Generating Tailored, Comparative Descriptions with Contextually Appropriate Intonation

Michael White*
The Ohio State University

Robert A. J. Clark**
University of Edinburgh

Johanna D. Moore†
University of Edinburgh

Generating responses that take user preferences into account requires adaptation at all levels of the generation process. This article describes a multi-level approach to presenting user-tailored information in spoken dialogues which brings together for the first time multi-attribute decision models, strategic content planning, surface realization that incorporates prosody prediction, and unit selection synthesis that takes the resulting prosodic structure into account. The system selects the most important options to mention and the attributes that are most relevant to choosing between them, based on the user model. Multiple options are selected when each offers a compelling trade-off. To convey these trade-offs, the system employs a novel presentation strategy which straightforwardly lends itself to the determination of information structure, as well as the contents of referring expressions. During surface realization, the prosodic structure is derived from the information structure using Combinatory Categorial Grammar in a way that allows phrase boundaries to be determined in a flexible, data-driven fashion. This approach to choosing pitch accents and edge tones is shown to yield prosodic structures with significantly higher acceptability than baseline prosody prediction models in an expert evaluation. These prosodic structures are then shown to enable perceptibly more natural synthesis using a unit selection voice that aims to produce the target tunes, in comparison to two baseline synthetic voices. An expert evaluation and f0 analysis confirm the superiority of the generator-driven intonation and its contribution to listeners' ratings.

1. Introduction

In an evaluation of nine spoken dialogue information systems, developed as part of the DARPA Communicator program, the information presentation phase of the dialogues

* 1712 Neil Ave., Columbus, OH 43210, USA. Web: http://www.ling.ohio-state.edu/~mwhite/.
** 10 Crichton Street, Edinburgh, Scotland EH8 1AB, UK. Web: http://www.cstr.ed.ac.uk/ssi/people/robert.html.
† 10 Crichton Street, Edinburgh, Scotland EH8 1AB, UK. Web: http://www.hcrc.ed.ac.uk/~jmoore/.

Submission received: 31 January 2008; revised submission received: 19 May 2009; accepted for publication: 24 September 2009.

SYS: I found 9 round trips from Hartford to Orlando on October 31st and returning Hartford on November 6th. For option 1, I have an outbound flight with U S Air departing at 7 AM and arriving at 9 49 AM non-stop. A return flight is with U S Air departing at 10 30 AM and arriving at 12 28 AM with 1 stopover. Ticket price is 165 dollars. Please say "next option", or, "flight details", or, "I'll take it".

USER: NEXT OPTION

SYS: For option 2, I have an outbound flight with U S Air departing at 3 30 PM and arriving at 6 19 PM nonstop. A return flight is with U S Air departing at 8 25 PM and arriving at 12 28 AM with 1 stopover. Ticket price is 165 dollars. Say "next option", or, "flight details", or, "I'll take it".

USER: NEXT OPTION

SYS: . . .

Figure 1
Typical information presentation phase of a Communicator dialogue.

was found to be the primary contributor to dialogue duration (Walker, Passonneau, and Boland 2001). During this phase, the typical system sequentially presents the set of options that match the user's constraints, as shown in Figure 1. The user can then navigate through these options and refine them by offering new constraints. When multiple options are returned, this process can be exacting, leading to reduced user satisfaction.

As Walker et al. (2004) observe, having to access the set of available options sequentially makes it hard for the user to remember information relevant to making a decision. To reduce user memory load, we need alternative strategies for sequential presentation. In particular, we require better algorithms for:

1. selecting the most relevant subset of options to mention, as well as the attributes that are most relevant to choosing among them; and
2. determining how to organize and express the descriptions of the selected options and attributes, in ways that are both easy to understand and memorable.[1]

In this article, we describe how we have addressed these points in the FLIGHTS[2] system, reviewing and extending the description given in Moore et al. (2004). FLIGHTS follows previous work (Carberry, Chu-Carroll, and Elzer 1999; Carenini and Moore 2000; Walker et al. 2002) in applying decision-theoretic models of user preferences to the generation of tailored descriptions of the most relevant available options. Multi-attribute decision theory provides a detailed account of how models of user preferences can be used in decision making (Edwards and Barron 1994). Such preference models have been shown to enable systems to present information in ways that are concise and

1 An issue we do not address in this article is whether a multimodal system would be more effective than a voice-only one. We believe that these needs, and in particular the need to express information with contextually appropriate prosody, are also highly relevant for multimodal systems. We also note that there is still strong demand for voice-oriented systems for eyes-busy use and for the blind.

2 FLIGHTS stands for Fancy Linguistically Informed Generation of Highly Tailored Speech.

tailored to the user's interests (Carenini and Moore 2001; Walker et al. 2004; Carenini and Moore 2006). Decision-theoretic models have also been commercially deployed in Web systems.[3]

To present multiple options, we introduce a novel strategy where the best option (with respect to the user model) is presented first, followed by the most compelling remaining options, in terms of trade-offs between attributes that are important to the user. (Multiple options are selected only when each offers a compelling trade-off.) An important property of this strategy is that it naturally lends itself to the determination of information structure, as well as the contents of referring expressions. Thus, to help make the trade-offs among the selected options clear to the user, FLIGHTS (1) groups attributes that are positively and negatively valued for the user, (2) chooses referring expressions that highlight the salient distinguishing attributes, (3) determines information structure and prosodic structure that express contrasts intelligibly, and (4) synthesizes utterances with a unit selection voice that takes the prosodic structure into account. As such, FLIGHTS goes beyond previous systems in adapting its output according to user preferences at all levels of the generation process, not just at the levels of content selection and text planning.

Our approach to generating contextually appropriate intonation follows Prevost (1995) and Steedman (2000a) in using Combinatory Categorial Grammar (CCG) to convey the information structure of sentences via pitch accents and edge tones. To adapt Prevost and Steedman's approach to FLIGHTS, we operationalize the information structural notion of **theme** to correspond to implicit questions that necessarily arise in presenting the trade-offs among options. We also refine the way in which prosodic structure is derived from information structure by allowing for a more flexible, one-to-many mapping between themes or rhemes and intonational phrases, where the final choice of the type and placement of edge tones is determined by n-gram models. To investigate the impact of information structural grammatical constraints in our hybrid rule-based, data-driven approach, we compare realizer outputs with those of baseline n-gram models, and show that the realizer yields target prosodic structures with significantly higher acceptability than the baseline models in an expert evaluation.

The prosodic structure derived during surface realization is passed as prosodic markup to the speech synthesizer. The synthesizer uses this prosodic markup in the text analysis phase of synthesis in place of the structures that it would otherwise have to predict from the text. The synthesizer then uses the context provided by the markup to enforce the selection of suitable units from the database. To verify that the prosodic markup yields improvements in the quality of synthetic speech, we present an experiment which shows that listeners perceive a unit selection voice that aims to produce the target prosodic structures as significantly more natural than either of two baseline unit selection voices that do not use the markup. We also present an expert evaluation and f0 analysis which confirm the superiority of the generator-driven intonation and its contribution to listeners' ratings.

The remainder of this article is structured as follows. Section 2 presents our approach to natural language generation (NLG) in the information presentation phase of a FLIGHTS dialogue, including how multi-attribute decision models are used in content selection; how rhetorical and information structure are determined during discourse planning; how lexical choice and referring expressions are handled in sentence planning; how prosodic structures are derived in surface realization; and how these

3 See `http://www.cogentex.com/solutions/recommender/index.shtml`, for example.

User	Output
S	*There's a direct flight on BMI with a good price. It arrives at four ten p.m. and costs a hundred and twelve pounds. The cheapest flight is on Ryanair. It arrives at twelve forty-five p.m. and costs just fifty pounds, but it requires a connection in Dublin.*
FF	*There's a KLM flight arriving Brussels at four fifty p.m., but business class is not available and you'd need to connect in Amsterdam. If you want to fly direct, there's a BMI flight that arrives at four ten p.m., but it has no availability in business class either. There are seats in business class on the British Airways flight that arrives at four twenty p.m. It requires a connection in Manchester though.*
BC	*You can fly business class on British Airways, arriving at four twenty p.m., but you'd need to connect in Manchester. There is a direct flight on BMI, arriving at four ten p.m., but it has no availability in business class.*

Figure 2
Tailored descriptions of the available flights for three different user models.

prosodic structures compare to those predicted by baseline *n*-gram models in an expert evaluation. Section 3 describes how the prosodic structures are used in the unit selection voice employed in the present study, and compares this voice to the baseline ones used in our perception experiment. Section 4 provides the methods and results of the perception experiment itself, along with the expert prosody evaluation and f0 analysis. Section 5 compares our approach to related work. Finally, Section 6 concludes with a summary and discussion of remaining issues.

2. NLG in FLIGHTS

2.1 Tailoring Flight Descriptions

To illustrate how decision-theoretic models of user preferences can be used to tailor descriptions of the available options at many points in the generation process, let us consider the following three hypothetical users of the FLIGHTS system:

Student (S) A student who cares most about price, all else being equal.
Frequent Flyer (FF) A business traveler who prefers business class, but cares most about building up frequent-flyer miles on KLM.
Business Class (BC) Another business traveler who prefers KLM, but wants, above all, to travel in business class.

Suppose that each user is interested in flying from Edinburgh to Brussels on a certain day, and would like to arrive by five o'clock in the afternoon. FLIGHTS begins the dialogue by gathering the details necessary to query the database for possible flights. Next, it uses the preferences encoded in the user model to select the highest ranked flight for each user, as well as those flights that offer interesting trade-offs. These flights are then described to the user, as shown in Figure 2.[4]

For the student (S), the BMI flight is rated most highly, because it is a fairly inexpensive, direct flight that arrives near the desired time. The Ryanair flight is also

4 The set of available flights was obtained by "screen scraping" data from several online sources. The hypothetical user preferences were chosen with an eye towards making the example interesting. Actual user preferences are specified as part of registering to use the FLIGHTS system.

mentioned as a possibility, as it has the best price; it ends up ranked lower overall than the BMI flight though, because it requires a connection and arrives well in advance of the desired arrival time. For the KLM frequent flyer (FF), life is a bit more complicated: A KLM flight with a good arrival time is offered as the top choice, even though it is a connecting flight with no availability in business class. As alternatives, the direct flight on BMI (with no business class availability) and the British Airways flight with seats available in business class (but requiring a connection) are described. Finally, for the must-have-business-class traveler (BC), the British Airways flight with business class available is presented first, despite its requiring a connection; the direct flight on BMI is offered as another possibility.

Although user preferences have an immediately apparent impact on content selection and ordering, they also have more subtle effects on many aspects of how the selected content is organized and expressed, as explained subsequently.

Referring expressions: Rather than always referring to the available flights in the same way, flights of interest are instead described using the attributes most relevant to the user: for example, *direct flight*, *cheapest flight*, *KLM flight*.

Aggregation: For conciseness, multiple attributes may be given in a single sentence, subject to the constraint that attributes whose values are positive (or negative) for the user should be kept together. For example, in *There's a KLM flight arriving Brussels at four fifty p.m., but business class is not available and you'd need to connect in Amsterdam*, the values of the attributes `airline` and `arrival-time` are considered good, and thus are grouped together to contrast with the values of the attributes `fare-class` and `number-of-legs`, which are considered bad.

Scalar terms: Scalar modifiers like *good*, as in *good price*, and *just*, as in *just fifty pounds*, are chosen to characterize an attribute's value to the user relative to values of the same attribute for other options.

Discourse cues: Attributes with negative values for the user are acknowledged using discourse cues, such as *but* and *though*. Interesting trade-offs can also be signaled using cues such as *if*-conditionals.

Information structure and prosody: Compelling trade-offs are always indicated via prosodic phrasing and emphasis, as in *(There* ARE *seats in business class)*$_{theme}$ *(on the British Airways flight)*$_{rheme}$ *(that arrives at four twenty p.m.)*$_{rheme}$, where the division of the sentence into theme and rheme phrases is shown informally using parentheses, and contrastive emphasis (on ARE) is shown using small caps. (See Section 2.6.2 for details of how emphasis and phrasing are realized by pitch accents and edge tones.)

2.2 Architecture

The architecture of the FLIGHTS generator appears in Figure 3. OAA (Martin, Cheyer, and Moran 1999) serves as a communications hub, with the following agents responsible for specific tasks: DIPPER (Bos et al. 2003) for dialogue management; a Java agent that implements an additive multi-attribute value function (AMVF), a decision-theoretic model of the user's preferences (Carenini and Moore 2000, 2006), for user modeling; OPlan (Currie and Tate 1991) for content planning; Xalan XSLT[5] and OpenCCG (White

5 `http://xml.apache.org/xalan-j/`.

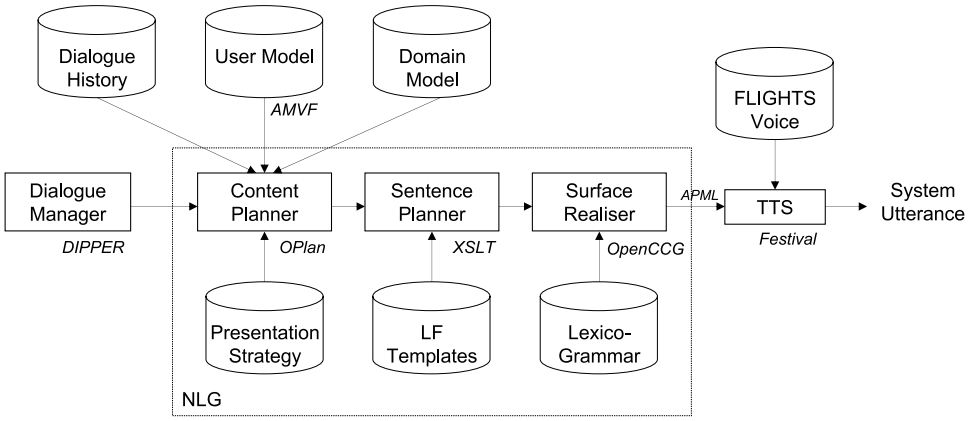

Figure 3
FLIGHTS generation architecture.

2004, 2006a, 2006b) for sentence planning and surface realization; and Festival (Taylor, Black, and Caley 1998) for speech synthesis. The user modeling, content planning, sentence planning, and surface realization agents are described in the ensuing subsections.

FLIGHTS follows a typical pipeline architecture (Reiter and Dale 2000) for NLG. The NLG subsystem takes as input an abstract communicative goal from the dialogue manager. In the information presentation phase of the dialogue, this goal is to describe the available flights that best meet the user's constraints and preferences. Given a communicative goal, the content planner selects and arranges the information to convey by applying the plan operators that implement its presentation strategy. In so doing, it makes use of three further knowledge sources: the user model, the domain model, and the dialogue history. Next, the content plan is sent to the sentence planner, which uses XSLT templates to perform aggregation, lexicalization, and referring expression generation. The output of sentence planning is a sequence of logical forms (LFs). The use of LF templates represents a practical and flexible way to deal with the interaction of decisions made at the sentence planning level, and further blurs the traditional distinction between template-based and "real" NLG that van Deemter, Krahmer, and Theune (2005) have called into question. Each LF is realized as a sentence using a CCG lexico-grammar (Steedman 2000a, 2000b). Note that in contrast to the generation architectures of, for example, Pan, McKeown, and Hirschberg (2002) and Walker, Rambow, and Rogati (2002), the prosodic structure of the sentence is determined as an integral part of surface realization, rather than in a separate prosody prediction component. The prosodic structure is passed to the Festival speech synthesizer using Affective Presentation Markup Language (de Carolis et al. 2004; Steedman 2004), or APML, an XML markup language for the annotation of affect, information structure, and prosody. Festival uses the Tones and Break Indices (Silverman et al. 1992), or ToBI,[6] pitch accents and edge tones—specified as APML annotations—in determining utterance phrasing and intonation, and employs a custom synthetic voice to produce the system utterances.

6 See `http://www.ling.ohio-state.edu/~tobi/` for an introduction to ToBI and links to on-line resources.

2.3 User Modeling

FLIGHTS uses an additive multi-attribute value function (AMVF) to represent the user's preferences, as in the GEA real estate recommendation system (Carenini and Moore 2000, 2006) and the MATCH restaurant recommendation system (Walker et al. 2004). Decision-theoretic models of this kind are based on the notion that, if anything is valued, it is valued for multiple reasons, where the relative importance of different reasons may vary among users.

The first step is to identify good flights for a particular origin, destination, and arrival or departure time. The following attributes contribute to this objective: `arrival-time`, `departure-time`, `number-of-legs`, `total-travel-time`, `price`, `airline`, `fare-class`, and `layover-airport`. As in MATCH, these attributes are arranged into a one-level tree.

The second step is to define a value function for each attribute. A value function maps from the features of a flight to a number between 0 and 1, representing the value of that flight for that attribute, where 0 is the worst and 1 is the best. For example, the function for `total-travel-time` computes the difference in minutes between the flight's arrival and departure times, and then multiplies the result by a scaling factor to obtain an evaluation between 0 and 1. The functions for the `airline`, `layover-airport`, and `fare-class` attributes make use of user-specified preferred or dispreferred values for that attribute. In the current version of these functions, a preferred value is given a score of 0.8, a dispreferred value 0.2, and all other values 0.5.[7]

The structure and weights of the user model represent a user's **dispositional** biases about flight selection. **Situational** features are incorporated in two ways. The requested origin and destination are used as a filter when selecting the set of available options by querying the database. In contrast, the requested arrival or departure time—if specified—is used in the corresponding attribute's evaluation function to give a higher score to flights that are closer to the specified time. If an arrival or departure time is not specified, the corresponding attribute is disabled in the user model.

As in previous work, the overall evaluation of an option is computed as the weighted sum of its evaluation on each attribute. That is, if f represents the option being evaluated, N is the total number of attributes, and w_i and v_i are, respectively, the weight and the value for attribute i, then the evaluation $v(f)$ of option f is computed as follows:

$$v(f) = \sum_{i=1}^{N} w_i v_i(f)$$

To create a user model for a specific user, two types of information are required. The user must rank the attributes in order of importance, and he or she must also specify any preferred or dispreferred attribute values for the `airline`, `layover-airport`, and `fare-class` attributes. In FLIGHTS, we also allow users to specify a partial ordering of the rankings, so that several attributes can be given equal importance when registering to use the system. Figure 4 shows the user models for the student (S), frequent-flyer (FF), and business-class (BC) users discussed earlier; because no departure time is specified in the sample query, `departure-time` is not included in these examples.

7 In informal experiments, we did not find the system to be particularly sensitive to the exact values used in the attribute functions when selecting content.

	Weights						
	Arrival	**# Legs**	**Time**	**Price**	**Airline**	**Layover**	**Class**
S	.1049	.1049	.1049	***.3704***	.1049	.1049	.1049
FF	.1641	.1641	.0728	.0323	***.3704***	.0323	.1641
BC	.1641	.1641	.1641	.0323	.0728	.0323	***.3704***

	Preferences		
	Airline	**Layover**	**Class**
S	—	—	+economy
FF	+KLM	−LHR	+business
BC	+KLM	−LHR	+business

Figure 4
Sample user models.

Based on the user's ranking of the attributes, weights are assigned to each attribute. As in previous work, we use Rank Order Centroid (ROC) weights (Edwards and Barron 1994). This allows weights to be assigned based on rankings, guaranteeing that the sum will be 1. The n^{th} ROC weight w_n^R of N total weights is computed as follows:

$$w_n^R = \frac{1}{N}\sum_{i=n}^{N}\frac{1}{i}$$

We extend these initial weights to the partial-ordering case as follows. If attributes $i \ldots j$ all have the same ranking, then the weight of each will be the mean of the relevant ROC weights; that is,

$$(\sum_{k=i}^{j} w_k^R)/(j-i+1)$$

As a concrete example, if there is a single highest-ranked attribute followed by a three-way tie for second, then $w_1 = w_1^R$, and $w_2 = w_3 = w_4 = \frac{1}{3}(w_2^R + w_3^R + w_4^R)$.

2.4 Content Planning

2.4.1 Content Selection. Once a specific user model has been created, the AMVF can be used to select a set of flights to describe for that user, and to determine the features of those flights that should be included in the descriptions. We use a novel strategy that combines features of the **Compare** and **Recommend** strategies of Walker et al. (2004), refining them with an enhanced method of selecting options to mention. In brief, the idea behind the strategy is to select the top-ranked flight, along with any other highly ranked flights that offer a compelling trade-off—that is, a better value (for the user) on one of its attributes. As we shall see in Section 2.4.2, by guaranteeing that any option beyond the first one offers such a trade-off, our strategy lends itself naturally to the determination of information structure and the contents of referring expressions identifying the option. By contrast, Walker et al.'s Recommend strategy only presents a single option, and their Compare strategy does not present options in a ranked order that facilitates making trade-offs salient. In addition, we may observe that with our strategy, the user model need only contain a rough approximation of the user's true

Step 1: Set $Sel := \{f_0\}$, where f_0 is the top-ranked option.

Step 2: For each flight f, in decreasing order of evaluation (after f_0):

Step 2a: If $z(f) < k_z$, stop.

Step 2b: Otherwise, for each attribute a in the user model: If $\forall g \in Sel, a \in comp(f, g, k_c)$, set $Sel := Sel \cup \{f\}$ and continue.

Figure 5
Algorithm for selecting the options to describe.

preferences in order for it to do its job of helping to identify good flights for the user to consider.[8] A similar observation underlies the **candidate/critique** model of Linden, Hanks, and Lesh's (1997) Web-based system.

Selecting the Options to Describe. In determining whether an option is worth mentioning, we make use of two measures. Firstly, we use the z-score of each option; this measures how far the evaluation $v(f)$ of an option f is from the mean evaluation. Formally, it is defined using the mean (μ_V) and standard deviation (σ_V) of all evaluations, as follows:

$$z(f) = (v(f) - \mu_V)/\sigma_V$$

We also make use of the **compellingness** measure described by Carenini and Moore (2000, 2006), who provide a formal definition. Informally, the compellingness of an attribute measures its strength in contributing to the overall difference between the evaluation of two options, all other things being equal. For options f, g, and threshold value k_c, we define the set $comp(f, g, k_c)$ as the set of attributes that have a higher score for f than for g, and for which the compellingness is above k_c.

The set *Sel* of options to describe is constructed as follows. First, we include the top-ranked option. Next, for all of the other options whose z-score is above a threshold k_z, we check whether there is an attribute of that option that offers a compelling trade-off over the already selected options; if so, we add that option to the set.[9] This algorithm is presented in Figure 5.

For the BC user model, for example, this algorithm proceeds as follows. First, it selects the top-ranked flight: a connecting flight on British Airways with availability in business class. The next-highest-ranked flight is a morning flight, which does not have any attributes that are compellingly better than those of the top choice, and is therefore skipped. However, the third option presents an interesting trade-off: even though business class is not available, it is a direct flight, so it is also included. None of the other options above the threshold present any interesting trade-offs, so only those two flights are included.

8 Of course, if the user model could be relied upon to contain perfect information, the system could always just recommend a single best flight; however, because we do not expect our models to capture a user's preferences perfectly, we have designed the system to let the user weigh the alternatives when there appear to be interesting trade-offs among the available options.

9 The requirement that an option offer a compelling trade-off is similar to the exclusion of **dominated** solutions in Linden, Hanks, and Lesh (1997).

The selected flights for the other sample user models show similar trade-offs, as mentioned in the discussion of Figure 2. For FF, the selected flights are a connecting, economy-class flight on the preferred airline; a direct, economy-class flight on a neutral airline; and a connecting, business-class flight on a neutral airline. For S, the top choices are a reasonably cheap direct flight that arrives near the desired time, and an even cheaper, connecting flight that arrives much earlier in the day.

Selecting the Attributes to Include. When selecting the attributes to include in the description, we make use of an additional measure, **s-compellingness**. Informally, the s-compellingness of an attribute represents the contribution of that attribute to the evaluation of a single option; again, the formal definition is given by Carenini and Moore (2000, 2006). Note that an attribute may be s-compelling in either a positive or a negative way. For an option f and threshold k_c, we define the set *s-comp*(f, k_c) as the set of attributes whose s-compellingness for f is greater than k_c.

The set *Atts* of attributes is constructed in two steps. First, we add the most compelling attributes of the top choice. Next, we add all attributes that represent a trade-off between any two of the selected options; that is, attributes that are compellingly better for one option than for another. The algorithm appears in Figure 6.

For the BC user model, the s-compelling attributes of the top choice are `arrival-time` and `fare-class`; the latter is also a compelling advantage of this flight over the second option. The advantage of the second option over the first is that it is direct, so `number-of-legs` is also included. A similar process on the other user models results in `price`, `arrival-time`, and `number-of-legs` being selected for S, and `arrival-time`, `fare-class`, `airline`, and `number-of-legs` for FF.

2.4.2 Planning Texts with Information Structure. Based on the information returned by the content selection process, together with further information from the user model and the current dialogue context, the content planning agent develops a plan for presenting the available options. A distinguishing feature of the resulting content plans is that they contain specifications of the **information structure** of sentences (Steedman 2000a), including sentence **theme** (roughly, the topic the sentence addresses) and sentence **rheme** (roughly, the new contribution on a topic).

Steedman (2000a) characterizes the notions of theme and rheme more formally by stating that a theme presupposes a rheme alternative set, in the sense of Rooth (1992), while a rheme restricts this set. Because Steedman does not fully formalize the discourse update semantics of sentence themes, we have chosen to operationalize themes in FLIGHTS in one particular way, namely, as corresponding to implicit questions that arise in the context of describing the available options. Our reasoning is as follows. First, we note that a set of alternative answers corresponds formally to the meaning of a question. Next, we observe that whenever a flight option presents a compelling trade-off

Let *Sel* be the set of selected options, and f_0 the top-ranked option.

Step 1: Set *Atts* := *s-comp*(f_0, k_c).

Step 2: For all options $f, g \in$ *Sel*: Set *Atts* := *Atts* $\cup$ *comp*(f, g, k_c).

Figure 6
Algorithm for selecting the attributes to include.

for a particular attribute, it at least partially addresses the question of whether there are any flights that have a desirable value for that attribute; moreover, whenever a flight is presented that has a less-than-optimal value for an attribute, its mention implicitly raises the question of whether any flights are available with a better value for that attribute. Finally, we conclude that by specifying and realizing content as thematic, the system can help the user understand why a flight option is being presented, since the theme (by virtue of its presupposition) identifies what implicit question—that is, what trade-off—the option is addressing.

In all, the content planner's presentation strategy performs the following functions:

- marking the status of items as *definite/indefinite* and predications as *theme/rheme* for information structure;
- determining *contrast* between options and attributes, or between groups of options and attributes;
- *grouping* and *ordering* of similar options and attributes (e.g., presenting the top scoring option first vs. last);
- choosing the contents of *referring expressions* (e.g., referring to a particular option by airline); and
- *decomposing* strategies into basic dialogue acts and hierarchically organized rhetorical speech acts.

The presentation strategy is specified via a small set (about 40) of content planning operators. These operators present the selected flights as an ordered sequence of options, starting with the best one. Each flight is suggested and then further described. As part of suggesting a flight, it is identified by its most compelling attribute, according to the user model. (Recall that any selected flight options beyond the highest ranked one must offer a compelling trade-off.) Flights are additionally identified by their `airline`, which we deemed sufficiently significant to warrant special treatment (otherwise the strategy is domain-independent).

The information for the first flight is presented as all rheme, as no direct link is made to the preceding query. Each subsequent, alternative flight is presented with its most compelling attribute in the theme, and the remaining attributes in the rheme. For instance, consider the student example (S) in Figure 2. After presenting the BMI flight—which has a good price, is direct, and arrives near the desired time—the question arises whether there are any cheaper alternatives. Because the second option has `price` as its most compelling attribute—and given that it is the least expensive flight available—it is identified as *the* CHEAPEST *flight*, with this phrase forming the theme of the utterance. As another illustration, consider the business-class traveler example (BC) in Figure 2. After presenting the British Airways flight, which has availability in business class but is not direct, the question arises whether there are any direct flights available. The presentation of the second option addresses this implicit question, introducing the BMI flight with the theme phrase *There is a* DIRECT *flight*.

The output of the content planner is derived from the hierarchical goal structure produced during planning. Figure 7 shows the resulting content plan for the student example. Note that, as part of suggesting the second flight option in the sequence (`f2`), subgoals are introduced that identify the option by informing the user that the option has type `flight` and that the option has the attribute `cheapest`, where this attribute is the most compelling one for the user. Both subgoals are marked as part of the theme

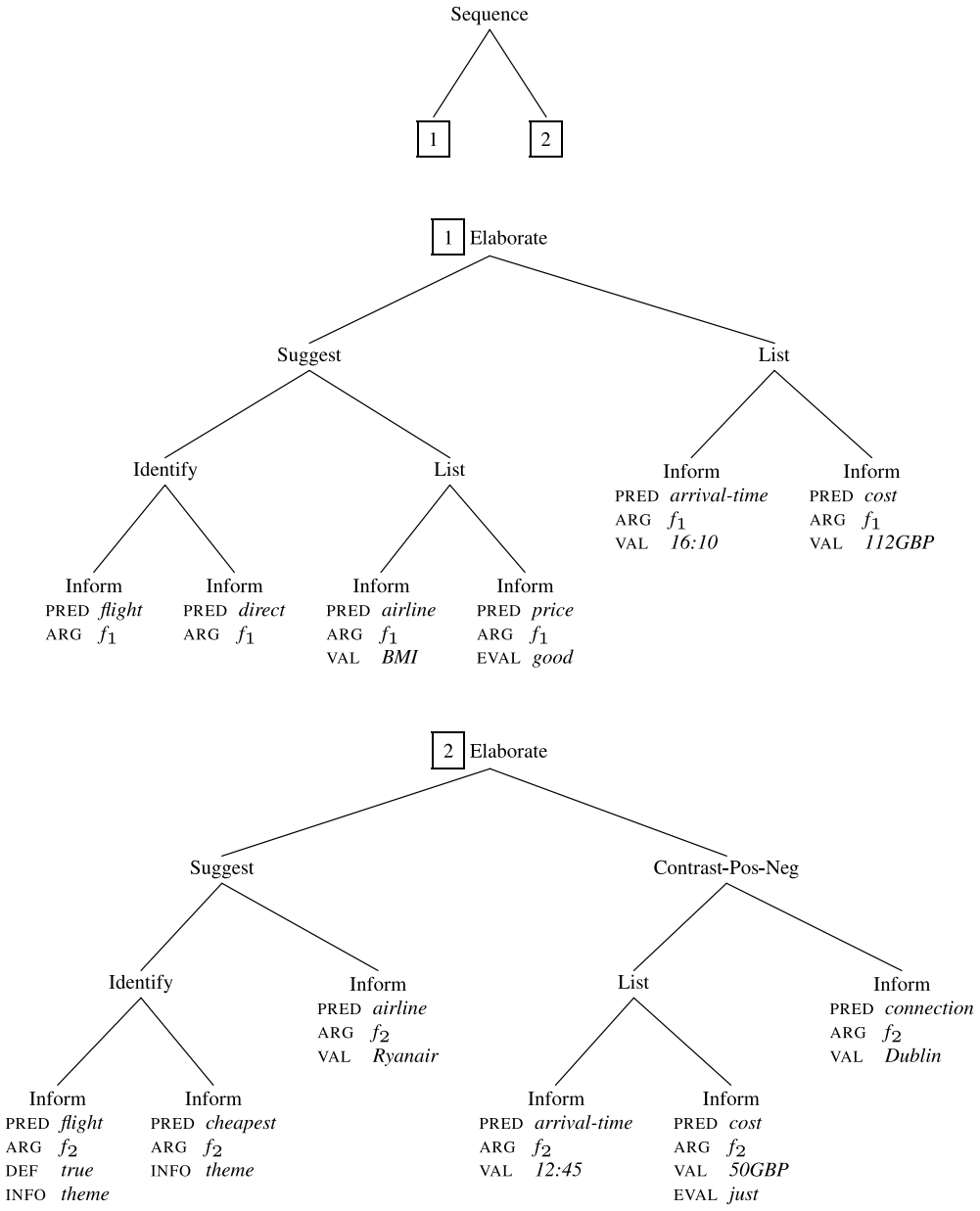

Figure 7
Content plan for student example (S).

(rheme marking is the default); the subgoal for the option type is also marked for definiteness (indefinite is the default), as the `cheapest` attribute uniquely identifies the flight. The remaining information for the second flight is presented in terms of a contrast between its positive and negative attributes, as determined by the user model.

The way in which our presentation strategy uses theme phrases to connect alternative flight suggestions to implicit questions is related to Prevost's (1995) use of theme phrases to link system answers to immediately preceding user questions. It is also

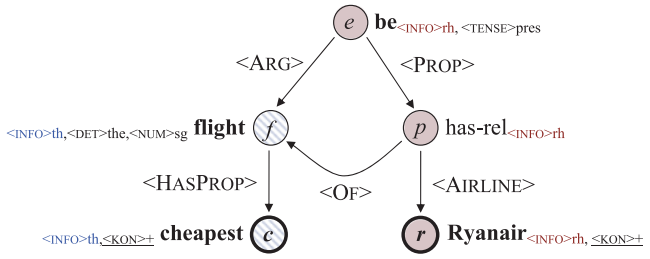

Figure 8
Semantic dependency graph produced by the sentence planner for *(The* CHEAPEST *flight)*$_{theme}$ *(is on* RYANAIR*)*$_{rheme}$.

related to Kruijff-Korbayová et al.'s (2003) use of theme phrases to link utterances with questions under discussion (Ginzburg 1996; Roberts 1996) in an information-state based dialogue system. An interesting challenge that remains for future work is to determine to what extent our presentation strategy can be generalized to handle theme/rheme partitioning, both for explicit questions across turns as well for implicit questions within turns.

2.5 Sentence Planning

The sentence planning agent uses the Xalan XSLT processor to transform the output of the content planner into a sequence of LFs that can be realized by the OpenCCG agent. It is intended to be a relatively straightforward component, as the content planner has been designed to implement the most important high-level generation choices. Its primary responsibility is to lexicalize the basic speech acts in the content plan—which may appear in referring expressions—along with the rhetorical speech acts that connect them together. When alternative lexicalizations are available, all possibilities are included in a packed structure (Foster and White 2004; White 2006a). The sentence planner is also responsible for adding discourse markers such as *also* and *but*, adding pronouns, and choosing sentence boundaries. It additionally implements a small number of rhetorical restructuring operations for enhanced fluency.

The sentence planner makes use of approximately 50 XSLT templates to recursively transform content plans into logical forms. An example logical form that results from applying these templates to the content plan shown in Figure 7 appears in Figure 8 (with alternative lexicalizations suppressed). As described further in Section 2.6.1, the logical forms produced by the sentence planner are semantic dependency structures,[10] which make use of an `info` feature to encode theme/rheme partitioning, and a `kon` feature to implement Steedman's (2006) notion of **kontrast** (Vallduví and Vilkuna 1998). Following Steedman, kontrast is assigned to the interpretations of words which contribute to distinguishing the theme or rheme of the utterance from other alternatives that the context makes available.

In order to trigger the inclusion of context-sensitive discourse markers such as *also* and *either*, the sentence planner compares the `inform` acts for consecutive flight options

10 The nodes of the graph are typically labelled by lexical predicates. An exception here is the **has-rel** predicate, which allows the predicative complement *on Ryanair* to introduce the ⟨AIRLINE⟩ role in a way that is similar to the dependency structure for *the Ryanair flight*.

to see whether acts with the same type have the same value.[11] These same checks can also trigger de-accenting. For example, when two consecutive flights have no seats in business class, the second one can be described using the phrase *it has* NO AVAILABILITY *in business class either*, where *business class* has been de-accented.

The development of the XSLT templates was made considerably easier by the ability to invoke OpenCCG to parse a target sentence and then use the resulting logical form as the basis of a template. (See Section 3 for a description of how the target sentences in the FLIGHTS voice script were developed.) Using LF templates, rather than templates at the string level, makes it simpler to uniformly handle discourse markers such as *also* and *either*, which have different preferred positions within a clause, depending in part on which verb they modify. LF templates also simplify the treatment of subject–verb agreement. Additionally, by employing LF templates with a single theme/rheme partition, it becomes possible to underspecify whether the theme and rheme will be realized by one intonational phrase each or by multiple phrases (see Section 2.6.2). At the same time, certain aspects of the LF templates can be left alone, when there is no need for further analysis and generalization.

As noted earlier, the use of LF templates further blurs the traditional distinction between template-based and "real" NLG that van Deemter, Krahmer, and Theune (2005) have called into question. In the case of referring expressions especially, LF templates in FLIGHTS represent a practical and flexible way to deal with the interaction of decisions made at the sentence planning level, as the speech acts identifying flight options are considered together with the other basic and rhetorical speech acts in the applicability conditions for the templates that structure clauses. In this way, options can be identified not only in definite NPs, such as *the* CHEAPEST *flight*, but also in *there*-existentials and conditionals, such as *there is a* DIRECT *flight on* BMI *that* ... or *if you prefer to fly* DIRECT, *there's a* BMI *flight that* .... We may further observe that the traditional approach to generating referring expressions (Reiter and Dale 2000), where a distinguishing description for an entity is constructed during sentence planning without regard to a user model, would not fit in our architecture, where the user model drives the selection of a referring expression's content at the content planning level.

Although the use of LF templates in XSLT represents a practical approach to handling sentence planning tasks that were not the focus of our research, it is not one that promotes reuse, and thus it is worth noting which aspects of our sentence planner would pose challenges for a more declarative and general treatment. The bulk of the templates concern domain-specific lexicalization and are straightforward; given the way these were developed from the results of OpenCCG parsing, it is conceivable that this process could be largely automated from example input–output pairs. The templates for adding pronouns and discourse markers require more expertise but remain reasonably straightforward; the templates for rhetorical restructuring and choosing sentence boundaries, in contrast, are fairly intricate. In principle, satisfactory results might be obtained using a more general set of options for handling pronouns, discourse markers, and sentence boundaries, together with an overgenerate-and-rank methodology; we leave this possibility as a topic for future research.

2.6 Surface Realization

2.6.1 Chart Realization with OpenCCG. For surface realization, we use the OpenCCG open source realizer (White 2004, 2006a, 2006b). A distinguishing feature of OpenCCG is that

11 In future work, these checks could be extended to apply across turns as well.

it implements a hybrid symbolic-statistical chart realization algorithm that combines (1) a theoretically grounded approach to syntax and semantic composition, with (2) the use of integrated language models for making choices among the options left open by the grammar. In so doing, it brings together the symbolic chart realization (Kay 1996; Shemtov 1997; Carroll et al. 1999; Moore 2002) and statistical realization (Knight and Hatzivassiloglou 1995; Langkilde 2000; Bangalore and Rambow 2000; Langkilde-Geary 2002; Oh and Rudnicky 2002; Ratnaparkhi 2002) traditions. Another recent approach to combining these traditions appears in Carroll and Oepen (2005), where parse selection techniques are incorporated into an HPSG realizer.

Like other realizers, the OpenCCG realizer is partially responsible for determining word order and inflection. For example, the realizer determines that *also* should preferably follow the verb in *There is also a very cheap flight on Air France*, whereas in other cases it typically precedes the verb, as in *I also have a flight that leaves London at 3:45 p.m.* It also enforces subject–verb agreement, for example, between *is* and *flight*, or between *are* and *seats*. Less typically, in FLIGHTS and in the COMIC[12] system, the OpenCCG realizer additionally determines the prosodic structure, in terms of the type and placement of pitch accents and edge tones, based on the information structure of its input logical forms. Although OpenCCG's algorithmic details have been described in the works cited above, details of how prosodic structure can be determined from information structure in OpenCCG appear for the first time in this article.

The grammars used in the FLIGHTS and COMIC systems have been manually written with the aim of achieving very high quality. However, to streamline grammar development, the grammar has been allowed to overgenerate in areas where rules are difficult to write and where *n*-gram models can be reliable; in particular, it does not sufficiently constrain modifier order, which in the case of adverb placement especially can lead to a large number of possible orderings. Additionally, it allows for a one-to-many mapping from themes or rhemes to edge tones, yielding many variants that differ only in boundary type or placement. As will be explained subsequently, we consider this more flexible, data-driven approach to phrasing to be better suited to the needs of generation than would be a more direct implementation of Steedman's (2000a) theory, which would require all phrasing choices to be made at the sentence planning level.

We have built a language model for the system from the FLIGHTS speech corpus described in Section 3.1. To enhance generalization, named entities and scalar adjectives have been replaced with the names of their semantic classes (such as TIME, DATE, CITY, AIRLINE, etc.), as is often done in limited domain systems. Note that in the corpus and the model, pitch accents are treated as integral parts of words, whereas edge tones and punctuation marks appear as separate words. The language model is a 4-gram back-off model built with the SRI language modeling toolkit (Stolcke 2002), keeping all 1-counts and using Ristad's (1995) natural discounting law for smoothing. OpenCCG has its own implementation for run-time scoring; in FLIGHTS, the model additionally incorporates an *a/an*-filter, which assigns a score of zero to sequences containing *a* followed by a vowel, or *an* followed by a consonant, subject to exceptions culled from bigram counts.

2.6.2 Deriving Prosody. CCG is a unification-based categorial framework that is both linguistically and computationally attractive. We provide here a brief overview of CCG; an extensive introduction appears in Steedman (2000b).

12 http://www.hcrc.ed.ac.uk/comic/.

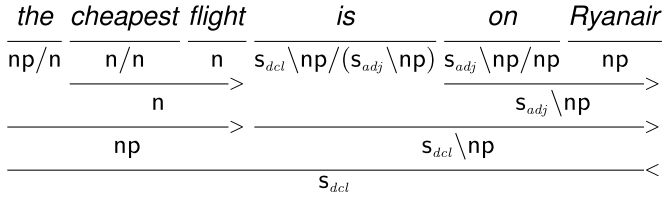

Figure 9
A simple CCG derivation.

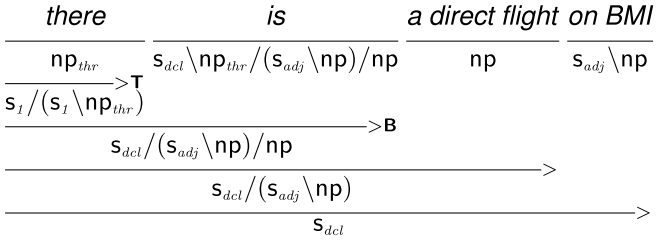

Figure 10
A CCG derivation with non-standard constituents.

A grammar in CCG is defined almost entirely in terms of the entries in the lexicon, which are (possibly complex) categories bearing standard feature information (such as verb form, agreement, etc.) and subcategorization information. CCG has a small set of rules which can be used to combine categories in derivations. The two most basic rules are forward (>) and backward (<) function application. These rules are illustrated in Figure 9, which shows the derivation of a simple copular sentence (with no prosodic information).[13] In the figure, the noun and proper name receive atomic categories with labels n and np, respectively. The remaining words receive functional categories, such as $s_{dcl}\backslash np/(s_{adj}\backslash np)$ for the verb *is*; this category seeks a predicative adjective ($s_{adj}\backslash np$) to its right and an np to its left, and returns the category s_{dcl}, for a declarative sentence. Note that *dcl* and *adj* are values for the `form` feature; other features, such as those for number and case, have been suppressed in the figure (as has the feature label, `form`).

CCG also employs further rules based on the composition (**B**), type raising (**T**), and substitution (**S**) combinators of combinatory logic. These rules add an element of associativity to the grammar, making possible multiple derivations with the same semantics. They are crucial for building the "non-standard" constituents that are the hallmark of categorial grammars, and which are essential for CCG's handling of coordination, extraction, intonation, and other phenomena. For example, Figure 10 shows how the category for *there* can be type-raised and composed with the category for *is* in order to derive the constituent *there is a direct flight* ($s_{dcl}/(s_{adj}\backslash np)$), which cuts across the VP in traditional syntax. Because *there is a direct flight* corresponds to the theme phrase, and *on BMI* to the rheme phrase, in the first clause of the second sentence in

13 These derivations are shown from the parsing perspective, starting with an ordered sequence of words at the top. During realization, it is the semantics rather than the word sequence which is given; the word sequence is determined during the search for a derivation that covers the input semantics. See the discussion of Figure 11 (later in this section) for a discussion of the semantic representations used in OpenCCG.

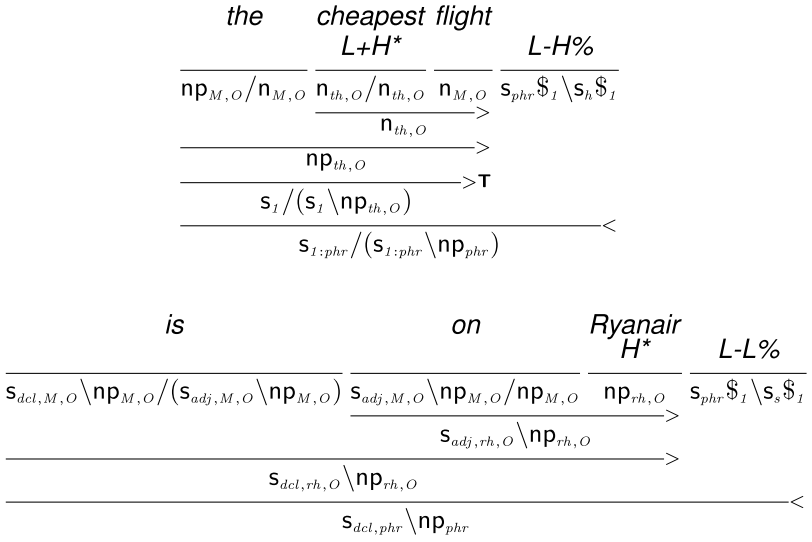

Figure 11
A derivation of theme and rheme phrases.

the business-class (BC) example in Figure 2, the derivation in Figure 10 also shows how CCG's flexible notion of constituency allows the intonation structure and information structure to coincide, where the intonation coincides with surface structure, and the information structure is built compositionally from the constituent analysis.

In Steedman (2000a), theme and rheme tunes are characterized by distinct patterns of pitch accents and edge tones. The notation for pitch accents and edge tones is taken from Pierrehumbert (1980) and ToBI (Silverman et al. 1992). We have implemented a reduced version of Steedman's theory in OpenCCG, where theme tunes generally consist of one or more L+H* pitch accents followed by a L-H% compound edge tone at the end of the phrase,[14] and rheme tunes consist of one or more H* pitch accents followed by a L- or L-L% boundary at phrase end. Additionally, *yes/no* questions typically receive a H-H% final boundary, and L-H% boundaries are often used as continuation rises to mark the end of a non-final conjoined phrase.

The key implementation idea in Steedman's (2000a) approach is to use features on the syntactic categories to enforce information structural phrasing constraints—that is, to ensure that the intonational phrases are consistent with the theme/rheme partition in the semantics. For example, if *there is a direct flight* corresponds to the theme and *on BMI* the rheme, then the syntactic features ensure that the intonation brackets the clause as *(there is a direct flight) (on BMI)*, rather than *(there) (is a direct flight on BMI)* or *(there is) (a direct flight on BMI)*, etc. To illustrate, Figure 11 shows, again from the parsing perspective, how theme and rheme phrases are derived in OpenCCG for the subject NP and VP of Figure 9, respectively. (To save space, pitch accents and edge tones will henceforth be written using subscripts and string elements, rather than appearing on a separate tonal tier.) In the figure, the category for *cheapest* has the `info` feature on each

14 To reduce ambiguity in the grammar, unmarked themes—that is, constituents which appear to be thematic but which are not grouped intonationally into a separate phrase—are assumed to be incorporated as background parts of the rheme, as suggested in Calhoun et al. (2005).

atomic category set to *th(eme)*, and *Ryanair* has its `info` feature set to *rh(eme)*, due to the presence of the L+H* and H* accents, respectively. The remaining words have no accents, and thus their categories have a variable (*M*, for *EME*) as the value of the `info` feature on each atomic category.[15] All the words also have a variable (*O*) as the value of the `owner` feature, discussed subsequently, on each atomic category. As words combine into phrases, the `info` variables serve to propagate the theme or rheme status of a phrase; thus, the phrase *the cheapest*$_{L+H*}$ *flight* has category $\mathsf{np}_{th,O}$, whereas *is on Ryanair*$_{H*}$ ends up with category $\mathsf{s}_{dcl,rh,O}\backslash\mathsf{np}_{rh,O}$. Because these two phrases have incompatible `info` values, they cannot combine before being "promoted" to intonational phrases by their respective edge tones. In this way, the constraint that phrasing choices respect the theme/rheme partition is enforced.

The edge tones allow complete intermediate or intonational phrases to combine by changing the value of the `info` feature to *phr(ase)* in the result category.[16] Note that the argument categories of the edge tones do not select for a particular value of the `info` feature; instead, L-H% has $\mathsf{s}_h\$_1$ as its argument category, whereas L-L% has $\mathsf{s}_s\$_1$, where *h(earer)* and *s(peaker)* are the respective values of the `owner` feature.[17] Combination with an edge tone unifies the `owner` features throughout the phrase. In a prototypical theme phrase, the hearer is the owner, whereas in a rheme phrase, the speaker is the owner.

Like other compositional grammatical frameworks, CCG allows logical forms to be built in parallel with the derivational process. Traditionally, the λ-calculus has been used to express semantic interpretations, but OpenCCG instead makes use of a more flexible representational framework, Hybrid Logic Dependency Semantics (Baldridge and Kruijff 2002; Kruijff 2003), or HLDS. In HLDS, hybrid logic (Blackburn 2000) terms are used to describe semantic dependency graphs, such as the one seen earlier in Figure 8. As discussed in White (2006b), HLDS is well suited to the realization task, in that it enables an approach to semantic construction that ensures semantic monotonicity, simplifies equality tests, and avoids copying in coordinate constructions.

To illustrate, four lexical items from Figure 11 appear in Example (1). The three words are derived by a lexical rule which adds pitch accents and information structure features to the base forms. The format of the entries is *lexeme* ⊢ *category*, where the category is itself a pair in the format *syntax* : *logical form*. The logical form is a conjunction of elementary predications (EPs), which come in three varieties: lexical predications, such as $@_E\mathbf{be}$; semantic features, such as $@_E\langle\text{TENSE}\rangle\text{pres}$; and dependency relations, such as $@_E\langle\text{ARG}\rangle X$.

(1) a. *cheapest*$_{L+H*}$ ⊢ $\mathsf{n}_{X,th,O}/\mathsf{n}_{X,th,O} : @_X\langle\text{HASPROP}\rangle P \wedge @_P\mathbf{cheapest} \wedge$
$@_P\langle\text{INFO}\rangle\text{th} \wedge @_P\langle\text{OWNER}\rangle O \wedge @_P\langle\text{KON}\rangle+$

b. *Ryanair*$_{H*}$ ⊢ $\mathsf{np}_{X,rh,O} : @_X\mathbf{Ryanair} \wedge$
$@_X\langle\text{INFO}\rangle\text{rh} \wedge @_X\langle\text{OWNER}\rangle O \wedge @_X\langle\text{KON}\rangle+$

15 In practice, variables like *M* are replaced with "fresh" variables (such as *M_1*) during lexical lookup, so that the *M* variables are distinct for different words.

16 The $ variables range over a (possibly empty) stack of arguments, allowing the edge tones to employ a single $\mathsf{s}\$_1\backslash\mathsf{s}\$_1$ category to combine with $\mathsf{s}/(\mathsf{s}\backslash\mathsf{np})$, $\mathsf{s}\backslash\mathsf{np}$, and so on. As indicated in the figure, the new value of the `info` feature (*phr*) is automatically distributed throughout the arguments of the result category.

17 See Steedman (2000a) for a discussion of this notion of "ownership," or responsibility.

c. *is* $\vdash$ $\mathsf{s}_{E,dcl,M,O}\backslash\mathsf{np}_{X,M,O}/(\mathsf{s}_{P,adj,M,O}\backslash\mathsf{np}_{X,M,O}) : @_E\mathbf{be} \wedge$
$@_E\langle\textsc{tense}\rangle\mathrm{pres} \wedge @_E\langle\textsc{arg}\rangle X \wedge @_E\langle\textsc{prop}\rangle P \wedge$
$@_E\langle\textsc{info}\rangle M \wedge @_E\langle\textsc{owner}\rangle O \wedge @_E\langle\textsc{kon}\rangle -$

d. *L-H%* $\vdash$ $\mathsf{s}_{phr}\$_1\backslash\mathsf{s}_h\$_1$

In these entries, the indices (or **nominals**, in hybrid logic terms) in the logical forms, such as X, P, and E, correspond to nodes in the semantic graph structure, and are linked to the syntactic categories via the `index` feature. Similarly, the syntactic features `info` and `owner` have associated $\langle\textsc{info}\rangle$ and $\langle\textsc{owner}\rangle$ features in the semantics. As discussed previously, in derivations the values of the `info` and `owner` features are propagated throughout an intonational phrase, which has the effect of propagating the values of the $\langle\textsc{info}\rangle$ and $\langle\textsc{owner}\rangle$ semantic features to every node in the dependency graph corresponding to the phrase. In this way, a distributed representation of the theme/rheme partition is encoded, in a fashion reminiscent of Kruijff's (2003) approach to representing topic–focus articulation using hybrid logic. By contrast, the $\langle\textsc{kon}\rangle$ feature (cf. Section 2.5) is a purely local one, and thus appears only in the semantics. Note that because the edge tones do not add any elementary predications, one or more edge tones—and thus one or more intonational phrases—may be used to derive the same theme or rheme in the logical form.

To simplify the semantic representations the sentence planner must produce, OpenCCG includes default rules that (where applicable) propagate the value of the $\langle\textsc{info}\rangle$ feature to subtrees in the logical form, set the $\langle\textsc{owner}\rangle$ feature to its prototypical value, and set the value of the $\langle\textsc{kon}\rangle$ feature to false. When applied to the logical form for the semantic dependency graph in Figure 8, the rules yield the HLDS term in Example (2). After this logical form is flattened to a conjunction of EPs, lexical instantiation looks up relevant lexical entries, such as those in Example (1), and instantiates the variables to match those in the EPs. The chart-based search for complete realizations proceeds from these instantiated lexical entries.

(2) $@_e(\mathbf{be} \wedge \langle\textsc{tense}\rangle\mathrm{pres} \wedge \langle\textsc{info}\rangle\mathrm{rh} \wedge \langle\textsc{owner}\rangle\mathrm{s} \wedge \langle\textsc{kon}\rangle- \wedge$
$\langle\textsc{arg}\rangle(f \wedge \mathbf{flight}) \wedge \langle\textsc{det}\rangle\mathrm{the} \wedge \langle\textsc{num}\rangle\mathrm{sg} \wedge \langle\textsc{info}\rangle\mathrm{th} \wedge \langle\textsc{owner}\rangle\mathrm{h} \wedge \langle\textsc{kon}\rangle- \wedge$
$\langle\textsc{HasProp}\rangle(c \wedge \mathbf{cheapest} \wedge \langle\textsc{info}\rangle\mathrm{th} \wedge \langle\textsc{owner}\rangle\mathrm{h} \wedge \langle\textsc{kon}\rangle+)) \wedge$
$\langle\textsc{prop}\rangle(p \wedge \mathbf{has\text{-}rel} \wedge \langle\textsc{info}\rangle\mathrm{rh} \wedge \langle\textsc{owner}\rangle\mathrm{s} \wedge \langle\textsc{kon}\rangle- \wedge$
$\langle\textsc{of}\rangle f \wedge$
$\langle\textsc{airline}\rangle(r \wedge \mathbf{Ryanair} \wedge \langle\textsc{info}\rangle\mathrm{rh} \wedge \langle\textsc{owner}\rangle\mathrm{s} \wedge \langle\textsc{kon}\rangle+)))$

Given our focus in FLIGHTS on using intonation to express contrasts intelligibly, we have chosen to employ hard constraints in the OpenCCG grammar on the choice of pitch accents and on the use of edge tones to separate theme and rheme phrases. However, the grammar only partially constrains the type of edge tone (as explained previously), and allows the theme and rheme to be expressed by one or more intonational phrases each; consequently, the final choice of the type and placement of edge tones is determined by the n-gram model. To illustrate, consider Example (3), which shows how the frequent flyer sentence seen in Figure 2 is divided into four intonational phrases. Other possibilities (among many) allowed by the grammar include leaving out the L-L% boundary between *flight* and *arriving*, which would yield a phrase that's likely to be perceived as too long, or adding a L- or L-L% boundary between *there's* and *a*, which would yield two unnecessarily short phrases.

(3) There's a KLM_{H*} flight L-L% arriving $\mathrm{Brussels}_{H*}$ at four_{H*} fifty_{H*} $\mathrm{p.m.}_{H*}$ L-L%, but $\mathrm{business}_{H*}$ class_{H*} is not_{H*} $\mathrm{available}_{H*}$ L-H% and you'd need to $\mathrm{connect}_{H*}$ in $\mathrm{Amsterdam}_{H*}$ L-L%.

To allow for this flexible mapping between themes and rhemes and one or more intonational phrases, we take advantage of our distributed approach to representing the theme/rheme partition, where edge tones mark the ends of intonational phrases without introducing their own elementary predications. As an alternative, we could have associated a theme or rheme predication with the edge tones, which would be more in line with Steedman's (2000a) approach. However, doing so would make it necessary to include one such predication per phrase in the logical forms, thereby anticipating the desired number of output theme and rheme phrases in the realizer's input. Given that the naturalness of intonational phrasing decisions can depend on surface features like phrase length, we consider the distributed approach to representing theme/rheme status to be better suited to the needs of generation.

2.7 Comparison to Baseline Prosody Prediction Models

As noted in Section 2.2, spoken language dialogue systems often include a separate prosody prediction component (Pan, McKeown, and Hirschberg 2002; Walker, Rambow, and Rogati 2002), rather than determining prosodic structure as an integral part of surface realization, as we do here. Although it is beyond the scope of this article to compare our approach to a full-blown, machine learning–based prosody prediction model, we do present in this section an expert evaluation that shows that our approach outperforms strong baseline *n*-gram models. In particular, we show that the information structural constraints in the grammar play an important role in producing target prosodic boundaries, and that these boundary choices are preferred to those made by an *n*-gram model in isolation.

According to Pan, McKeown, and Hirschberg (2002, page 472), word-based *n*-gram models can be surpringly good: "The word itself also proves to be a good predictor for both accent and break index prediction. . . . Since the performance of this [word] model is the best among all the [single-feature] accent prediction models investigated, it seems to suggest that for a CTS [Concept-to-Speech] application created for a specific domain, features like word can be quite effective in prosody prediction." Indeed, although their best accent prediction model exceeded the word-based one, the difference did not reach statistical significance (page 485). Additionally, word predictability, measured by the log probability of bigrams and trigrams, was found to significantly correlate with pitch accent decisions (pages 482–483), and contributed to their best machine-learned models for accent presence and boundary presence.[18]

As baseline *n*-gram models, we trained 1- to 4-gram models for predicting accents and boundaries using the FLIGHTS speech corpus (Section 3.1), the same corpus used to train the realizer's language model. The baseline *n*-gram models are factored language models (Bilmes and Kirchhoff 2003), with words, accents, and boundaries as factors. The accent models have accent as the child variable and 1–4 words as parent variables, starting with the current word, and including up to three previous words. The boundary models are analogous, with boundary as the child variable. With these models, each maximum likelihood prediction of pitch accent or edge tone is independent of all other choices, so there is no need to perform a best-path search. The majority baseline predicts no accent and no edge tone for each word.

18 Note that in our setting, it would be impossible to exactly replicate Pan et al.'s models, in which syntactic boundaries play an important role, as CCG does not have a rigid notion of syntactic boundary.

Table 1
Baseline pitch accent and boundary prediction accuracy against target tunes.

	Accuracy					
	Majority	1-gram	2-gram	3-gram	4-gram	N
Accent Presence	73.3%	98.0%	98.6%	97.4%	97.4%	344
Accent Type	73.3%	96.5%	97.4%	96.2%	96.8%	344
Boundary Presence	81.1%	91.9%	94.5%	95.4%	95.1%	344
Boundary Type	81.1%	91.0%	92.7%	93.0%	93.9%	344

We tested the realizer and the baseline *n*-gram models on the 31 sentences used to synthesize the stimuli in the perception experiment described in Section 4 (see Figure 13 for an example mini-dialogue). None of the test sentences appear verbatim in the FLIGHTS speech corpus. The test sentences contain target prosodic structures intended to be appropriate for the discourse context. Given these structures, we can quantify the accuracy with which the realizer is able to reproduce the pitch accent and edge tone choices in the target sentences, and compare it to the accuracy with which *n*-gram models predict these choices using maximum likelihood. Note that the target prosodic structures may not represent the only natural choices, motivating the need for the expert evaluation described further subsequently.

Although the realizer is capable of generating the target prosodic structure of each test sentence exactly, the test sentence (with its target prosody) is not always the top-ranked realization of the corresponding logical form, which may differ in word order or choice of function words. Thus, to compare the realizer's choices against the target accents and boundaries, we generated *n*-best lists of realizations that included the target realization, and compared this realization to others in the *n*-best list with the same words in the same order (ignoring pitch accents and edge tones). In each case, the target realization was ranked higher than all other realizations with the same word sequence, and so we may conclude that the realizer reproduces the target accent and boundary choices in the test sentences with 100% accuracy.

The accuracy with which the baseline *n*-gram models reproduce the target tunes is shown in Table 1. As the test sentences are very similar to those in the FLIGHTS speech corpus, the accent model performs remarkably well, with the bigram model reproducing the exact accent type (including no accent) in 97.4% of the cases, and agreeing on the choice of whether to accent the word at all in 98.6% of the cases. The boundary model also performs well, though substantially worse than the realizer, with the 4-gram model reproducing the boundary type (including no boundary) in 93.9% of the cases, and agreeing on boundary presence in 95.1% of the cases.[19]

Inspired by Marsi's (2004) work on evaluating optionality in prosody prediction, we asked an expert ToBI annotator, who was unfamiliar with the experimental hypotheses under investigation, to indicate for each test sentence the range of contextually appropriate tunes by providing all the pitch accents and edge tones that would be acceptable

19 Because the boundary models sometimes failed to include a boundary before a comma or full stop, default L-L% boundaries were added in these cases.

Table 2
Examples comparing target tunes to baseline prosody prediction models, with expert corrections (Items 07-2 and 06-1).

(a)	target	the only direct$_{L+H*}$ flight L-H% leaves at 5:10$_{H*}$ L-L% .
	edits	*none*
	n-grams	the only direct$_{H*}$ flight leaves at 5:10$_{H*}$ L-L% .
	edits	the only direct$_{L+H*}$ flight L- leaves at 5:10$_{H*}$ L-L% .
(b)	target	there 's a direct$_{H*}$ flight on British_Airways$_{H*}$ with a good$_{H*}$ price L-L% .
	edits	there 's a direct$_{H*}$ flight on British_Airways$_{H*}$ L- with a good$_{H*}$ price L-L% .
	n-grams	there 's a direct$_{H*}$ flight on British_Airways$_{H*}$ L-L% with a good$_{L+H*}$ price L-L% .
	edits	*none*

for each word.[20] However, our annotator found this task to be too difficult, in part because of the difficulty of coming up with all possible acceptable tunes, and in part because of dependencies between the choices made for each word. For this reason, we instead chose to follow the approach taken with the Human Translation Error Rate (HTER) post-edit metric in MT (Snover et al. 2006), and asked our annotator to indicate, for the target tune and the *n*-gram baseline tune, which accents and boundaries would need to change in order to yield a tune appropriate for the context. For the *n*-gram baseline tune, we used the choices of the bigram accent model and the 4-gram boundary model.

Examples of how the target tunes (and realizer choices) compare to those of the baseline accent and boundary prediction models appear in Table 2, along with the corrections provided by our expert ToBI annotator.[21] In Table 2(a), the target tune, which contains the theme phrase *the only direct$_{L+H*}$ flight L-H%*, was considered fully acceptable. By contrast, with the tune of the *n*-gram models, the H* accent on *direct* was not considered acceptable for the context, and at least a minor phrase boundary was considered necessary to delimit the theme phrase. In Table 2(b), we have an example where the target tune was not considered fully acceptable: Although the target consisted of a single, all-rheme intonational phrase, with no intermediate phrases, our annotator indicated that at least a minor phrase break was necessary after *British Airways*. The *n*-gram models assigned a major phrase break at this point, which was also considered acceptable. Note also that the *n*-gram models had a L+H* accent on *good*, in contrast to the target tune's H*, but both choices were considered acceptable.

Table 3 shows the number of accent and boundary corrections for all 31 test sentences at different levels of specificity. Overall, there were just 10 accent or boundary corrections for the target tunes, versus 24 for those of the baseline models, out of 688 total accent and boundary choices, a significant difference ($p = 0.01$, Fisher's Exact Test [FET], 1-tailed). With the accents, there were fewer corrections for the target tunes, but not many in either case, and the difference was not significant. Of course, with a

20 We thank one of the anonymous reviewers for this suggestion. Note that in Marsi's study, having one annotator indicate optionality was found to be a reasonable approximation of deriving optionality from multiple annotations.

21 During realization, multiwords are treated as single words, such as *British_Airways* and *5:10 (five ten a.m.)*. Accents on multiwords are distributed to the individual words before the output is sent to the speech synthesizer.

Table 3
Number of prosodic corrections of different types in all utterances and theme utterances for the target tunes and the ones selected by *n*-gram models. Items in bold are significantly different at p = 0.05 or less by a one-tailed Fisher's Exact Test.

	All Utts		*Theme Utts*	
	Target	*n*-grams	Target	*n*-grams
Total corrections	**10**	24	**3**	11
Accents	4	7	2	5
Presence	3	4	2	3
Boundaries	**6**	17	1	6
Presence	**2**	13	0	4
Major	**0**	9	0	3

larger sample size, or with test sentences that are less similar to those in the corpus, a significant difference could arise. With the boundaries, out of 344 choices, the target tunes had six corrections, only two of which involved the presence of a boundary, and none of which involved a missing major boundary; by contrast, the *n*-gram baseline had 17, 13, and 9 such corrections, respectively, a significant difference in each case (p = 0.02, p = 0.003, p = 0.002, respectively, FET). In the subset of 12 sentences involving theme phrases, where the intonation is more marked than in the all-rheme sentences, the target tunes again had significantly fewer corrections overall (3 vs. 11 corrections out of 220 total choices; p = 0.03, FET), and the difference in boundary and boundary presence corrections approached significance (p = 0.06 in each case, FET).

Having shown that the information structural constraints in the grammar play an important role in producing target realizations with contextually appropriate prosodic structures—in particular, in making acceptable boundary choices, where the choice is not deterministically rule-governed—we now briefly demonstrate that the realizer's *n*-gram model (see Section 2.6.1) has an important role to play as well. Table 4 compares the realizer's output on the test sentences against its output with the language model disabled, in which case an arbitrary choice is effectively made among those outputs allowed by the grammar. Not surprisingly, given that the grammar has been allowed to overgenerate, the realizer produces far more exact matches and far higher BLEU (Papineni et al. 2001) scores with its language model than without. Looking at the differences between the realizer's highest scoring outputs and the target realizations, the differences largely appear to represent cases of acceptable variation. The most frequent difference concerns whether an auxiliary or copular verb is contracted or not, where either choice seems reasonable. Most of the other differences represent minor variations in word order, such as $direct_{H*}$ *Air_France*$_{H*}$ *flight* vs. $direct_{H*}$ *flight on Air_France*$_{H*}$. By

Table 4
Impact of language model on realizer choice.

	Exact Match	BLEU
Realizer	61.3%	0.9505
No LM	0.0%	0.6293

contrast, many of the outputs chosen with no language model scoring contain undesirable variation in word order or phrasing; for example: *Air_France*$_{H*}$ *direct*$_{H*}$ *flight* instead of *direct*$_{H*}$ *flight on Air_France*$_{H*}$, and *you L- though would need L- to connect*$_{H*}$ *in Amsterdam*$_{H*}$ *L-* instead of *you'd need to connect*$_{H*}$ *in Amsterdam*$_{H*}$ *though L-L%*.

2.8 Interim Summary

In this section, we have introduced a presentation strategy for highlighting the most compelling trade-offs for the user that straightforwardly lends itself to the determination of information structure, which then drives the choice of prosodic structure during surface realization with CCG. We have also shown how phrase boundaries can be determined in a flexible, data-driven fashion, while respecting the constraints imposed by the grammar. An expert evaluation demonstrated that the approach yields prosodic structures with significantly higher acceptability than strong *n*-gram baseline prosody prediction models. In the next two sections, we show how the generator-driven prosody can be used to produce perceptibly more natural synthetic speech.

3. Unit Selection Synthesis with Prosodic Markup

In this section we describe the unit selection voices that we employed in our perception experiment (Section 4). Three voices in total were used in the evaluation: GEN, ALL, and APML. Each was a state-of-the-art unit selection voice using the Festival *Multisyn* speech synthesis engine (Clark, Richmond, and King 2007). The GEN voice, used as a baseline, is a general-purpose unit selection voice. The ALL voice is a voice built using the same data as the GEN voice but with the addition of the data from the FLIGHTS speech corpus described in Section 3.1. The APML voice augments the in-domain data in the ALL voice with prosodic markup, which is then used at run-time by the synthesizer in conjunction with marked-up input to guide the selection of units.

To provide in-domain data for the ALL and APML voices, we needed to record a suitable data set from the original speaker used for the GEN voice. We describe the process of constructing this speech corpus for FLIGHTS next, in Section 3.1. Then, in Section 3.2, we describe the unit selection voices in detail, along with how the in-domain data was used in building them.

3.1 The FLIGHTS Speech Corpus

The FLIGHTS speech corpus was intended to have a version of each word that needs to be spoken by the system, recorded in the context in which it will be spoken. This context can be thought of as a three-word window centered around the given word, together with the word's target pitch accent and edge tone, if any. This would theoretically provide more than sufficient speech data for a full limited domain voice, and a voice using this data would be guaranteed to have a unit sequence for any sentence generated by FLIGHTS where each individual unit is a tight context match for the target unit.

Because the system was still limited in its generation capabilities at the time of recording the voice data for FLIGHTS, we developed a recording script by combining the generated data that was available with additional utterances that we anticipated the finished system would generate. To do so, we assembled a set of around 50 utterance templates that describe flight availability—with slots to be filled by times, dates, amounts, airlines, airports, cities, and flight details—to which we added approximately two hundred individual or single-slot utterances, which account for the introductions,

questions, confirmations, and other responses that the system makes. We then made use of an algorithm designed by Baker (2003) to iterate through the filler combinations for each utterance template to provide a suitable recording script.[22]

To illustrate, Example (4) shows an example utterance template, along with two utterances in the recording script created from this template. The example demonstrates that having multiple slots in a template makes it possible to reduce the number of sentences that need to be recorded to cover all possible combinations of fillers. With (b) and (c) recorded as illustrated to fill the template in (a), we would then have the recorded data to mix and match the slot fillers with two different values each to synthesize eight different sentences. It is also often possible to use the fillers for a slot in one utterance as the fillers for a slot in another utterance, if the phrase structure is sufficiently similar. A more complex case is shown in Example (5), involving adjacent slots. In this case, it is not sufficient to just record one example of each possible filler, as the context around that filler is changed by the presence of other slots; instead, combinations of fillers must be considered (Baker 2003).

(4) a. It arrives at $\langle$TIME$\rangle_{H*}$ L-H% and costs just $\langle$AMOUNT$\rangle_{H*}$ L-L%, but it requires a connection$_{H*}$ in $\langle$CITY$\rangle_{H*}$ L-L%.
b. It arrives at six$_{H*}$ p.m.$_{H*}$ L-H% and costs just fifty$_{H*}$ pounds$_{H*}$ L-L%, but it requires a connection$_{H*}$ in Paris$_{H*}$ L-L%.
c. It arrives at nine$_{H*}$ a.m.$_{H*}$ L-H% and costs just eighty$_{H*}$ pounds$_{H*}$ L-L%, but it requires a connection$_{H*}$ in Pisa$_{H*}$ L-L%.

(5) a. There are $\langle$NUM$\rangle_{H*}$ $\langle$MOD$\rangle_{H*}$ flights L-L% from $\langle$CITY$\rangle_{H*}$ to $\langle$CITY$\rangle_{H*}$ today L-L%.
b. There are two$_{H*}$ earlier$_{H*}$ flights L-L% from Bordeaux$_{H*}$ to Amsterdam$_{H*}$ today L-L%.

The recording script presented similar utterances in blocks. It made use of small caps to indicate intended word-level emphasis and punctuation to partially indicate desired phrasing, as the speaker was not familiar with ToBI annotation. The intended dialogue context for the utterances was discussed with our speaker, but the recordings did not consist of complete dialogues, as using complete dialogues would have been too time consuming. The recording took place during two afternoon sessions and yielded approximately two hours of recorded speech. The resulting speech database consisted of 1,237 utterances, the bulk of which was derived from the utterances that describe flight availability.

The recordings were automatically split into sentence-length utterances, each of which had a generated APML file with the pitch accents and edge tones predicted for the utterance. The prosodic structures were based on intuition, as there was no corpus of human–human dialogues sufficiently similar to what the system produces that could have been used to inform prosodic choice.

The recordings were then manually checked against the predicted APML, by listening to the audio and looking at the f0 contours to see if the pitch accents and edge tones matched the predicted ones. In the interest of consistency, changes were only

22 The target prosody for each utterance template was based on the developers' intuitions, with input from Mark Steedman. As one of the anonymous reviewers observed, an alternative approach might have been to conduct an elicitation study in a Wizard-of-Oz setup to determine target tunes. This strategy may have yielded more natural prosodic variation, but perhaps at the expense of employing less distinctive tunes. For this project, it was not practical to take the extra time required to conduct such an elicitation study.

made to the APML files when there were clear cases of the speaker diverging from the predicted phrasing or emphasis; nevertheless, the changes did involve mismatches in both the presence and the type of the accents and boundaries. As an example of the kind of changes that were made, in Example (4), we had predicted that the speaker would use a L-H% boundary (a continuation rise) prior to the word *but*, however she instead consistently used a low boundary tone in this location. Consequently, we changed all the APML files for sentences with this pattern to include a L-L% compound edge tone at that point. Further details of this data cleanup process are given in Rocha (2004).

3.2 The Synthetic Voices

Ideally, we would like to build a general purpose unit selection voice where we can fully specify the intonation in terms of ToBI accents and boundaries and then have the synthesizer generate this for us. This, however, is currently not a fully viable option for the reasons discussed subsequently, so instead we built a number of different voices using the unit selection technique to investigate how working towards a voice with full intonation control would function. There are a number of ways in which prosodic generation for unit selection can be approached (Aylett 2005; van Santen et al. 2005; Clark and King 2006), with most systems designed to produce best general prosody. This contrasts with the framework chosen here, which is designed to make maximum use of in-domain data, in an attempt to produce a system that realizes prosody as well as is possible in that domain. This can be considered an upper bound for what more general systems could achieve under optimal conditions.

The ideal way to use intonation within the unit selection framework requires three components: the database of speech, which must be fully annotated for intonation; the predicted intonation for a target being synthesized; and a target cost that ensures suitable candidates are found to match the target. Each of these requirements prove to be difficult to achieve successfully.

Manually labeling a large speech database accurately and consistently with intonation labels is both difficult and expensive, whereas automatic labeling techniques tend to produce mediocre results at best. Producing accent labeling for the flight information script is somewhat easier because of the limited number of templates that the script is based upon, and once the templates are labeled, intonation for individual sentences can be derived. To build a general purpose unit selection voice for the FLIGHTS domain, we would want to combine the FLIGHTS data with speech data designed to provide more general coverage. Providing accent labeling for the extra, non-FLIGHTS, data is the main problem here.

Predicting intonation for a target utterance is generally a difficult task, and can only really be done well when the domain of the speech synthesis is constrained. For example, a number of statistical modeling techniques may be able to provide adequate accent prediction for a system to read the news or forecast the weather, because the sentence structure is usually limited to a sequence of simple statements. The task becomes difficult when the sentence structure is more variable, for example in dialogue where a number of different question forms may exist along with contrastive statements, and so forth. In the FLIGHTS domain we can side-step the prediction problem by providing a specification for the intonation of a sentence as part of language generation.

In one sense, defining a target-cost component to direct the search towards finding suitable intonation is not difficult, as a simple penalty for not matching the target intonation suffices. However, standard unit selection techniques only ever take into

account local effects, and no provision is made to ensure a suitable global intonation contour.

These problems, and a number of technical issues relating to the use of markup, automatic segment alignment, and the voice building process, prohibit us from building a general purpose unit selection voice where we can specify the required intonation. One of the questions that this study is attempting to answer is whether it is worth trying to resolve these issues to build such a system in the future. To address this question we present a number of voices designed to investigate the issues of producing natural speech synthesis in the FLIGHTS domain.

3.2.1 The GEN Voice. The GEN voice uses a database of approximately 2,000 sentences of read newspaper speech. The Festival *Multisyn* unit selection engine selects diphone units from the database by minimizing a combination of a target cost and a join cost. The target cost scores the linguistic context of the diphone in terms of stress, position of the diphone in the current word, syllable and phrase, phonetic context, and part of speech. The join cost scores the continuity between selected units in terms of spectrum, energy, and f0. There is no explicit modeling of prosody in this system.

The GEN voice was used as a baseline. There is nothing specific to the FLIGHTS domain associated with any part of this voice, and the quality of the resulting synthesis is comparable with a typical unit selection speech synthesiser.

3.2.2 The ALL voice. The ALL voice was created by augmenting the GEN voice with the speech data recorded as part of the FLIGHTS speech corpus described herein. The motivation here is to attempt to provide a system which would have the best possible quality that a general purpose unit selection synthesiser could have when working in this domain. The additional data increases the availability of units in the exact contexts that would be required by the FLIGHTS system. Having examples of airport and airline names in the database, for example, increases the likelihood that when these words are synthesized there are appropriate, often consecutive, units available to synthesize them. Naturalness is improved both by the better context match and by the need for fewer joins in constructing these words and the utterances they are used in.

3.2.3 The APML voice. The APML voice is different from the ALL voice in that it is designed to take APML input from the FLIGHTS system rather than text input. The APML voice comprises the same speech data as the ALL voice, but also includes the APML annotation for the FLIGHTS part of the corpora. The target cost for the synthesizer is augmented with a prosodic component which penalizes any mismatch between supplied APML input and the APML specification associated with a particular unit. As the read newspaper component of this voice does not have accompanying APML markup, and because the voice is required to work for text input (although this capability is not used here), the target cost penalizes (1) the synthesis of an APML-specified target with a unit that does not have accompanying APML markup; (2) the synthesis of a target without APML specification with APML-specified units, and (3) synthesis where there is an APML mismatch between the target and candidate. It is important that these prosodic mismatches are only discouraged, rather than forbidden, to allow the system to synthesize out of the original domain if required.

As work on the FLIGHTS system progressed, we found numerous cases where the speech corpus failed to anticipate all the possible outputs of the system. For example, although we included an utterance template for conveying price in a conjoined verb phrase, as in Example (4), we did not include a template for conveying price in a

single independent clause, as in *It costs just fifty pounds*. Had the system been further along in its development when we recorded the speech data, we could have pursued a strategy of selecting utterances to record from generated outputs, in order to maximize some coverage criterion.[23] In either case, though, we consider it difficult to develop a recording script that will completely cover all three-word sequences that a system will ever generate, especially if further development is considered. For this reason, we believe it to be essential to have a strategy to handle the cases where generation needs go beyond the original plan. The use of an augmented general purpose unit selection system—rather than, for example, a strictly limited domain system—means that there is no difficulty in synthesizing extra material as needed, although there is the risk that it will not sound quite as good as that which is closer in context to the original in-domain recordings. For an APML voice where the input is "new" APML, if there are no suitable units within the APML-annotated section of the corpus, units will be chosen from the main portion of the corpus. There will be a penalty in terms of target cost for doing so, but the best sequence will still be found.

4. Synthesis Evaluation

In this section, we describe a perceptual experiment which was carried out to determine whether the prosodic structures generated by the FLIGHTS system actually result in improved naturalness in speech synthesis. We also describe an evaluation of the prosody in the synthesized speech that makes use of an expert annotator's assessments of the acceptability of the perceived tunes for the given context, and an evaluation that examines objective differences in the f0 contours of the theme phrases in the synthesized speech. The three types of evaluation (subject ratings, expert annotation, and objective measures) all show the APML voice outperforming the ALL voice, both on the complete set of test utterances as well as on the subset containing theme phrases. Additionally, the three types of evaluation support each other in that differences in ratings of particular items correspond to differences in acceptability annotations and to differences in the objective measures, as will be explained in the following.

4.1 Perception Experiment

4.1.1 Methodology. Our experimental hypothesis was that listeners would prefer the APML voice, used with contextually appropriate intonation markup, over the ALL and GEN control voices. We further hypothesized that the preference would be larger for the utterances containing theme phrases, where the intonation is more marked than it is in the all-rheme utterances.

Subjects were presented with mini-dialogues consisting of a summary of the user's request in text and three versions of the system's response, one for each of the three voices, as shown in Figure 12. System utterances were presented side-by-side, with each system turn comprising two to four utterances, and each voice labeled as A, B, or C. The label assignments were balanced across the mini-dialogues so that each voice appeared an equal number of times labeled as A, B, or C, and the mini-dialogues were presented in an individually randomized order. Subjects were asked to assign ratings to each version of each utterance on a 1–7 scale, with 7 corresponding to "completely natural" and 1 corresponding to "completely unnatural." Ratings were gathered on-line

23 For example, in developing a custom voice for the COMIC system, we selected from generated utterances in order to maximize bigram coverage with high priority named entities.

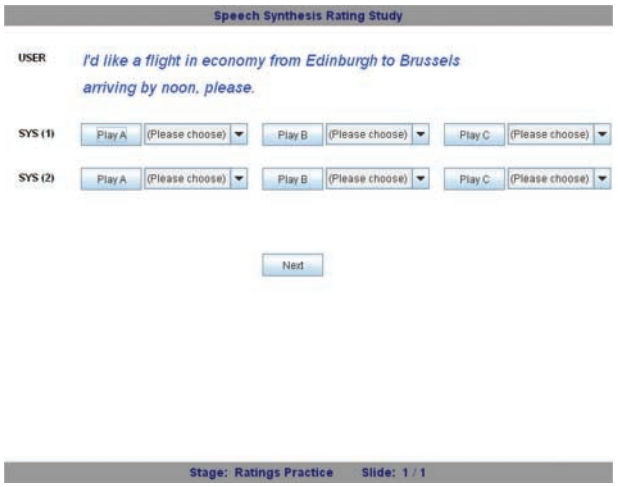

Figure 12
Screenshot of `webexp2` interface for gathering listener ratings.

using `webexp2`.[24] Subjects were allowed to play the sound files any number of times in any order, but were required to assign ratings to all the utterances before proceeding to the next screen. In assigning their ratings, subjects were instructed to pay attention to the context given by the summary of the user's request, keeping the following questions in mind:

- Does the utterance make it clear how well the flight (or flights) in question meet the user's needs?
- Are words emphasised in a way that highlights the trade-offs among the different options?
- For the second and subsequent utterances, is emphasis used in a way that makes sense given the previous system utterances?
- Is the utterance clear and easy to understand, or garbled and difficult to understand?

Twelve mini-dialogues were used as stimuli, comprising 31 utterances in total. The dialogues were constructed so as to contain a representative range of theme phrases, with each mini-dialogue containing one utterance with a theme phrase. An example dialogue, with target tunes for the APML voice, appears in Figure 13; the second system utterance contains a theme phrase. The complete set of stimuli, including sound files, is available on-line[25] from the first author's web page.

Fourteen native English speaking subjects participated in the study. The subjects were all from the UK or USA and had no known hearing deficits. For participating in the study, subjects were entered into a prize drawing.

24 `http://www.webexp.info/`.
25 `http://www.ling.ohio-state.edu/~mwhite/flights-stimuli/`.

USER: I'd like a morning flight from Edinburgh to Berlin please, preferably on Lufthansa.

SYS (1): There's a direct$_{H*}$ flight on Flybe$_{H*}$ L-L% , departing Edinburgh$_{H*}$ at eleven$_{H*}$ eleven$_{H*}$ a.m.$_{H*}$ L-L% .
SYS (2): The Lufthansa$_{L+H*}$ flight L-H% leaves at ten$_{H*}$ a.m.$_{H*}$ L-L% .
SYS (3): It requires a connection$_{H*}$ in Frankfurt$_{H*}$ though L-L% .

Figure 13
Example mini-dialogue (Dialogue 03).

4.1.2 Results. As shown in Figure 14, the APML voice received higher ratings than the ALL voice, and both the APML and ALL voices scored much higher than the GEN voice. Overall, the APML voice's average rating surpassed that of the ALL voice by 0.77; its score of 5.83 was close to 6 on the rating scale, corresponding to "mostly natural," while the ALL voice's score was 5.06, just above 5 on the scale, corresponding to "somewhat natural." The difference between the two voices was highly significant (paired t-test, $t_{433} = 10.20$, $p < 0.001$). On the theme utterances, the difference between the APML and ALL voices was even larger, at 0.91. With the APML voice, there was no significant difference between the average ratings of the theme utterances and those without theme phrases. In contrast, the ALL voice showed a trend towards the theme utterances scoring worse than the remaining ones (t-test, 1-tailed, $t_{432} = -1.39$, p = 0.08), with the average of the theme utterances 0.19 lower than that of the all-rheme utterances. The GEN voice did considerably worse (0.48) on the theme utterances (t-test, 1-tailed, $t_{432} = -3.06$, p = 0.001).

4.2 Expert Prosody Evaluation

4.2.1 Methodology. To more directly examine whether the APML voice yielded more contextually appropriate prosody than the ALL voice, we asked an expert ToBI annotator, who was unfamiliar with the experimental hypotheses under investigation, to annotate the perceived tunes in the synthesized stimuli for these two voices. In cases of uncertainty, multiple annotations were given. The stimuli from the ALL voice were always presented first, so that listening to the tune from the APML item would not bias our annotator against the corresponding ALL item. Because there is not necessarily a unique tune that is acceptable for the context, we also asked our annotator to indicate the closest acceptable tune by indicating which accents and boundaries would need to

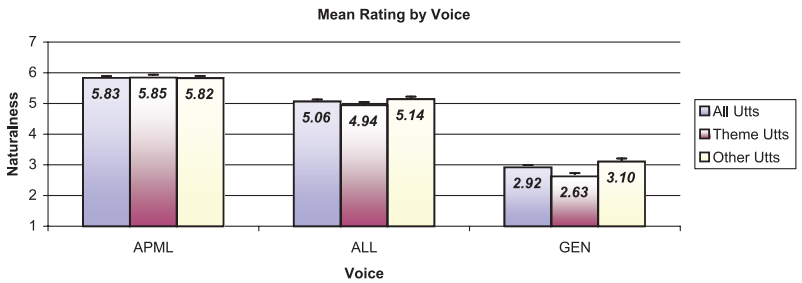

Figure 14
Mean ratings for each voice. Theme utterances are the subset containing a theme phrase. (Error bars show standard errors.)

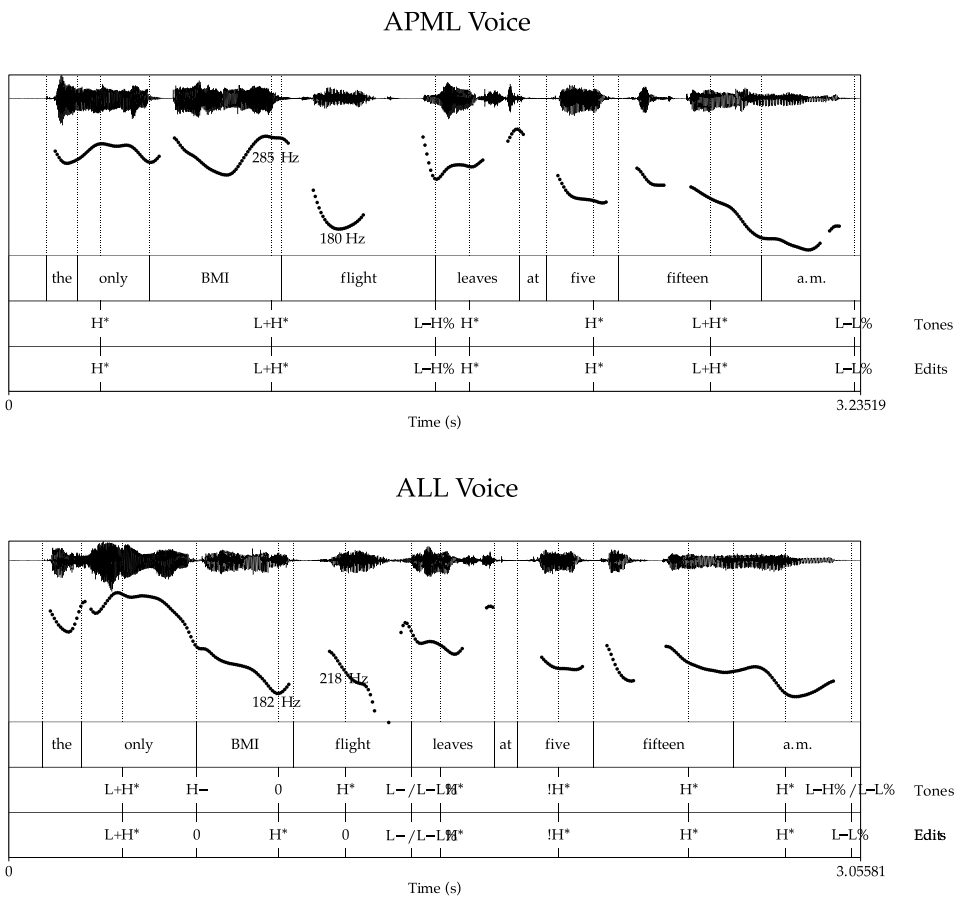

Figure 15
Example annotation and f0 showing tune corrections (Item 05-2).

change in order to yield a tune appropriate for the context, in much the same fashion as in our prosody prediction evaluation (Section 2.7).[26] We then counted the number of accent or boundary corrections at different levels of specificity.

An example annotation with tune corrections appears in Figure 15. This utterance is the second one in Dialogue 05, where the user prefers to fly BMI; accordingly, the utterance contains the theme phrase (with target tune) *the only* BMI_{L+H*} *flight L-H%*. With the APML version of the utterance, the annotator perceived a L+H* accent on *BMI* and a L-H% boundary on de-accented *flight*, as desired. Additionally, the annotator perceived a H* accent on *only*, which was not part of the target tune. The tune was nevertheless considered completely acceptable, as the Edits tier is identical to the Tones tier. By contrast, with the ALL version of the utterance, *BMI* was less prominent than *only* and *flight*, and accordingly it received no pitch accent, whereas *only* and *flight* received L+H* and H* accents, respectively; in addition, a H- boundary was annotated on *only*, and a boundary that was uncertain between L- and L-L% was annotated on

26 The target tunes were not presented together with the synthesized stimuli, to avoid influencing the tunes perceived by our expert.

Table 5
Number of prosodic corrections of different types in all utterances and theme utterances for the two voices. Items in **bold** are significantly different at p = 0.05 or less by a one-tailed Fisher's Exact Test.

	All Utts		*Theme Utts*	
	APML	ALL	APML	ALL
Total corrections	**22**	49	**11**	26
Accents	**10**	33	**4**	16
Presence	**6**	20	**2**	8
L+H*	**4**	13	**2**	8
Boundaries	12	16	7	10
Presence	7	6	3	4

flight. This tune for the theme phrase was not considered acceptable for the context: As the Edits tier shows, the lack of an accent on *BMI*, and the presence of an accent on *flight*, was corrected by our annotator, as was the H- minor phrase boundary on *only*. This choice makes sense for the context, as *BMI* is what distinguishes this option from the one suggested in the first utterance, while *flight* is given information at this point in the dialogue. Indeed, it is difficult to come up with an interpretation of $only_{L+H*}$ *BMI*, with no accent on *BMI*, though this tune might make sense if the question of whether the flight was code-shared with another airline was a salient one in the context.

4.2.2 Results. The results of the expert prosody evaluation appear in Table 5. Across all 31 utterances, there were 49 accent or boundary corrections for the ALL voice, versus just 22 for the APML voice, out of 688 total accent and boundary choices, a highly significant difference ($p < 0.001$, Fisher's Exact Test, 1-tailed). With the 12 theme utterances, there were 26 corrections for the ALL voice, versus 11 for the APML voice, out of 220 total choices (p = 0.008, FET). Looking at the pitch accents, the difference in the number of accent corrections was significant in each case ($p < 0.001$, FET, and p = 0.004, FET, respectively), as was the number of corrections involving the presence or absence of a pitch accent (p = 0.004, FET, and p = 0.05, FET, respectively) and the number involving L+H* accents (p = 0.02, FET, and p = 0.05, FET, respectively).[27] Looking at the boundaries, we may observe that with the ALL voice, the majority of corrections were with accents, whereas with the APML voice, the majority were with boundaries. Although the APML voice had fewer boundary corrections, the difference was not significant.

As noted in the discussion of Figure 15, the perceived tunes with the APML voice did not always exactly match the target tunes, sometimes in ways that our expert annotator considered acceptable. More commonly, however, mismatches between the perceived and target tunes were not judged to be acceptable. In fact, of the 22 corrections for the APML voice, 19 would have been acceptable had the target tune been successfully achieved; that is, 19 of the 22 corrections changed a perceived accent or boundary to the one in the target tune. In 12 of these 19 cases, our annotator perceived an accent or boundary where the target tune had none. There were also five other cases

27 A correction was counted as involving the presence or absence of a pitch accent if the word was perceived as having an accent when it was deemed that it should have none, or was perceived as having no accent when it was deemed that it should have one; L+H* accent and boundary presence corrections were counted in the same fashion.

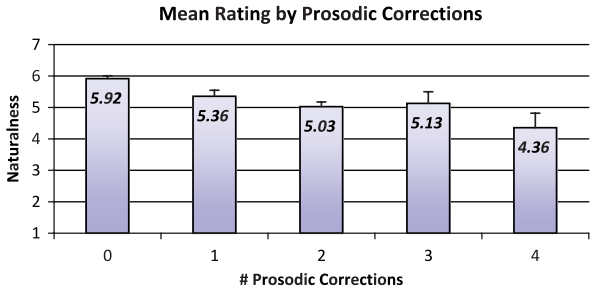

Figure 16
Mean ratings of items with 0–4 prosodic corrections for the APML and ALL voices combined. (Error bars show standard errors.)

where the target was a L-L% boundary, but a rising boundary was perceived instead. The remaining cases involved accent type mismatches.

To examine the relationship between the number of corrections indicated by our expert ToBI annotator and the ratings in the perception experiment, we grouped the APML and ALL utterances by the number of corrections and calculated the mean ratings for each group. The results appear in Figure 16, which shows the expected relationship between mean ratings and the number of corrections, with ratings going down as the number of corrections go up. The pattern is less clear with two to four corrections, where there are fewer tokens. One-tailed t-tests show that utterances with zero corrections were rated significantly higher than utterances with one to four corrections ($t_{39} = 2.96$, $t_{35} = 5.14$, $t_{30} = 3.03$, $t_{28} = 5.35$, respectively; $p < 0.01$ in each case); utterances with one correction were also significantly higher than those with four corrections ($t_{17} = 2.30$, $p = 0.02$).

We also examined the relationship between the listener ratings and the number of errors noted by our expert annotator through a multiple regression analysis. The regressors were the number of accent or boundary presence errors and the number of remaining accent or boundary errors, involving only a difference in type. The regression equation was $Rating = 5.85 - 0.41 \times PresenceErrors - 0.28 \times TypeOnlyErrors$, which accounted for 32% of the variance and was highly significant ($F_{2,59} = 15.3$, $p < 0.001$). As expected, ratings were negatively related to accent and boundary errors, with presence errors showing a greater impact than type-only errors. Both the effect of presence errors ($t_{59} = -4.34$, $p < 0.001$) and the effect of type-only errors ($t_{59} = -2.65$, $p = 0.01$) were significant. Because both kinds of errors had a significant impact on ratings, the results suggest that it is worthwhile to make use of fine-grained ToBI categories where feasible, as we discuss further in Section 6.

4.3 Objective Measures

4.3.1 Methodology. In addition to the expert prosody evaluation, we also measured the f0 values and durations of the adjectives and nouns in the theme phrases, to see whether they differed significantly between the APML and ALL voices. We also measured the drop in f0 between the adjective, which should receive a contrastive pitch accent, and the noun, which should be deaccented. Note that the f0 drop is a potentially more fine-grained measure than pitch accent corrections, because the accents could be considered acceptable even though the tune is less distinct in some cases.

An example in which this occurs appears in Figure 17, which shows the second utterance of Dialogue 03, seen earlier in Figure 13, for the APML and ALL voices. Here,

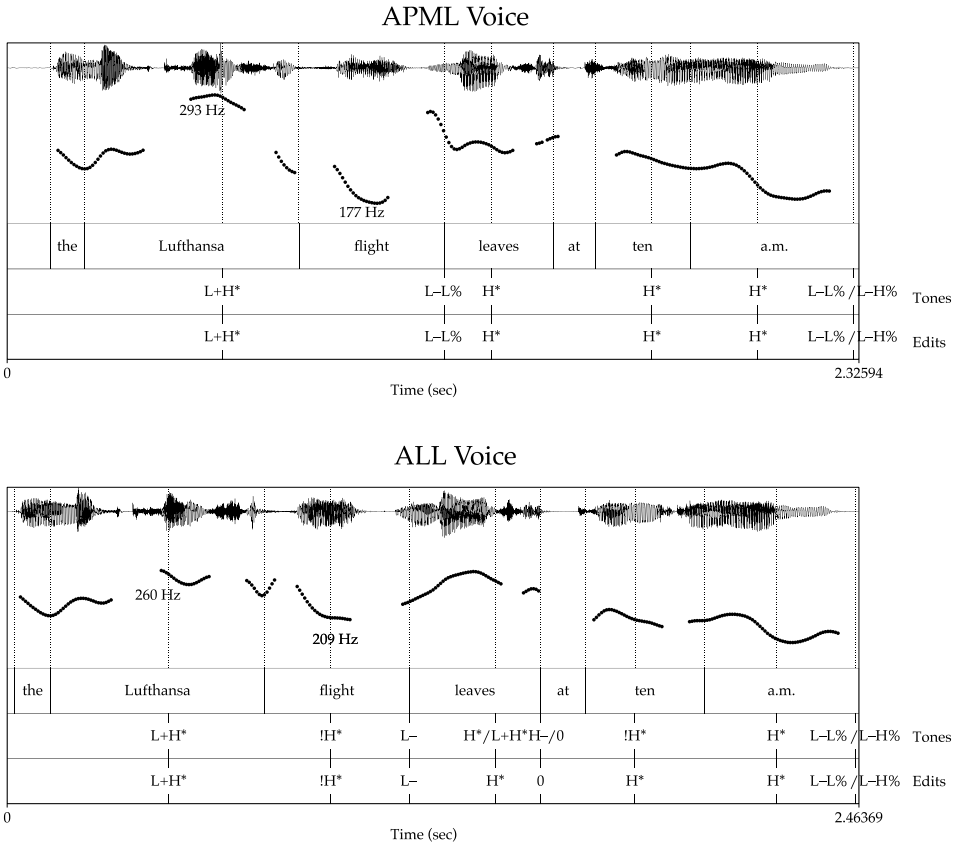

Figure 17
Example annotation and f0 showing difference in f0 drop in the theme phrases (Item 03-2).

the APML version was annotated with the target accents, namely L+H* for *Lufthansa* and none for *flight*, whereas the ALL version was annotated with a L+H* !H* pattern, which was also considered acceptable. However, with the APML version the pitch drops from an f0 value of 293 Hz to 177 Hz, whereas with the ALL version the pitch only drops from 260 Hz to 209 Hz. As a result, the theme tune is much less distinctive in the ALL version, in a way that could impact the ease with which the theme phrase's discourse function is identified.

4.3.2 Results. Figure 18 shows the mean f0 values across all 12 theme phrases for the adjective, which should receive contrastive emphasis, and the head noun, which should be reduced. As the figure shows, the pattern seen in Figure 17 was borne out on the whole in the stimuli with theme phrases. In particular, the theme phrases for the APML voice had an f0 drop of 90 Hz on average from the adjective to the noun, whereas the ALL voice only had an f0 drop of 50 Hz, a significant difference (t-test, 1-tailed, $t_{22} = 2.60$, $p = 0.008$). The durations of the adjectives and nouns, however, did not show a significant difference. In addition to the f0 drop, a significant difference was also found with the f0 max on the adjective and the f0 min on the noun (t-tests, 1-tailed, $t_{28} = 1.70$, $p = 0.05$ and $t_{22} = -1.86$, $p = 0.04$, respectively). Finally, a relatively high correlation ($r = 0.78$) was also found between the difference in f0 drop and the difference in the ratings of the theme utterances for the two voices ($t_{10} = 3.94$, $p = 0.001$), suggesting that the less

distinctive theme tunes produced by the ALL voice were perceived as less natural than the ones produced by the APML voice.

4.4 Discussion

The perception experiment confirmed our hypothesis that listeners would prefer the APML voice, used with contextually appropriate intonation markup, over the ALL and GEN control voices. Both on the complete set of utterances, as well as the subset containing theme phrases, the APML voice was rated substantially higher on average than the ALL voice, and much higher than the GEN voice. Note that the ALL voice was also rated much higher on average than the GEN voice—which is not surprising, given that it has access to the same limited domain data as the APML voice—showing that the ALL voice serves as a tough baseline to beat. The expert prosody evaluation and the objective measures also confirmed the superiority of the APML voice, in particular on the theme utterances.

We also found evidence in favor of our hypothesis that the preference for the APML voice would be larger for the utterances containing theme phrases—where the intonation is more marked than it is in the all-rheme utterances—as the difference between the mean ratings of the APML and ALL voices was larger on the theme utterances than on the remaining ones. Additionally, with the APML voice, there was no significant difference between the mean ratings of the theme utterances and those without theme phrases, whereas the ALL voice showed a trend towards the theme utterances scoring worse than the all-rheme ones, and the GEN voice clearly did considerably worse on the theme utterances. One small surprise was that with the ALL voice, the difference between the mean ratings of the theme and all-rheme utterances did not reach the standard level of statistical significance. Of course, it could be that with a larger sample size, a more significant difference would be found. However, it is undoubtedly the case that the ALL voice dropped off less on the theme utterances than did the GEN voice, and the reason is almost certainly that the limited domain data has good coverage of the theme phrases, and thus the ALL voice often does reasonably well on the theme utterances even without explicit prosodic control. What is perhaps more remarkable is that the ALL voice did not do better on the all-rheme utterances, as can be seen in the number of expert corrections listed in Table 5 for all the utterances, which go well beyond those in the theme utterance subset. That the ALL voice had more than double

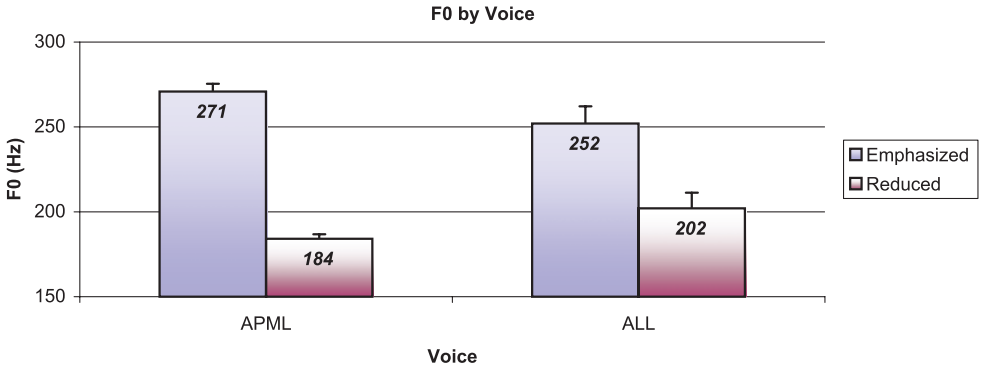

Figure 18
Mean f0 values for emphasized and reduced words in theme phrases for the APML and ALL voices. (Error bars show standard errors.)

the number of corrections as the APML voice on both the complete set of utterances and the subset containing theme phrases shows that the prosodic specifications were important throughout.

Finally, we may observe that although the APML voice had fewer boundary corrections than did the ALL voice, the difference did not reach significance, suggesting that there is room for improvement in how boundaries are handled in the APML voice. In particular, this result suggests that edge tones, and intermediate phrase boundaries in particular, should affect the selection of units non-locally, as theoretically their effect on the pitch contour spreads back to the last pitch accent. In fact, it may well be that because the speech synthesis system only models prosodic effects locally, essentially at the syllable level, and does not take utterance-level structures such as tune into account, a ceiling has been reached for both accent and phrasing performance. Representing global prosodic structures to ensure prosodic coherence will be one of the major challenges for future generations of speech synthesis systems.

5. Related Work

The FLIGHTS system combines and extends earlier approaches to user-tailored generation in spoken dialogue. A distinguishing feature of FLIGHTS is that it adapts its output according to user preferences at all levels of the generation process, from the selection of content to linguistic realization and the prosody targeted in speech synthesis.

The most similar system to ours is MATCH (Walker et al. 2004), which employs simpler content planning strategies and does not explicitly point out the trade-offs among options. MATCH also uses simple templates for realization, and does not attempt to control intonation. Carenini and Moore's (2000, 2006) system is also closely related, but it does not make comparisons, and generates text rather than speech. Carberry et al.'s (1999) system likewise employs additive decision models in recommending courses, though their focus is on dynamically acquiring a model of the student's preferences, and the system is limited to recommending a single option considered better than the user's current one. In addition, the system only addresses the problem of selecting positive attributes to justify the recommendation, and does not plan and prosodically realize the positive and negative attributes of multiple suggested options.

These systems all employ a user model to select a small set of good options, and to identify the attributes that justify their desirability, in order to present a summary, comparison, or recommendation to the user. Evaluation showed that tailoring recommendations and comparisons to the user increased argument effectiveness and improved user satisfaction (Walker et al. 2004). Thus, the user-model (UM-) based approach is an appropriate strategy for spoken dialogue systems when there are a small number of options to present, either because the number of options is limited or because users can supply sufficient constraints to winnow down a large set before querying the database of options.

Other researchers have argued that it is important to allow users to browse the data for a number of reasons: (1) if there are many options that share attribute values, they will be very close in score when ranked using the UM-based approach; (2) users may not be able to provide constraints until they hear more information about the space of options; and (3) the UM-based approach does not give users an overview of the option space, and this may reduce their confidence that they have been told about the best option(s) (Demberg and Moore 2006).

Polifroni, Chung, and Seneff (2003) proposed a "summarize and refine" (SR) approach, in which the system structures a large number of options into a small number

of clusters that share attributes. The system then summarizes the clusters based on their attributes, implicitly prompting the user to provide additional constraints. The system produces summaries such as

> *I have found 983 restaurants. Most of them are located in Boston and Cambridge. There are 32 choices for cuisine. I also have information about price range.*

which help the user get an overview of the option space. Chung (2004) extended this approach by proposing a constraint relaxation strategy for coping with queries that are too restrictive to be satisfied by any option. Pon-Barry, Weng, and Varges (2006) found that fewer dialogue turns were necessary when the system proactively suggested refinements and relaxations.

However, as argued in Demberg and Moore (2006), there are several limitations to the SR approach. First, many turns may be required during the refinement process. Second, if there is no optimal solution, exploration of trade-offs is difficult. Finally, the attributes on which the data has been clustered may be irrelevant for the specific user. Demberg and Moore (2006) subsequently developed the user-model-based summarize and refine approach (UMSR) to combine the benefits of the UM and SR approaches, by integrating user modeling with automated clustering. When there are more than a small number of relevant options, the UMSR approach builds a cluster-based tree structure which orders the options to allow for stepwise refinement. The effectiveness of the tree structure, which directs the dialogue flow, is optimized by taking the user's preferences into account. Trade-offs between alternative options are presented explicitly to give the user a better overview of the option space and lead the user to a more informed choice. To give the user confidence that they are being presented with all relevant options, a brief account of the remaining (irrelevant) options is also provided. Results of a laboratory experiment comparing the SR and UMSR approaches demonstrated that (1) participants preferred UMSR, (2) UMSR presentations are as easy to understand as those of SR, (3) UMSR increases overall user satisfaction, (4) UMSR significantly improves the user's overview of the available options, and (5) UMSR increases users' confidence in having heard about all relevant options. Although the UMSR approach has not been implemented in the FLIGHTS system, it could be used when there are a large number of available options to winnow them down to a handful of relevant ones, which would then be compared following the approach described in this article.

As regards our work on intonation, as stated in the introduction, Prevost's (1995) generator has directly informed our approach to information structure and prosody; his system does not make use of quantitative user models though, and only describes single options. Theune (2002) likewise follows Prevost's approach in her system, refining the way contrast is determined in assigning pitch accents. Theune et al. (2001) show that a system employing syntactic templates and a rule-based prosody assignment algorithm leads to more natural synthesis (of Dutch); unlike FLIGHTS though, their D2S system does not employ a user preference model or a notion of theme phrase, and does not distinguish different types of pitch accents. Also closely related is Kruijff-Korbayová et al.'s (2003) information-state based dialogue system, in which the authors explore a similar approach to using information structure across dialogue turns; however, their system does not make use of a user model, and employs template-based realization with much simpler sentence structures. Kruijff-Korbayová et al. likewise present an evaluation indicating that the contextual appropriateness of spoken output (in German) improves when intonation is assigned on the basis of information structure. In comparison with our evaluation, theirs examines improvements over a general purpose

text-to-speech voice with default intonation, rather than a limited domain voice, which provides a much higher baseline in terms of the naturalness of the resulting intonation.

Less closely related is most work on machine-learning approaches to prosody prediction in text-to-speech (TTS) systems (Hirschberg 1993; Hirschberg and Prieto 1996; Taylor and Black 1998; Dusterhoff, Black, and Taylor 1999; Brenier et al. 2006) and concept-to-speech (CTS) systems (Hitzeman et al. 1998, 1999; Pan, McKeown, and Hirschberg 2002). These approaches have typically aimed to develop generic models of prosody prediction, by training classifiers for accents and boundaries that make use of a considerable variety of features. For example, in predicting accent placement (but not type), Hitzeman et al.'s CTS system makes use of rhetorical relations such as **list** and **contrast**, along with the reference type of NPs and whether they represent first mentions of an entity in the discourse. In Pan et al.'s more comprehensive study, their system predicts accent placement (but not type), break indices, and edge tones based on features extracted from the SURGE realizer (Elhadad 1993), deep semantic and discourse features, including semantic type, semantic abnormality and given/new status, and surface features, such as part of speech and word informativeness. However, neither of these CTS approaches makes use of the theme/rheme distinction, or the notion of kontrast that stems from Rooth's (1992) work on alternative sets, both of which are crucial to Steedman's (2000a) theory of how information structure constrains prosodic choices. More recently, Brenier et al. have shown that the ratio of accented to unaccented tokens of a word in spontaneous speech is a surprisingly effective feature in predicting pitch accents; they also argued that using information status and contrast is unlikely to improve upon prominence prediction based only on surface features, since these manually labeled features did not yield substantial improvements in their decision tree models. Again, however, their approach does not make use of the theme/rheme distinction, and does not attempt to predict pitch accent type or edge tones; in addition, they report frequent errors on auxiliaries and negatives (e.g., *no*), which we have found to be important for highlighting trade-offs prosodically.

In contrast to these approaches, we have emphasized the generation and synthesis of sharply distinctive theme and rheme tunes in the limited domain of a dialogue system, using a hybrid rule-based and data-driven approach. In particular, whereas Hitzeman et al. (1998, 1999) and Pan, McKeown, and Hirschberg (2002) make use of individual classifiers for prosodic realization decisions—with no means of tying the decisions of these classifiers together—our approach instead uses rules and constraints in the grammar to specify a space of possible realizations, through which the realizer searches to find a sequence of words, pitch accents, and edge tones that maximizes the probability assigned by an *n*-gram model for the domain.

In an approach that is more similar in spirit to ours, Bulyko and Ostendorf (2002) likewise aim to reproduce distinctive intonational patterns in a limited domain. However, unlike our approach, theirs makes use of simple templates for generating paraphrases, as their focus is on how deferring the final choice of wording and prosodic realization to their synthesizer enables them to achieve more natural sounding synthetic speech. Following on the work described in this article, Nakatsu and White (2006) present a discriminative approach to realization ranking based on predicted synthesis quality that is directly compatible with the FLIGHTS system.

Turning to our synthesis evaluation, we note that debate over the standardization of speech synthesis evaluation continues, with the Blizzard Challenge (Black and Tokuda 2005; Fraser and King 2007) proving to be a useful forum for discussing and performing evaluation across different synthesis platforms. Mayo, Clark, and King (2005) have proposed to evaluate speech synthesis evaluation from a perceptual viewpoint to discover

exactly what subjects pay attention to, in order to ensure that evaluation actually is evaluating what we think it is. The authors found through the use of multidimensional scaling (MDS) techniques that, when asked to make judgments on the naturalness of synthetic speech, subjects made judgments relating to a number of different dimensions, including both segmental quality and prosody. Subsequently, Clark et al. (2007) found correlations between results in MDS spaces and standard mean opinion score (MOS) tests, but as the MOS tests did not correspond to single dimensions in the MDS space, they suggested that it may be possible to design more informative evaluations by asking subjects to specifically rate each factor of interest (e.g., prosody), where each factor relates to one dimension in the MDS space. However, as no specific method is suggested to guarantee reliable prosodic judgments from naive listeners, we have left this question for future research, opting instead to augment the listener ratings gathered in our perception experiment with an expert prosody evaluation and an f0 analysis of the theme phrases.

6. Conclusions and Discussion

In this article, we have described an approach to presenting user-tailored information in spoken dialogues which for the first time brings together multi-attribute decision models, strategic content planning, surface realization which incorporates prosodic features, and unit selection synthesis that takes the resulting prosodic markup into account. Based on the user model, the system selects the most important subset of the available options to mention and the attributes that are most relevant to choosing between them. To convey these trade-offs, the system employs a novel presentation strategy which makes it straightforward to determine information structure and the contents of referring expressions. During surface realization with OpenCCG, the prosodic structure is derived from the information structure in a way that allows phrase boundaries to be determined in a flexible, data-driven fashion, and with significantly higher acceptability than baseline prosody prediction models in an expert evaluation. We hypothesize that the resulting descriptions are both memorable and easy for users to understand. As a step towards verifying this hypothesis, we have presented an experiment which shows that listeners perceive a unit selection voice that makes use of the prosodic markup as significantly more natural than either of two baseline synthetic voices. Through an expert evaluation and f0 analysis, we have also confirmed the superiority of the generator-driven intonation and its contribution to listeners' ratings. In future work, we intend to examine the impact of our generation and synthesis methods on memorability or other task-oriented measures.

The present study provides evidence that it is worthwhile to investigate methods of developing general purpose synthesizers that accept prosodic specifications in their input. The main reason that we did not use such a synthesizer in our evaluation is that in order to build a general purpose Festival APML voice, suitable APML markup would be required for the 2,000–3,000 utterances that make up the core unit selection database. As these utterances are outside of the FLIGHTS domain (and thus not generated by the NLG system), it would not be possible with current technology to provide accurate APML markup for these utterances. Given the difficulty of automatically annotating general texts with APML, it may be worth considering a simplified version of the markup for the database. For example, a system which marks the location of primary phrasal stress, other pitch accents, and a simple categorization of overall tune (*wh*-question, *yes/no*-question, statement, etc.) could be used to annotate the speech database. This could be achieved with an accent detector and a very simple parser to

determine tune type. The APML specification on the input could easily be mapped to information equivalent to the database annotation by simple rules. Reasonable quality synthesis could then be achieved without the database being fully parsed and annotated. Additionally, if there is a portion of in-domain data in the database where full annotation is available, it could be used directly when those units are searched.

An interesting unresolved question is the extent to which more generic prosody prediction models, along the lines of Hitzeman et al. (1999) and Pan, McKeown, and Hirschberg (2002)—which make no use of such information structural notions as theme, rheme, and kontrast (cf. Section 2.5)—could be trained to produce tunes as distinctive as those we have targeted. We suspect that such models would have trouble doing so, given data sparsity issues and the fact that machine-learned classification models tend to discover general trends, rather than aiming to reproduce aspects of specific examples, which may contain important but rare events. At the same time, it remains for us to investigate whether our hybrid rule-based and data-driven approach can be generalized to be as flexible and widely applicable as these machine-learned models, while retaining its ability to express contrasts intelligibly. In so doing, we expect information structural constraints in the grammar to continue to play an important role.

Acknowledgments

We thank Rachel Baker, Steve Conway, Mary Ellen Foster, Kallirroi Georgila, Oliver Lemon, Colin Matheson, Neide Franca Rocha, and Mark Steedman for their contributions to the FLIGHTS system; Eric Fosler-Lussier and Craige Roberts for helpful discussion; and Julie McGory for providing the expert ToBI annotations. We also thank the anonymous reviewers for helpful comments and suggestions. This work was supported in part by EPSRC grant GR/R02450/01 and an Arts & Humanities Innovation grant from The Ohio State University.

References

Aylett, Matthew. 2005. Merging data driven and rule based prosodic models for unit selection TTS. In *5th ISCA Speech Synthesis Workshop*, pages 55–59, Pittsburg, PA.

Baker, Rachel Elizabeth. 2003. Using unit selection to synthesise contextually appropriate intonation in limited domain synthesis. Master's thesis, Department of Linguistics, University of Edinburgh.

Baldridge, Jason and Geert-Jan Kruijff. 2002. Coupling CCG and Hybrid Logic Dependency Semantics. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 319–326, Philadelphia, PA.

Bangalore, Srinivas and Owen Rambow. 2000. Exploiting a probabilistic hierarchical model for generation. In *Proceedings of COLING-00*, pages 42–48, Saarbrucken, Germany.

Bilmes, Jeff and Katrin Kirchhoff. 2003. Factored language models and general parallelized backoff. In *Proceedings of HLT-03*, pages 4–6, Edmonton, Canada.

Black, Alan W. and Keiichi Tokuda. 2005. The blizzard challenge—2005: Evaluating corpus-based speech synthesis on common datasets. In *Interspeech 2005*, pages 77–80, Lisbon.

Blackburn, Patrick. 2000. Representation, reasoning, and relational structures: A hybrid logic manifesto. *Logic Journal of the IGPL*, 8(3):339–625.

Bos, Johan, Ewan Klein, Oliver Lemon, and Tetsushi Oka. 2003. DIPPER: Description and formalisation of an information-state update dialogue system architecture. In *4th SIGdial Workshop on Discourse and Dialogue*, pages 115–124, Sapporo.

Brenier, Jason, Ani Nenkova, Anubha Kothari, Laura Whitton, David Beaver, and Dan Jurafsky. 2006. The (non)utility of linguistic features for predicting prominence in spontaneous speech. *IEEE/ACL 2006 Workshop on Spoken Language Technology*, pages 54–57, Palm Beach, Aruba.

Bulyko, Ivan and Mari Ostendorf. 2002. Efficient integrated response generation from multiple targets using weighted finite state transducers. *Computer Speech and Language*, 16:533–550.

Calhoun, Sasha, Malvina Nissim, Mark Steedman, and Jason Brenier. 2005. A framework for annotating information structure in discourse. *Proceedings of the ACL-05 Workshop on Frontiers in Corpus*

Annotation II: Pie in the Sky, pages 45–52, Ann Arbor, Michigan.
Carberry, Sandra, Jennifer Chu-Carroll, and Stephanie Elzer. 1999. Constructing and utilizing a model of user preferences in collaborative consultation dialogues. *Computational Intelligence Journal*, 15(3):185–217.
Carenini, Giuseppe and Johanna D. Moore. 2000. A strategy for generating evaluative arguments. In *Proceedings of INLG-00*, pages 47–54, Mitzpe Ramon.
Carenini, Giuseppe and Johanna D. Moore. 2001. An empirical study of the influence of user tailoring on evaluative argument effectiveness. In *Proceedings of IJCAI-01*, pages 1307–1314, Seattle, WA.
Carenini, Giuseppe and Johanna D. Moore. 2006. Generating and evaluating evaluative arguments. *Artificial Intelligence*, 170:925–952.
Carroll, John, Ann Copestake, Dan Flickinger, and Victor Poznański. 1999. An efficient chart generator for (semi-) lexicalist grammars. In *Proceedings of EWNLG-99*, pages 86–95, Toulouse, France.
Carroll, John and Stefan Oepen. 2005. High efficiency realization for a wide-coverage unification grammar. In *Proceedings of IJCNLP-05*, pages 165–176, Jeju Island, Korea.
Chung, Grace. 2004. Developing a flexible spoken dialog system using simulation. In *Proceedings of ACL '04*, pages 63–70, Barcelona.
Clark, Robert A. J. and Simon King. 2006. Joint prosodic and segmental unit selection speech synthesis. In *Proceedings of Interspeech 2006*, pages 1312–1315, Pittsburgh, PA.
Clark, Robert A. J., Monika Podsiadlo, Mark Fraser, Catherine Mayo, and Simon King. 2007. Statistical analysis of the Blizzard Challenge 2007 listening test results. In *Proceedings of Blizzard Workshop (in Proceedings of the 6th ISCA Workshop on Speech Synthesis)*, pages 1–6, Bonn, Germany.
Clark, Robert A. J., Korin Richmond, and Simon King. 2007. Multisyn: Open-domain unit selection for the Festival speech synthesis system. *Speech Communication*, 49(4):317–330.
Currie, K. and A. Tate. 1991. O-Plan: The open planning architecture. *Artificial Intelligence*, 52:49–86.
de Carolis, Bernadina, Catherine Pelachaud, Isabella Poggi, and Mark Steedman. 2004. APML, a Mark-up Language for Believable Behavior Generation. In H. Prendinger and M. Ishizuka, editors, *Life-like Characters. Tools, Affective Functions and Applications*. Springer, Berlin, pages 65–85.
Demberg, Vera and Johanna D. Moore. 2006. Information presentation in spoken dialogue systems. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06)*, pages 5–72, Trento, Italy.
Dusterhoff, Kurt E., Alan W. Black, and Paul A. Taylor. 1999. Using decision trees within the tilt intonation model to predict f0 contours. In *Eurospeech-99*, pages 1627–1630, Budapest, Hungary.
Edwards, W. and F. H. Barron. 1994. SMARTS and SMARTER: Improved simple methods for multiattribute utility measurement. *Organizational Behavior and Human Decision Processes*, 60:306–325.
Elhadad, Michael. 1993. *Using Argumentation to Control Lexical Choice: A Functional Unification Implementation*. Ph.D. thesis, Columbia University.
Foster, Mary Ellen and Michael White. 2004. Techniques for Text Planning with XSLT. In *Proceedings of the 4th NLPXML Workshop*, pages 1–8, Barcelona, Spain.
Fraser, Mark and Simon King. 2007. The Blizzard Challenge 2007. In *Proceedings of Blizzard Workshop (in Proceedings of the 6th ISCA Workshop on Speech Synthesis)*, pages 7–12, Bonn, Germany.
Ginzburg, Jonathan. 1996. Interrogatives: Questions, facts and dialogue. In Shalom Lappin, editor, *The Handbook of Contemporary Semantic Theory*. Blackwell, Oxford, pages 385–422.
Hirschberg, Julia. 1993. Pitch accent in context: Predicting intonational prominence from text. *Artificial Intelligence*, 63:305–340.
Hirschberg, Julia and Pilar Prieto. 1996. Training intonational phrasing rules automatically for English and Spanish text-to-speech. *Speech Communication*, 18:281–290.
Hitzeman, Janet, Alan W. Black, Chris Mellish, Jon Oberlander, Massimo Poesio, and Paul Taylor. 1999. An annotation scheme for concept-to-speech synthesis. In *Proceedings of EWNLG-99*, pages 59–66, Toulouse.
Hitzeman, Janet, Alan W. Black, Chris Mellish, Jon Oberlander, and Paul Taylor. 1998. On the use of automatically generated discourse-level information in a concept-to-speech synthesis system. In

Proceedings of ICSLP-98, pages 2763–2768, Sydney, Australia.
Kay, Martin. 1996. Chart generation. In *Proceedings of ACL-96*, pages 200–204, Santa Cruz, USA.
Knight, Kevin and Vasileios Hatzivassiloglou. 1995. Two-level, many-paths generation. In *Proceedings of ACL-95*, pages 252–260, Cambridge, MA.
Kruijff, Geert-Jan M. 2003. Binding across boundaries. In Geert-Jan M. Kruijff and Richard T. Oehrle, editors, *Resource-Sensitivity in Binding and Anaphora*. Kluwer Academic Publishers, pages 123–158, Dordrecht, The Netherlands.
Kruijff-Korbayová, Ivana, Stina Ericsson, Kepa J. Rodríguez, and Elena Karagjosova. 2003. Producing contextually appropriate intonation in an information-state based dialogue system. In *Proceedings of EACL-93*, pages 227–234, Budapest, Hungary.
Langkilde, Irene. 2000. Forest-based statistical sentence generation. In *Proceedings of NAACL-00*, pages 170–177, Seattle, Washington.
Langkilde-Geary, Irene. 2002. An empirical verification of coverage and correctness for a general-purpose sentence generator. In *Proceedings of INLG-02*, pages 17–24, New York, NY.
Linden, Greg, Steve Hanks, and Neal Lesh. 1997. Interactive assessment of user preference models: The automated travel assistant. In *Proceedings of User Modeling '97*, pages 67–78, Chia Laguna, Sardinia, Italy.
Marsi, Erwin. 2004. Optionality in evaluating prosody prediction. In *Proceedings of the 5th ISCA Speech Synthesis Workshop*, pages 13–18, Pittsburg, PA.
Martin, D. L., A. J. Cheyer, and D. B. Moran. 1999. The Open Agent Architecture: A framework for building distributed software systems. *Applied Artificial Intelligence*, 13(1):91–128.
Mayo, C., R. A. J. Clark, and S. King. 2005. Multidimensional scaling of listener responses to synthetic speech. In *Interspeech 2005*, pages 1725–1728, Lisbon.
Moore, Johanna, Mary Ellen Foster, Oliver Lemon, and Michael White. 2004. Generating tailored, comparative descriptions in spoken dialogue. In *Proceedings of FLAIRS-04*, pages 917–922, Miami Beach, USA.
Moore, Robert C. 2002. A complete, efficient sentence-realization algorithm for unification grammar. In *Proceedings of INLG-02*, pages 41–48, New York, NY.
Nakatsu, Crystal and Michael White. 2006. Learning to say it well: Reranking realizations by predicted synthesis quality. In *Proceedings of COLING-ACL '06*, pages 1113–1120, Sydney, Australia.
Oh, Alice H. and Alexander I. Rudnicky. 2002. Stochastic natural language generation for spoken dialog systems. *Computer, Speech & Language*, 16(3/4):387–407.
Pan, Shimei, Kathleen McKeown, and Julia Hirschberg. 2002. Exploring features from natural language generation for prosody modeling. *Computer Speech and Language*, 16:457–490.
Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: A method for Automatic Evaluation of Machine Translation. Technical Report RC22176, IBM.
Pierrehumbert, Janet. 1980. *The Phonology and Phonetics of English Intonation*. Ph.D. thesis, MIT.
Polifroni, Joseph, Grace Chung, and Stephanie Seneff. 2003. Towards automatic generation of mixed-initiative dialogue systems from Web content. In *Proceedings of Eurospeech '03*, pages 193–196, Geneva.
Pon-Barry, Heather, Fuliang Weng, and Sebastian Varges. 2006. Evaluation of content presentation strategies for an in-car spoken dialogue system. In *Proceedings of Interspeech 2006*, pages 1930–1933, Pittsburgh, PA.
Prevost, Scott. 1995. *A Semantics of Contrast and Information Structure for Specifying Intonation in Spoken Language Generation*. Ph.D. thesis, University of Pennsylvania.
Ratnaparkhi, Adwait. 2002. Trainable approaches to surface natural language generation and their application to conversational dialog systems. *Computer, Speech & Language*, 16(3/4):435–455.
Reiter, Ehud and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge.
Ristad, Eric S. 1995. A Natural Law of Succession. Technical Report CS-TR-495-95, Princeton University.
Roberts, Craige. 1996. Information structure: Towards an integrated formal theory of pragmatics. *Ohio State University Working Papers in Linguistics*, 49:91–136.
Rocha, Neide Franca. 2004. Evaluating prosodic markup in a spoken dialogue system. Master's thesis, Department of Linguistics, University of Edinburgh.

Rooth, Mats. 1992. A theory of focus interpretation. *Natural Language Semantics*, 1:75–116.

Shemtov, Hadar. 1997. *Ambiguity Management in Natural Language Generation*. Ph.D. thesis, Stanford University.

Silverman, K., M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg. 1992. ToBI: A standard for labeling English prosody. *Proceedings of ICSLP92*, 2:867–870, Banff, Canada.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the Association for Machine Translation in the Americas (AMTA-06)*, pages 223–231, Cambridge, MA.

Steedman, Mark. 2000a. Information structure and the syntax-phonology interface. *Linguistic Inquiry*, 31(4):649–689.

Steedman, Mark. 2000b. *The Syntactic Process*. MIT Press, Cambridge, MA.

Steedman, Mark. 2004. Using APML to specify intonation. Magicster Project Deliverable 2.5. University of Edinburgh. Available at `http://www.ltg.ed.ac.uk/magicster/deliverables/annex2.5/apml-howto.pdf`

Steedman, Mark. 2006. Information-structural semantics for English intonation. In Chungmin Lee, Matt Gordon, and Daniel Büring, editors, *Topic and Focus: Cross-Linguistic Perspectives on Meaning and Intonation*. Springer, Dordrecht, pages 245–264.

Stolcke, Andreas. 2002. SRILM — An extensible language modeling toolkit. In *Proceedings of ICSLP-02*, pages 901–904, Denver, Colorado.

Taylor, Paul and Alan Black. 1998. Assigning phrase breaks from part-of-speech sequences. *Computer Speech and Language*, 12:99–117.

Taylor, P., A. Black, and R. Caley. 1998. The architecture of the the Festival speech synthesis system. In *Third International Workshop on Speech Synthesis*, pages 147–151, Sydney.

Theune, Mariët. 2002. Contrast in concept-to-speech generation. *Computer Speech and Language*, 16:491–531.

Theune, Mariët, Esther Klabbers, Jan-Roelof de Pijper, Emiel Krahmer, and Jan Odijk. 2001. From data to speech: A general approach. *Natural Language Engineering*, 7(1):47–86.

Vallduví, Enric and Maria Vilkuna. 1998. On rheme and kontrast. In Peter Culicover and Louise McNally, editors, *Syntax and Semantics, Vol. 29: The Limits of Syntax*. Academic Press, San Diego, CA, pages 79–108.

van Deemter, Kees, Emiel Krahmer, and Mariët Theune. 2005. Real versus template-based natural language generation: A false opposition? *Computational Linguistics*, 31(1):15–23.

van Santen, Jan, Alexander Kain, Esther Klabbers, and Taniya Mishra. 2005. Synthesis of prosody using multi-level unit sequences. *Speech Communication*, 46(3–4):365–375.

Walker, M. A., S. Whittaker, A. Stent, P. Maloor, J. D. Moore, M. Johnston, and G. Vasireddy. 2002. Speech-plans: Generating evaluative responses in spoken dialogue. In *Proceedings of INLG '02*, pages 73–80, New York, NY.

Walker, M. A., S. J. Whittaker, A. Stent, P. Maloor, J. D. Moore, M. Johnston, and G. Vasireddy. 2004. Generation and evaluation of user-tailored responses in multimodal dialogue. *Cognitive Science*, 28:811–840.

Walker, Marilyn A., Rebecca Passonneau, and Julie E. Boland. 2001. Quantitative and qualitative evaluation of DARPA Communicator spoken dialogue systems. In *Proceedings of ACL-01*, pages 515–522, Toulouse, France.

Walker, Marilyn A., Owen C. Rambow, and Monica Rogati. 2002. Training a sentence planner for spoken dialogue using boosting. *Computer Speech and Language*, 16:409–433.

White, Michael. 2004. Reining in CCG Chart Realization. In *Proceedings of INLG-04*, pages 182–191, Brockenhurst, UK.

White, Michael. 2006a. CCG chart realization from disjunctive inputs. In *Proceedings of INLG-06*, pages 12–19, Sydney, Australia.

White, Michael. 2006b. Efficient realization of coordinate structures in Combinatory Categorial Grammar. *Research on Language & Computation*, 4(1):39–75.