures*
2. *Two data structures are contrastive if*
 - (a) *they are of the same type T and*
 - (b) *they have contrastive values for at least one attribute A of T*
3. *Two values are contrastive if*
 - (a) *they are non-identical primitives or*
 - (b) *they are contrastive data structures*

The data structures used in the LGM of D2S consist of a number of attributes, which may either have a ‘primitive’ value (i.e., a value that has no internal structure) or a complex value (i.e., a value which is itself a data structure). In the latter case, part 2 of Definition 1 is applied recursively. Given Definition 1, the proposed method for assigning contrastive accent can be described as follows:

Assignment of contrastive accent: if two consecutive sentences S_{i-1} and S_i are contrastive as defined in Definition 1, the constituents of S_i that express contrastive values should be marked [+C], indicating that they are *in focus* due to contrast.

As discussed in Section 2.2, constituents that are marked [+C] must receive a pitch accent, the landing place of which is decided on the basis of (a version of) Focus-Accent theory. They cannot be deaccented due to givenness.

It should be noted that only the relevant constituents of S_i , the *second* sentence in a sequence of two, are marked [+C]. In the LGM, the texts that are generated are not planned in advance but produced incrementally, as described in Section 2.1. This lack of pre-planning makes it impossible to assign contrastive accents to the first sentence (S_{i-1}) of a pair of contrastive sentences. During the generation of a sentence, it is not yet known whether the following sentence will stand in contrast to it; the only thing that can be determined is whether the current sentence is contrastive to its predecessor. This is comparable to the situation where a speaker has not yet thought of what to say next when he utters the first of two contrastive utterances.

5.2. Contrastive accent assignment: illustrations

We now illustrate the mechanism for predicting contrast from data structures using some (translated) examples from the GoalGetter system. As a first illustration, we use the middle paragraph of the example text in Figure 5, discussed in Section 2.1 and repeated here as (18). As was explained in Section 2.1, GoalGetter’s football reports are generated on the basis of a typed data structure, which is derived from an input table containing information about a specific match. An example data structure is shown in Figure 3. The field `goallist` of the GoalGetter data structure contains a sequence of structures of type `goal_event`,

each specifying the team for which a goal was scored, the player who scored, the time and the kind of goal: normal, own goal or a goal resulting from a penalty (the latter goal types are indicated by special markers in the input table). The sentences in example (18) all express such a *goalEvent*. The last two sentences and their data structures are shown in Figure 10. For simplicity, the values of the attributes **team** and **player** are represented as primitives, corresponding to the names of the team and the player. In reality, these values are complex data structures consisting of a number of attributes (see example (20) below).

- (18) The team from Sittard took the lead after seventeen minutes through a goal by Hamming. One minute later Schenning from Go Ahead Eagles equalised the score. After FORTY-EIGHT minutes the FORWARD HAMMING had his SECOND goal noted.

As can be seen in Figure 10, three attributes (**team**, **player** and **minute**) of the *goalEvent* expressed by the final sentence of (18) have a value that is different from the corresponding value in the preceding *goalEvent*. This means that the two data structures are contrastive, as defined in Definition 1: they are of the same type (*goalEvent*), and they have contrastive (i.e., non-identical) values for at least one – in fact, three – of their attributes. Therefore, the two sentences are regarded as contrastive, even though they do not use the same verbs, and all phrases in the final sentence expressing a contrastive value receive a contrastive accent. As noted in Section 2.2, these are the time expression *after forty-eight minutes* (expressing the value of the **minute** attribute) and the description *the forward Hamming* (expressing the value of **player**), which must be accented despite expressing previously mentioned information. The value of the **team** attribute is not expressed in the surface structure of the sentence, because after having mentioned the first goal by Hamming, the system assumes that the hearer knows to which team this player belongs. However, if the team had been mentioned in the last sentence, it would have received contrastive accent.

(Figure 10 approximately here.)

A somewhat more complicated example of contrastive accentuation in D2S is (19), a translation of another text fragment generated by GoalGetter. Version (19)a shows the accentuation pattern which would result from deaccenting previously mentioned information without taking contrast into account, whereas (19)b shows the accentuation pattern generated by the D2S accentuation algorithm, which includes determination of contrast as sketched above.

- (19)a In the sixteenth minute the Ajax player Kluivert kicked the ball into the wrong goal. TEN minutes LATER OVERMARS scored for Ajax.
 b In the sixteenth minute the Ajax player Kluivert kicked the ball into the wrong goal. TEN minutes LATER OVERMARS scored for AJAX.

In (19)a the second reference to the team Ajax is deaccented because Ajax was previously mentioned. This results in a confusing accentuation pattern: on the one hand the listener knows that scoring an own goal means scoring for the opposing team (i.e., *not* Ajax), but on the other hand the accentuation pattern suggests that both Kluivert and Overmars scored for Ajax. In contrast, the accentuation pattern in (19)b does not produce such a clash, but actually seems to make it easier for the hearer to process this text fragment. The accent on the second *Ajax* emphasises that the previous goal was not for Ajax, and thus helps to guide the hearer to the correct interpretation. As in the previous example, the required contrastive accents can be immediately derived from the data structures expressed by the subsequent sentences in (19), given in Figure 11. A simple inspection of the two **team** attributes reveals that they have different values, and so the phrase expressing the **team** attribute in the final sentence of (19) should receive contrastive accent, even though the corresponding value of the previous sentence was not overtly expressed.

(Figure 11 approximately here.)

Definition 1 allows ‘contrastive values’ to be determined recursively: a contrastive value may correspond either to a primitive or to a data structure, which in its turn contains at least one contrastive value. In this way, contrastive values can occur at different levels in the data structure. The effects of this mechanism can be illustrated using the following example, of which the corresponding data structures are given in Figure 12.

- (20) In the twelfth minute the Nigerian player Kanu scored a goal for Ajax.
Six minutes later Larsson, the Swedish forward, scored for Feyenoord.

(Figure 12 approximately here.)

Figure 12 shows the actual values of the **player** attributes, which are data structures instead of primitives. (In Figures 10 and 11, simplified values were shown for the **player** attributes.) Some of the values of the attributes in these **player** data structures are used in the referring expressions for the players, e.g., the description *the Nigerian player Kanu* includes the values for the **last_name** and **nationality** attributes. Now, using Definition 1, it can be determined that not only the values for the *goal_events*’ **player** attributes in Figure 12 are contrastive, but also the values for the players’ attributes **first_name**, **last_name** and **nationality**. This means that in the second sentence, not only the expression for the player as a whole (*Larsson, the Swedish forward*) should be marked as contrastive, but also the expressions for the **last_name** and **nationality** values (*Larsson* and *Swedish*, respectively). This is shown in Figure 13. The word *forward* does not express a contrastive value, and is therefore not marked [+C]. Still, this word does receive an accent due to ‘newness’ (since it has not been previously mentioned), which results in the accentuation pattern LARSSON, *the* SWEDISH FORWARD.

(Figure 13 approximately here.)

The approach described in this section is in keeping with the results of the experiment discussed in Section 4; contrastive accent is triggered by the subsequent mention of similar events. The presence of an alternative item in the preceding context does not trigger the assignment of contrastive accent. For instance, in example (21) below, the second and third sentences will not be regarded as contrastive, since the third expresses a *goal_event* but the second does not. Therefore, despite the presence of the alternative *Ajax*, the contrast algorithm will not assign an accent to *Feyenoord* in the final sentence.

- (21) After three minutes Feyenoord took the lead through a goal by Koeman. This caused Ajax to fall behind. In the NINETEENTH minute LARSSON scored for Feyenoord.

5.3. Discussion

An important advantage of using data structures rather than syntactic or semantic representations to assign contrastive accents in D2S is that it allows for the detection of contrast in cases where there is no surface parallelism between sentences. Such cases occur quite frequently in D2S-based applications, but the approaches discussed in Section 3 are not well suited to deal with them. Table IV shows two of the GoalGetter examples discussed in the previous sections, and the corresponding predictions of Prevost [1995], Pulman [1997] and van Deemter [1994a, 1998, 1999]. Example (18) is not included in Table IV, because of its similarity to (17).

(Table IV approximately here.)

The contrasts in (17) are correctly predicted by Prevost, but not by the other two approaches. In addition to the two pairs of alternative items required by Prevost (i.e., the two players and the two teams), this example involves a third pair (the two different times), of which we assume that they are also counted as contrastive in Prevost’s approach. Because the backgrounds of the two sentences do not show any parallelism, Pulman’s conditions on contrast are not met. This also holds for van Deemter, who can explain example (17) neither through parallelism nor through contrariety (as the reader may check). For the more complicated example (19) things are slightly different. Again, Pulman and van Deemter do not predict any contrast, because like (17), (19) involves neither parallelism nor contrariety. Prevost does predict a contrast in (19), based on the presence of two pairs of alternatives (the players and the times). However, he fails to predict the crucial contrastive accent on *Ajax* in the final sentence, because no alternative team has been mentioned. It is interesting to see what happens if we simplify example (19) by leaving out the time references, as in (19)’. The example clearly remains contrastive, but Prevost’s approach no longer recognises this because only one pair of alternatives has remained. On the other hand, in

van Deemter’s approach the adapted example (19) is recognised as a case of contrast. Replacing both *Kluivert* and *Overmars* by a constant now results in a contrariety, because an Ajax player cannot kick the ball into the wrong goal and score for Ajax at the same time. Of course, for a spoken language generation to recognise such a contrariety would require a certain amount of (domain-specific) world knowledge. Such knowledge is not required for contrast detection in D2S, since the relevant information is encoded in the data structures from which the sentences are generated; how this information is expressed in the generated sentences is irrelevant.

It is clear that the method for contrast determination described in this section places a great responsibility on the data structures that are used. The problem of defining parallelism is arguably shifted to the design of the data structures: these must be set up in such a way that potentially contrastive items get assigned identical data types. It is not always clear, however, which items should be regarded as ‘potentially contrastive’. For example, in the GoalGetter application a distinction is made between *card_events* and *goal_events*, as depicted on the left in Figure 14. Following Definition 1, *goal_events* are potentially contrastive to other *goal_events* but not to events of type *card_event*. However, other classifications of events could be made as well, e.g., by adding separate data types for normal goals, own goals and penalties, as shown on the right in Figure 14. Here, the type of a goal is reflected directly in its data structure type, instead of being specified as the value of the **goaltype** attribute of a *goal_event*. (Similarly, *card_events* might be divided into *red_card_events* and *yellow_card_events*. This is not shown.) Since in the alternative situation a penalty and a normal goal would be represented by different types of data structures, two sentences expressing those events would not count as contrastive. As a consequence, no contrastive accent would be predicted in cases like example (19), for which the alternative data structures are given in Figure 15. Clearly, this is not a desirable result.

(Figure 14 approximately here.)

(Figure 15 approximately here.)

We see that the success of the method for contrastive accent assignment presented here depends largely on the level of specification of the data structures; for instance, the alternative event hierarchy shown on the right in Figure 14 appears to be too specific for contrast prediction in the football domain, whereas the current, more global classification shown on the left gives better results. This suggests that there is a level of specificity that is most suitable for use in the determination of contrast. It seems likely that this level corresponds to the ‘basic conceptual level’, the preferred level of categorisation discussed by Cruse [1977], Rosch [1978], Levelt [1989] and others. The notion ‘basic level’ is mostly used in the classification of objects (e.g., people prefer the basic level description ‘apple’ over the more general ‘fruit’ or the more specific ‘Jonagold’), but it can also be used in the classification of events, where for instance ‘walking’ can be regarded

as the basic level between ‘moving’ (too general) and ‘sauntering’, ‘striding’, and so on (too specific). Similarly, it could be argued that ‘goal scoring’ is a basic level concept whereas, e.g., ‘penalty scoring’ is not.

In short, the choice of data structures is essential for the result of contrast detection in D2S, and therefore the data structures should be designed in a way that is psychologically realistic. A plausible requirement is that data structures should be typed at the level of ‘basic level categories’. As a resource for determining these categories, a lexical database like WordNet might be used [Miller, 1995, Fellbaum, 1998]. WordNet is based on theories of human lexical memory and has a large coverage of English nouns, verbs, adjectives and adverbs. For nouns and verbs, basic concept levels are specified.

Of course, the importance of correct classification is not unique for the data structure-based approach advocated here; it also holds for the approaches to contrast described in Section 3 (except the contrariety-approach of van Deemter): to determine which items count as alternatives, a suitable ontology is indispensable. In fact, almost all systems that deal with natural language processing or generation must make use of some kind of ontology to represent world knowledge. For most practical applications, a relatively small, domain-specific classification may be sufficient, but – as argued above – care should be taken to set this up in a psychologically realistic way. At the same time, application domains are continuously getting larger, causing an increased interest in the development of extensive, general ontologies that are re-usable in different systems. An overview of ontologies of various kinds is presented by Bateman [1992], who discusses the different problems and requirements related to their organisation. It would be interesting to investigate the suitability of different existing ontologies for contrast detection; however, this must be left for future research.

6. Future work

So far, the method for contrastive accent assignment described in Section 5 has only been used in the GoalGetter system, but its use in other applications is expected to give good results as well. Besides the generation of sports commentaries, as in GoalGetter, other common NLG applications are the generation of weather or stock market reports. In each of these cases, the input for NLG is formed by regular data. Two example data structures (or ‘messages’) from the weather domain are given in Figure 16. They are based on Reiter and Dale [2000], Figure 4.6. Two sentences that might be generated to express these data structures are shown in (22). Using the method described in Section 5, it can be easily established that the two data structures from Figure 16 are contrastive, and that the points of difference are the day and the type of rain. Contrastive accent is therefore predicted on *28th* (expressing the day) and *sprinkle* (expressing the rain type) in the second sentence of (22).

(Figure 16 approximately here.)

(22) Heavy rain fell on the 27th. March 28TH had a SPRINKLE.

The method for contrastive accent assignment that is proposed in this paper appears to handle most cases of contrast that can be expected in GoalGetter and similar applications; that is, contrasts between similar events involving different actors. However, other kinds of contrasts involve events or states that are contrary rather than similar (cf. van Deemter [1994a, 1998, 1999]). An example would be the sequence *John loves Mary; he hates Sue*. Here, it seems that the verb *hates* should be accented due to its contrast with *love*. Extending the D2S contrast algorithm to deal with this and other kinds of contrasts is a matter for future research. Here we only point out a few possible ways to handle contrasts of the love/hate type. The simplest solution would be to use a (domain-specific) list of contrastive data types, and to stipulate that the verbs used to express these should be marked as contrastive. Another way of dealing with ‘contrary’ events or states might be to represent them using the same data type, at the level of a shared mother category. For example, ‘love’ and ‘hate’ might be treated as alternative values within the general data type *emotional_attitude*. This solution has the problem that it goes against the ‘basic level principle’ suggested in Section 5.3.

In addition to further extensions of our approach, further experiments are required to properly test the assumptions on which it is based. In particular, we still need to validate the claim that contrast is determined at the level of events, regardless of surface form. This will require an experiment with semantically equivalent rather than identical verbs (e.g., *The sculptor received a bouquet; a bottle of wine was given to the mayor*). Another assumption underlying our approach is that any two items can be contrasted, if they play the same role in similar events. However, in the experiment from Section 4, most of the contrasted items were ‘obvious’ alternatives like {cat, dog} and {car, bike}. To properly test our assumption, we need to repeat the experiment with ‘non-obvious’ alternatives only. Finally, the results of our experiment seem to indicate a link between givenness and local coherence (as suggested by Nakatani [1997]). It would be interesting to further investigate this link by means of an experiment that is specifically designed for this purpose.

7. Concluding remarks

It is generally accepted that in Dutch, as in other Germanic languages, words or phrases expressing information that is *new* to the hearer are usually accented, whereas words or phrases expressing *given* information are usually unaccented. In concept-to-speech systems like D2S, discussed in this paper, this human behaviour can be mimicked relatively easily, because givenness or newness of information can be largely determined on the basis of the discourse history of the generated text. Another important factor influencing accentuation is contrast of information, and this is more difficult to account for in concept-to-speech generation. Only fairly recently, some formal approaches to the prediction of contrast

have been proposed. A key notion in these approaches is that of parallelism, which is usually related to the presence of pairs of ‘alternative items’ in subsequent sentences. To be counted as alternatives two items are typically required to have similar properties. An alternative approach, advocated in this paper, is to relate contrast to the presence of identical (or semantically equivalent) verbs, regardless of the properties of their arguments. To find experimental evidence for the latter approach, a small perception experiment has been conducted. In the experiment, two sentences were counted as parallel (and thus contrastive) if they had identical verbs with different arguments. The experimental results indicate that parallelism between sentences indeed causes a significant preference for contrastive accent, and that the mere presence of a pair of alternative items does not. Apparently, people prefer to emphasise the points of difference between similar events by assigning contrastive accents to the different actors in these events.

This idea has formed the basis for a practical method to detect contrast in the D2S system, which does not make use of surface structures or semantic representations. Instead, assignment of contrastive accent is based on the data structures expressed by the generated sentences: if two consecutive sentences express the same type of data structure, they are potentially contrastive, and contrastive accent is assigned to the phrases that express the different parts of the data structures. This method places a high responsibility on the data structures used as the basis for language generation, and further research into the requirements on these data structures is still needed. The proposed method for contrast detection has been implemented in the GoalGetter system.

Acknowledgements

The author wishes to thank the two anonymous reviewers who provided many useful comments for the improvement of this paper. Thanks are also due to Emiel Krahmer, Jan Landsbergen, Kees van Deemter, Jacques Terken, and René Collier, who commented on an earlier version. The work reported in this paper was carried out when the author was working at IPO, the Center for User-System Interaction at the Eindhoven University of Technology. This research was sponsored by NWO (the Netherlands Organisation for Scientific Research), as part of the Priority Programme Language and Speech Technology (TST).

References

- A. Abeillé and Y. Schabes. Parsing idioms in lexicalized TAGs. In *Proceedings of the 4th Conference of the European Chapter of the Association for Computational Linguistics (EACL'89)*, pages 1–9, Manchester, UK, 1989.
- H. Alshawi and R. Crouch. Monotonic semantic interpretation. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics (ACL'92)*, pages 32–39, Newark, USA, 1992.
- M. Ariel. *Accessing Noun-Phrase Antecedents*. Croon Helm Linguistics Series. Routledge, London, 1990.
- N. Asher, D. Hardt, and J. Busquets. Discourse parallelism, ellipsis, and ambiguity. *Journal of Semantics*, 18(1):1–25, 2001.
- H. Aust, M. Oerder, F. Seide, and V. Steinbiss. The Philips automatic train timetable information system. *Speech Communication*, 17:249–262, 1995.
- J.L.G. Baart. *Focus, Syntax and Accent Placement*. PhD thesis, University of Leiden, 1987.
- J. Bateman. The theoretical status of ontologies in natural language processing. In S. Preuss and B. Schmitz, editors, *Proceedings of the Workshop on Text Representation and Domain Modelling – Ideas from Linguistics and AI*, pages 50–99, Berlin, Germany, 1992. KIT-Report 97.
- J. Bock and J. Mazzella. Intonational marking of given and new information. *Memory and Cognition*, 11(1):64–76, 1983.
- D. Bolinger. *Intonation and its Parts: Melody in Spoken English*. Arnold, London, 1986.
- G. Brown. Prosodic structure and the given/new distinction. In D. R. Ladd and A. Cutler, editors, *Prosody: Models and Measurements*, pages 67–77. Springer Verlag, Berlin, 1983.
- D. Büring. *The Meaning of Topic and Focus: the 59th Street Bridge Accent*, volume 3 of *Routledge Studies in German Linguistics*. Routledge, London, 1997.
- J. Cahn. The effect of pitch accenting on pronoun referent resolution. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL'95)*, pages 290–292, Cambridge, USA, 1995.
- W.L. Chafe. Givenness, contrastiveness, definiteness, subjects, topics and points of view. In C. N. Li, editor, *Subject and Topic*, pages 25–55. Academic Press, New York, 1976.

- N. Chomsky. *Lectures on Government and Binding*. Foris, Dordrecht, 1981.
- D.A. Cruse. The pragmatics of lexical specificity. *Journal of Linguistics*, 13: 153–368, 1977.
- A. Cruttenden. *Intonation*. Cambridge University Press, Cambridge, 1986.
- J.R. Davis and J. Hirschberg. Assigning intonational features in synthesized spoken directions. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics (ACL'88)*, pages 187–193, Buffalo, USA, 1988.
- J. Delin and R. Zacharski. Pragmatic determinants of intonation contours for dialogue systems. *International Journal of Speech Technology*, 1:109–120, 1997.
- A. Dirksen. Accenting and deaccenting: A declarative approach. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING'92)*, pages 865–869, Nantes, France, 1992.
- A. Dirksen and H. Quené. Prosodic analysis: The next generation. In V. van Heuven and L. Pols, editors, *Analysis and Synthesis of Speech: Strategic Research Towards High-Quality Text-to-Speech Generation*, pages 131–144. Mouton de Gruyter, Berlin - New York, 1993.
- C. Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, 1998.
- E. Fitzpatrick. The prosodic phrasing of clause-final prepositional phrases. *Language*, 77(3):544–561, 2001.
- C. Gardent and M. Kohlhase. Computing parallelism in discourse. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97)*, pages 1016–1021, Tokyo, Japan, 1997.
- E. Gigi and L. Vogten. A mixed-excitation vocoder based on exact analysis of harmonic components. *IPO Annual Progress Report*, 32:105–110, 1997.
- B. Grosz, A. Joshi, and S. Weinstein. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225, 1995.
- B. Grosz and C. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204, 1986.
- J. Gundel, N. Hedberg, and R. Zacharski. Cognitive status and the form of referring expressions in discourse. *Language*, 69:274–307, 1993.
- M.A.K. Halliday. Notes on transitivity and theme in English. *Journal of Linguistics*, 3:199–244, 1967.

- J. Hirschberg. Using discourse context to guide pitch accent decisions in synthetic speech. In G. Bailly, C. Benoît, and T.R. Sawallis, editors, *Talking Machines: Theories, Models and Designs*, pages 367–376. Elsevier Science Publishers B.V., Amsterdam, 1992.
- J. Hirschberg and R. Sproat. Pitch accent prediction from text analysis. In J. Cole, G.M. Green, and J.L. Morgan, editors, *Linguistics and Computation*, pages 281–296. CSLI Publications, Stanford, 1993.
- J.R. Hobbs and A. Kehler. A theory of parallelism and the case of VP ellipsis. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and the 8th Conference of the European Chapter of the Association for Computational Linguistics (ACL/EACL'97)*, pages 394–401, Madrid, Spain, 1997.
- M. Horne and M. Filipsson. Computational extraction of lexico-grammatical information for generation of Swedish intonation. In J. van Santen, R. Sproat, J. Olive, and J. Hirschberg, editors, *Progress in Speech Synthesis*, pages 443–457. Springer Verlag, New York, 1997.
- A. Joshi. An introduction to Tree Adjoining Grammars. In A. Manaster-Ramer, editor, *Mathematics of Language*, pages 87–114. John Benjamins, Amsterdam, 1987.
- M. Kameyama. A property-sharing constraint in Centering. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics (ACL'86)*, pages 200–206, New York, USA, 1986.
- M. Kameyama. Stressed and unstressed pronouns: Complementary preferences. In P. Bosch and R. van der Sandt, editors, *Focus: Linguistic, Cognitive, and Computational Perspectives*, pages 306–321. Cambridge University Press, Cambridge, 1999.
- E. Klabbbers. *Segmental and Prosodic Improvements to Speech Generation*. PhD thesis, Eindhoven University of Technology, 2000.
- M. Kohlhase. *A Mechanization of Sorted Higher-Order Logic Based on the Resolution Principle*. PhD thesis, Universität des Saarlandes, 1994.
- E. Krahmer, J. Landsbergen, and J. Odijk. A guided tour through LGM; how to generate spoken route descriptions? Report 1182, IPO, Eindhoven, The Netherlands, 1997.
- E. Krahmer and M. Swerts. On the alleged existence of contrastive accents. *Speech Communication*, 34(4):391–405, 2001.
- D.R. Ladd. *The Structure of Intonational Meaning: Evidence from English*. Indiana University Press, Bloomington, 1980.

- G. Lakoff. Presupposition and relative well-formedness. In D. Steinberg and L. Jakobovits, editors, *Semantics: An Interdisciplinary Reader in Philosophy, Linguistics and Psychology*, pages 329–340. Cambridge University Press, Cambridge, 1971.
- W. Levelt. *Speaking, from Intention to Articulation*. MIT Press, Cambridge, 1989.
- G. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.
- A. Monaghan. Intonation accent placement in a concept-to-dialogue system. In *Proceedings of the AAI/ESCA/IEEE Conference on Speech Synthesis*, pages 171–174, New York, USA, 1994.
- D. Nachttegaal. An evaluation of GoalGetter’s accentuation. Report 1142, IPO, Eindhoven, The Netherlands, 1997.
- C. Nakatani. *The Computational Processing of Intonational Prominence: A Functional Prosody Perspective*. PhD thesis, Harvard University, 1997.
- C. Nakatani and J. Chu-Carroll. Using dialogue representations for concept-to-speech generation. In *Proceedings of the ANLP-NAACL Workshop on Conversational Systems*, Seattle, USA, 2000.
- J. Odijk. Generation of coherent monologues. In T. Andernach, M. Moll, and A. Nijholt, editors, *CLIN V: Proceedings of the 5th CLIN Meeting*, pages 123–131, Enschede, The Netherlands, 1995.
- J. Pierrehumbert and J. Hirschberg. The meaning of intonational contours in the interpretation of discourse. In P.R. Cohen, J. Morgan, and M.E. Pollack, editors, *Intentions in Communication*, chapter 14, pages 271–311. MIT Press, Cambridge, 1990.
- X. Pouteau and L. Arévalo. Robust spoken dialogue systems for consumer products: A concrete application. In *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP’98)*, volume 4, pages 1231–1234, Sydney, Australia, 1998.
- S. Prevost. *A Semantics of Contrast and Information Structure for Specifying Intonation in Spoken Language Generation*. PhD thesis, University of Pennsylvania, 1995.
- S. Prevost. An information structural approach to spoken language generation. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL’96)*, pages 294–301, Santa Cruz, USA, 1996.

- E.F. Prince. Toward a taxonomy of given/new information. In P. Cole, editor, *Radical Pragmatics*, pages 223–255. Academic Press, New York, 1981.
- H. Prüst. *On Discourse Structuring, VP Anaphora and Gapping*. PhD thesis, University of Amsterdam, 1992.
- S. Pulman. Higher order unification and the interpretation of focus. *Linguistics and Philosophy*, 20:73–115, 1997.
- E. Reiter and R. Dale. *Building Applied Natural Language Generation Systems*. Cambridge University Press, Cambridge, 2000.
- T. Rietveld, J. Kerkhoff, M.J.W.M. Emons, E.J. Meijer, A.A. Sanderman, and A.M.C. Sluijter. Evaluation of speech synthesis systems for Dutch in telecommunication applications in GSM and PSTN networks. In *Proceedings of the 5th European Conference on Speech Communication and Technology (Eurospeech'97)*, pages 577–580, Rhodes, Greece, 1997.
- M. Rooth. *Association with Focus*. PhD thesis, University of Massachusetts, 1985.
- M. Rooth. A theory of focus interpretation. *Natural Language Semantics*, 1: 75–116, 1992.
- E. Rosch. Principles of categorization. In E. Rosch and B. Lloyd, editors, *Cognition and Categorization*, pages 27–48. Lawrence Erlbaum, Hillsdale, 1978.
- A. Sanderman. *Prosodic Phrasing: Production, Perception, Acceptability and Comprehension*. PhD thesis, Eindhoven University of Technology, 1996.
- S.F. Schmerling. *Aspects of English Sentence Stress*. University of Texas Press, Austin, 1976.
- A. Sluijter, E. Bosgoed, J. Kerkhoff, E. Meier, T. Rietveld, A. Sanderman, M. Swerts, and J. Terken. Evaluation of speech synthesis systems for Dutch in telecommunication applications. In *Proceedings of the 3rd ESCA/COCOSDA International Workshop on Speech Synthesis*, pages 213–218, Jenolan Caves, Australia, 1998.
- R. Sproat. Text interpretation for TtS synthesis. In R. Cole, J. Mariani, H. Uszkoreit, A. Zaenen, and V. Zue, editors, *Survey of the State of the Art in Human Language Technology*, pages 202–209. Cambridge University Press, Cambridge, 1995.
- J. 't Hart, R. Collier, and A. Cohen. *A Perceptual Study of Intonation: An Experimental Phonetic Approach to Speech Melody*. Cambridge University Press, Cambridge, 1990.

- J. Terken and J. Hirschberg. Deaccentuation of words representing ‘given’ information: effects of persistence of grammatical function and surface position. *Language and Speech*, 37(2):125–145, March 1994.
- J. Terken and S. Nootboom. Opposite effects of accentuation and deaccentuation on verification latencies for given and new information. *Language and Cognitive Processes*, 2:145–163, 1987.
- M. Theune. *From Data to Speech: Language Generation in Context*. PhD thesis, Eindhoven University of Technology, 2000.
- M. Theune, E. Klabbers, J.R. de Pijper, E. Kraemer, and J. Odijk. From data to speech: A generic approach. *Natural Language Engineering*, 7(1):47–86, 2001.
- K. van Deemter. Contrastive stress, contrariety, and focus. In P. Bosch and R. van der Sandt, editors, *Focus & Natural Language Processing, Volume 1: Intonation and Syntax*, number 6 in Working Papers of the Institute for Logic & Linguistics., pages 39–49. Cambridge University Press, Cambridge, 1994a.
- K. van Deemter. What’s new? A semantic perspective on sentence accent. *Journal of Semantics*, 11:1–31, 1994b.
- K. van Deemter. A blackboard model of accenting. *Computer Speech and Language*, 12:143–164, 1998.
- K. van Deemter. Contrastive stress, contrariety, and focus. In P. Bosch and R. van der Sandt, editors, *Focus: Linguistic, Cognitive, and Computational Perspectives*, pages 3–17. Cambridge University Press, Cambridge, 1999.
- K. van Deemter, J. Landsbergen, R. Leermakers, and J. Odijk. Generation of spoken monologues by means of templates. In *Proceedings of the 8th Twente Workshop on Language Technology (TWLT 8): Speech and Language Engineering*, pages 87–96, Enschede, The Netherlands, 1994.
- K. van Deemter and J. Odijk. Context modeling and the generation of spoken discourse. *Speech Communication*, 21(1/2):101–121, 1997.
- M. van Donzel. *Prosodic Aspects of Information Structure in Discourse*. PhD thesis, University of Amsterdam, 1999.
- G. van Noord, G. Bouma, R. Koeling, and M. Nederhof. Robust grammatical analysis for spoken dialogue systems. *Natural Language Engineering*, 5(1):45–93, 1999.
- G. Veldhuijzen van Zanten. Adaptive mixed-initiative dialogue management. In *Proceedings of IVTTA 1998*, pages 65–70, Turin, Italy, 1998.

- S. Williams. Generating pitch accents in a concept-to-speech system using a knowledge base. In *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP'98)*, volume 4, pages 1159–1162, Sydney, Australia, 1998.

FIGURE LEGENDS

- Figure 1: Global architecture of D2S.
- Figure 2: Example input table for GoalGetter.
- Figure 3: Data structure corresponding to the table in Figure 2.
- Figure 4: Syntactic template used in GoalGetter (CP = Complementiser Phrase, IP = Inflectional Phrase).
- Figure 5: Translation of a text generated by GoalGetter, based on Figure 2. Accents are indicated by small capital letters, phrase boundaries by /, // or ///, and the start of a new paragraph by <new-par>.
- Figure 6: Possible slot fillings for Template Sent16.
- Figure 7: Final metrical tree of the sixth sentence of Figure 5.
- Figure 8: Stylised examples of the different prosodic versions that are used. Two factors determine their pitch and pausing: accentuation and position relative to a minor/major/final phrase boundary. The pauses are indicated between brackets.
- Figure 9: Schematic representation and example text plus translation for each text category used in the experiment. Here, {X, Y} and {P, Q} are pairs of alternatives; A, B and C are ‘backgrounds’.
- Figure 10: Data structures expressed by the last two sentences of (18).
- Figure 11: Data structures expressed by (19).
- Figure 12: Data structures expressed by (20).
- Figure 13: Syntactic tree for the NP *Larsson, the Swedish forward*.
- Figure 14: Current and alternative event hierarchy in GoalGetter.
- Figure 15: Alternative data structures for the sentences in (19).
- Figure 16: Contrastive data structures from the weather domain.

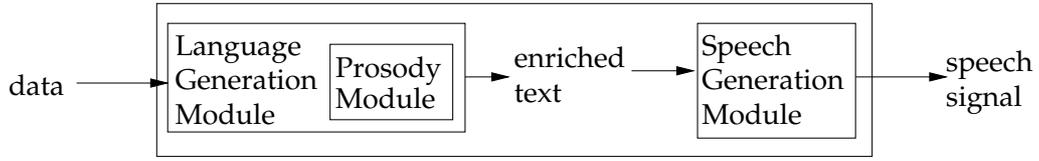


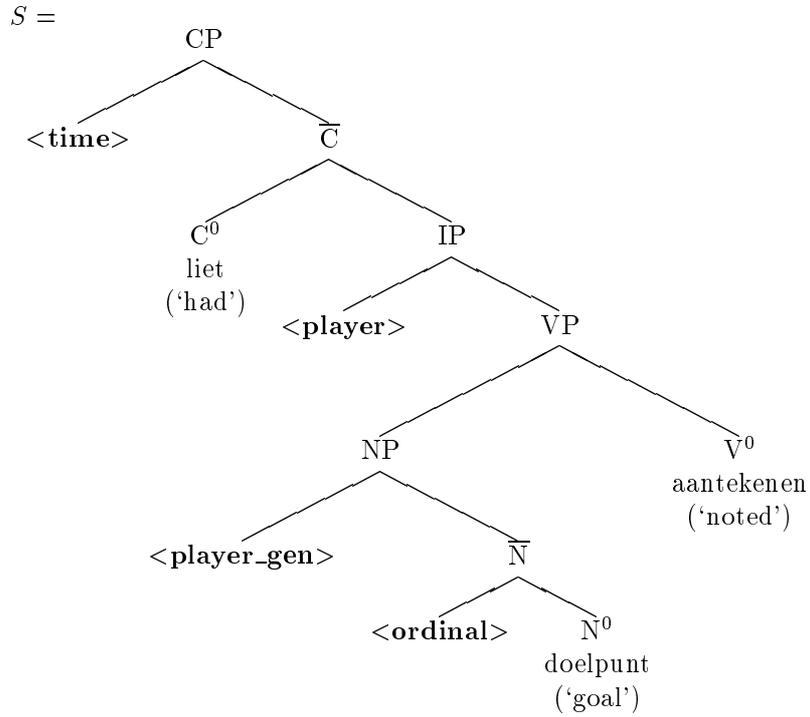
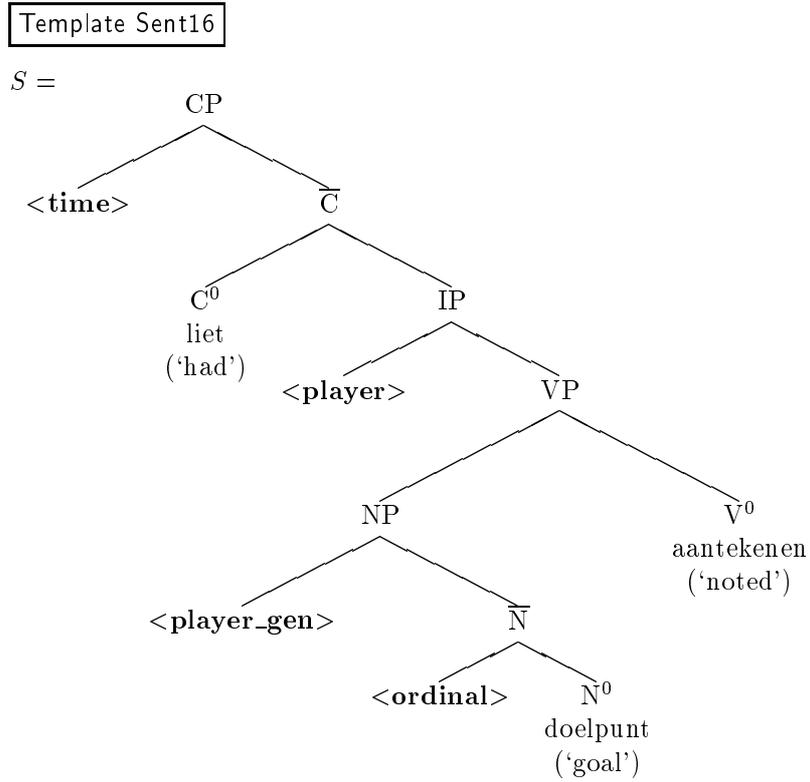
Figure 1: Global architecture of D2S.

FORTUNA SITTARD	2	GO AHEAD EAGLES	1
Hamming (17,48)		Schenning (18)	
Referee:		Spectators:	
Uilenberg		4,500	
Yellow:		Marbus	

Figure 2: Example input table for GoalGetter.

<i>match</i>	
teams :	$\left[\begin{array}{l} \textit{teampair} \\ \text{home_team : Fortuna_Sittard} \\ \text{visitors : Go_Ahead_Eagles} \end{array} \right]$
result :	$\left[\begin{array}{l} \textit{resulttype} \\ \text{home_team : 2} \\ \text{visitors : 1} \end{array} \right]$
goals :	$\left(\begin{array}{l} \textit{goallist} \\ \\ 1. \left[\begin{array}{l} \textit{goal_event} \\ \text{team : Fortuna_Sittard} \\ \text{player : Hamming} \\ \text{minute : 17} \\ \text{type : normal} \end{array} \right] \\ \\ 2. \left[\begin{array}{l} \textit{goal_event} \\ \text{team : Go_Ahead_Eagles} \\ \text{player : Schenning} \\ \text{minute : 18} \\ \text{type : normal} \end{array} \right] \\ \\ 3. \left[\begin{array}{l} \textit{goal_event} \\ \text{team : Fortuna_Sittard} \\ \text{player : Hamming} \\ \text{minute : 48} \\ \text{type : normal} \end{array} \right] \end{array} \right)$
referee :	Uilenberg
spectators :	4500
cards :	$\left(\begin{array}{l} \textit{cardlist} \\ \\ 1. \left[\begin{array}{l} \textit{card_event} \\ \text{team : Go_Ahead_Eagles} \\ \text{player : Marbus} \\ \text{minute : -} \\ \text{type : yellow} \end{array} \right] \end{array} \right)$

Figure 3: Data structure corresponding to the table in Figure 2.



$E = \mathbf{time} \leftarrow \text{ExpressTime}(\text{currentgoal.minute})$
 $\mathbf{player} \leftarrow \text{ExpressObject}(\text{currentgoal.player}, \text{nom})$
 $\mathbf{player_gen} \leftarrow \text{ExpressObject}(\text{currentgoal.player}, \text{gen})$
 $\mathbf{ordinal} \leftarrow \text{ExpressOrdinal}(\text{ordinalnumber})$

$C = \text{Known}(\text{match.teams}) \wedge$
 $\text{currentgoal} = \text{First}(\text{unknown}, \text{match.goals}) \wedge$
 $\text{GoalsScored}(\text{currentgoal.player}) > 1 \wedge$
 $\text{currentgoal.type} \neq \text{owngoal}$

$T = \text{'game_course'}$

Figure 4: Syntactic template used in GoalGetter (CP = Complementiser Phrase, IP = Inflectional Phrase).

Go Ahead EAGLES / visited Fortuna SITTARD // and LOST ///

The duel ended in TWO // - ONE ///

FOUR thousand FIVE hundred SPECTATORS / came to “de BAANDERT” ///

<new-par>

The TEAM from SITTARD / took the LEAD after SEVENTEEN MINUTES / through a GOAL by HAMMING ///

ONE minute LATER / SCHENNING from Go Ahead EAGLES / EQUALISED the score ///

After FORTY-EIGHT minutes / the FORWARD HAMMING / had his SECOND goal noted ///

<new-par>

The match was OFFICIATED by REFEREE UILENBERG ///

He did NOT issue any RED CARDS ///

MARBUS of Go Ahead EAGLES / picked up a YELLOW card ///

Figure 5: Translation of a text generated by GoalGetter, based on Figure 2. Accents are indicated by small capital letters, phrase boundaries by /, // or ///, and the start of a new paragraph by <new-par>.

<time>	{ <i>in the forty-eighth minute, after forty-eight minutes</i> }
<player>	{ <i>Hamming, he, the forward, the forward Hamming</i> }
<player_gen>	{ <i>Hamming's, his</i> }
<ordinal>	{ <i>second</i> }

Figure 6: Possible slot fillings for Template Sent16.

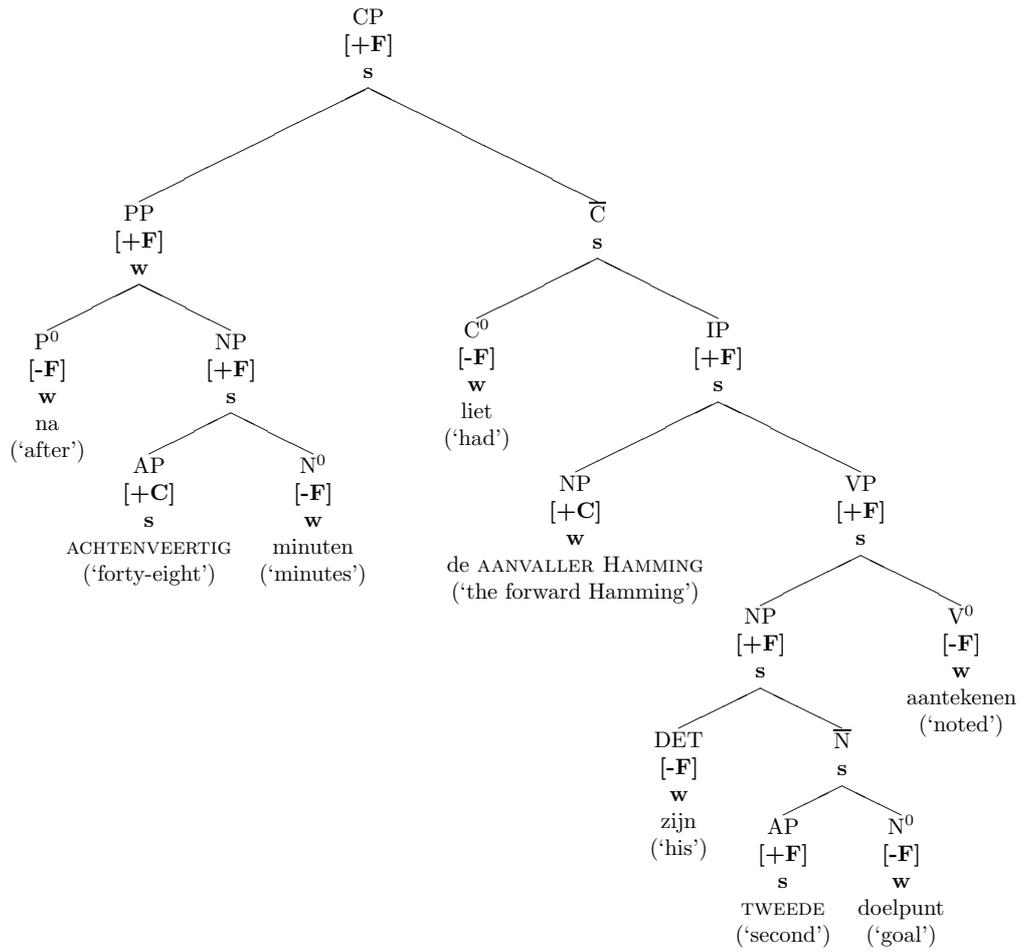


Figure 7: Final metrical tree of the sixth sentence of Figure 5.

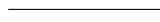
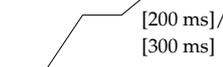
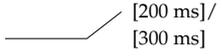
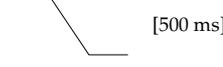
Accent Boundary	Yes	No
None	(1) 	(4) 
Minor / Major continuation	(2) 	(5) 
Finality	(3) 	(6) 

Figure 8: Stylised examples of the different prosodic versions that are used. Two factors determine their pitch and pausing: accentuation and position relative to a minor/major/final phrase boundary. The pauses are indicated between brackets.

		<i>De burgemeester onthulde een standbeeld.</i>
		<i>Als dank kreeg de beeldhouwer een bloemetje.</i>
		<i>De burgemeester hield een toespraak.</i>
I:	1. X A	
	2. Y B	
	3. X C	
		The mayor unveiled a statue.
		By way of thanks, the sculptor received a bouquet.
		The mayor made a speech.

		<i>De burgemeester onthulde een standbeeld.</i>
		<i>Als dank kreeg de beeldhouwer een bloemetje.</i>
		<i>De burgemeester kreeg een fles wijn.</i>
II:	1. X A	
	2. Y B P	
	3. X B Q	
		The mayor unveiled a statue.
		By way of thanks, the sculptor received a bouquet.
		The mayor received a bottle of wine.

		<i>De burgemeester ging op vakantie naar Afrika.</i>
		<i>Het was daar erg warm.</i>
		<i>De burgemeester had veel last van de hitte.</i>
III:	1. X A	
	2. B	
	3. X C	
		The mayor went on holiday to Africa.
		It was very hot there.
		The mayor suffered greatly from the heat.

Figure 9: Schematic representation and example text plus translation for each text category used in the experiment. Here, {X, Y} and {P, Q} are pairs of alternatives; A, B and C are ‘backgrounds’.

```
[ goal_event
  team :    Go Ahead Eagles
  player :  Schenning
  minute :  18
  goaltyp : normal ]
```

One minute later Schenning from Go Ahead Eagles equalised the score.

```
[ goal_event
  team :    Fortuna Sittard
  player :  Hamming
  minute :  48
  goaltyp : normal ]
```

After forty-eight minutes the forward Hamming had his second goal noted.

Figure 10: Data structures expressed by the last two sentences of (18).

$$\left[\begin{array}{l} \textit{goal_event} \\ \text{team :} \quad \text{Feyenoord} \\ \text{player :} \quad \text{Kluivert} \\ \text{minute :} \quad 16 \\ \text{goaltype :} \quad \text{own} \end{array} \right]$$

In the sixteenth minute the Ajax player Kluivert kicked the ball into the wrong goal.

$$\left[\begin{array}{l} \textit{goal_event} \\ \text{team :} \quad \text{Ajax} \\ \text{player :} \quad \text{Overmars} \\ \text{minute :} \quad 26 \\ \text{goaltype :} \quad \text{normal} \end{array} \right]$$

Ten minutes later Overmars scored for Ajax.

Figure 11: Data structures expressed by (19).

```

[ goal_event
  team:      Ajax
  player     [ first_name : Nwankwo
              last_name  : Kanu
              nationality : Nigerian
              position   : forward ]
  minute:    12
  goalttype: normal ]

```

In the twelfth minute the Nigerian player Kanu scored a goal for Ajax.

```

[ goal_event
  team:      Feyenoord
  player     [ first_name : Henryk
              last_name  : Larsson
              nationality : Swedish
              position   : forward ]
  minute:    18
  goalttype: normal ]

```

Six minutes later Larsson, the Swedish forward, scored for Feyenoord.

Figure 12: Data structures expressed by (20).

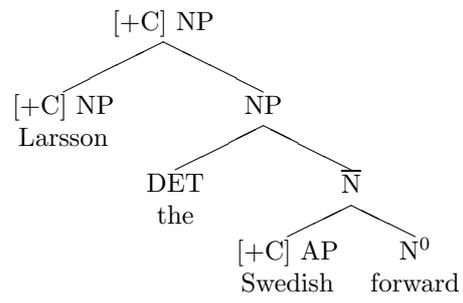


Figure 13: Syntactic tree for the NP *Larsson, the Swedish forward*.

CURRENT

ALTERNATIVE

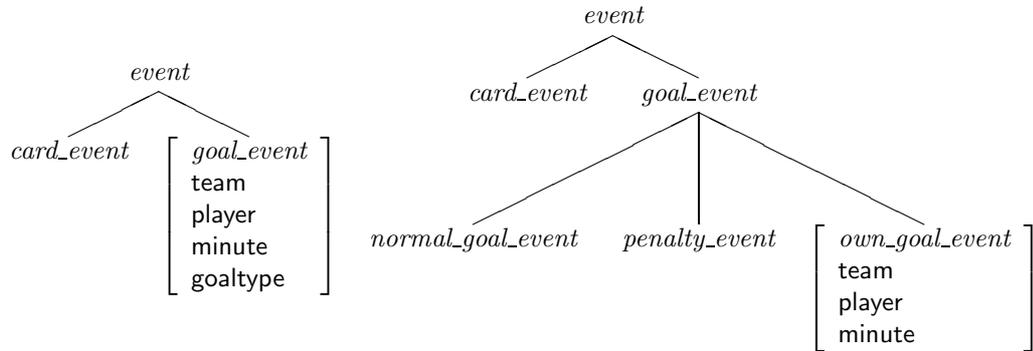


Figure 14: Current and alternative event hierarchy in GoalGetter.

```
[ own_goal_event
  team :      Feyenoord
  player :    Kluivert
  minute :    16 ]
```

In the sixteenth minute the Ajax player Kluivert kicked the ball into the wrong goal.

```
[ normal_goal_event
  team :      Ajax
  player :    Overmars
  minute :    26 ]
```

Ten minutes later Overmars scored for Ajax.

Figure 15: Alternative data structures for the sentences in (19).

$\left[\begin{array}{l} \textit{rain_event} \\ \\ \text{date} \left[\begin{array}{l} \text{day : } 27 \\ \text{month : } 07 \\ \text{year : } 1996 \end{array} \right] \\ \\ \text{raintype: heavy} \end{array} \right]$	$\left[\begin{array}{l} \textit{rain_event} \\ \\ \text{date} \left[\begin{array}{l} \text{day : } 28 \\ \text{month : } 07 \\ \text{year : } 1996 \end{array} \right] \\ \\ \text{raintype: light} \end{array} \right]$
---	---

Heavy rain fell on the 27th.

March 28th had a sprinkle.

Figure 16: Contrastive data structures from the weather domain.

Table 1: Contrast examples from Section 3 and the corresponding predictions by Prevost [1995], Pulman [1997], and van Deemter [1994a, 1998, 1999].

	Contrast predicted by:	Prevost	Pulman	v.Deemter
(4)	The x4 is a solid-state amplifier. The x5 is a TUBE amplifier.	yes	yes	yes
(5)	While he intently watched the clock, SHE watched the GAME.	no	no	yes
(7)	A: John kissed Sue. B: (No,)John kissed MARY.	no(?)	yes	yes
(9)	Bach was an organ mechanic; MOZART knew LITTLE about organs.	no	no	yes
(12)	Stereofool printed a favourable review of the British amplifier. The AMERICAN amplifier was praised by AUDIOFAD.	yes	no	no

Table 2: Average number of people that preferred either the accented or the unaccented versions of the texts in each category. Averages are computed over all texts within each category.

	unaccented	accented	significant
I	10.9	9.1	no
II	4.7	15.3	yes
III	11.0	9.0	no

Table 3: Average number of people that preferred either the accented or the unaccented versions of the texts in each coherence class, combined for Categories I and III. Averages are computed over all texts within each class (5 IC texts and 8 C texts).

	accented	unaccented
IC	12,2	7,8
C	7,2	12,9

Table 4: Contrast examples from GoalGetter and the corresponding predictions by Prevost [1995], Pulman [1997], and van Deemter [1994a, 1998, 1999].

Contrast predicted by:		Prevost	Pulman	v.Deemter
(17)	The score was opened in the sixth minute by Koeman from Feyenoord. EIGHT minutes LATER KLUIVERT scored a goal for AJAX.	yes	no	no
(19)	In the sixteenth minute the Ajax player Kluivert kicked the ball into the wrong goal. TEN minutes LATER OVERMARS scored for AJAX.	yes	no	no
(19)'	The Ajax player Kluivert kicked the ball into the wrong goal. OVERMARS scored for AJAX.	no	no	yes