Evaluating the impact of variation in automatically generated embodied object descriptions

Mary Ellen Foster

Doctor of Philosophy

Institute for Communicating and Collaborative Systems
School of Informatics
University of Edinburgh

2007

Abstract

The primary task for any system that aims to automatically generate human-readable output is *choice*: the input to the system is usually well-specified, but there can be a wide range of options for creating a presentation based on that input. When designing such a system, an important decision is to select which aspects of the output are hard-wired and which allow for dynamic variation. Supporting dynamic choice requires additional representation and processing effort in the system, so it is important to ensure that incorporating variation has a positive effect on the generated output.

In this thesis, we concentrate on two types of output generated by a multimodal dialogue system: linguistic descriptions of objects drawn from a database, and conversational facial displays of an embodied talking head. In a series of experiments, we add different types of variation to one of these types of output. The impact of each implementation is then assessed through a user evaluation in which human judges compare outputs generated by the basic version of the system to those generated by the modified version; in some cases, we also use automated metrics to compare the versions of the generated output.

This series of implementations and evaluations allows us to address three related issues. First, we explore the circumstances under which users perceive and appreciate variation in generated output. Second, we compare two methods of including variation into the output of a corpus-based generation system. Third, we compare human judgements of output quality to the predictions of a range of automated metrics.

The results of the thesis are as follows. The judges generally preferred output that incorporated variation, except for a small number of cases where other aspects of the output obscured it or the variation was not marked. In general, the output of systems that chose the majority option was judged worse than that of systems that chose from a wider range of outputs. However, the results for non-verbal displays were mixed: users mildly preferred agent outputs where the facial displays were generated using stochastic techniques to those where a simple rule was used, but the stochastic facial displays decreased users' ability to identify contextual tailoring in speech while the rule-based displays did not. Finally, automated metrics based on simple corpus similarity favour generation strategies that do not diverge far from the average corpus examples, which are exactly the strategies that human judges tend to dislike. Automated metrics that measure other properties of the generated output correspond more closely to users' preferences.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Mary Ellen Foster*)

Acknowledgements

First of all, I must thank my supervisors for their support and advice throughout what seemed like a never-ending process: Jon Oberlander for his unfailing enthusiasm and optimism and his amazing ability to generalise from experimental results, and Johanna Moore for making sure that my ideas were always well-grounded and well-supported. Special thanks to Jon for his help with the logistics of getting this thesis completed and submitted from abroad.

Many thanks also to my examiners, Donia Scott and Jean Carletta. Their many useful comments and concrete suggestions have helped to make this thesis a much more complete and coherent piece of research.

Most of the work described in this thesis was done during my time working on the COMIC project (IST-2001-32311). I am grateful to all of my colleagues for such a pleasant and productive introduction to the world of EU-funded research. In particular, I thank Michael White, my office-mate and colleague for three years, who taught me a great deal about doing practical research in computational linguistics, and in particular about carrying out publication-worthy work and then actually getting it published.

Thanks to Colin Fraser for providing an excellent bathroom-salesman impersonation, and to Amy Isard, Ron Petrick, and especially Tom Segler for invaluable annotation assistance. Thanks also to all of the participants in all of the human evaluations described in the thesis.

Two websites were particularly useful during the later stages of this thesis: the Language Experiments Portal has been an excellent resource for recruiting experiment participants, while the increasingly comprehensive archive at the ACL Anthology has been very useful for locating relevant papers and articles. Thanks to Frank Keller, Steven Bird, and their respective teams for providing and maintaining these sites.

I won't even attempt to list all of the other people who have been helpful, personally and professionally, during my time working on this thesis, because I'm sure I'll forget someone if I try. If you believe you belong on this list, consider yourself thanked as well.

Finally, two special acknowledgements:

To my parents: my mother Margaret and my father Leslie, who introduced me to the joy to be found in languages and computing, respectively, and who always made me believe I could do anything I wanted to.

And to Ron, for unwavering in-person, cross-Channel, and transatlantic support, including (but by no means limited to) annotation and proofreading assistance, late-night laptop transplant surgery, excellent cooking, and above all just being there whenever I needed him in any way. I look forward to the day when we can finally both live in the same place for good.

This thesis is dedicated to my father

Leslie A. Foster

31 December 1943 – 22 August 2005

Table of Contents

It doesn’t matter how beautiful your theory is, it doesn’t matter how smart you are. If it doesn’t agree with experiment, it’s wrong.

Richard P. Feynman

Chapter 1

Introduction

> Everything starts somewhere, though many physicists disagree.
>
> Terry Pratchett, *Hogfather*

A GENERATION SYSTEM is a computer system that transforms its inputs into an output presentation that includes linguistic content. The core task in this area is *natural language generation*: using techniques from artificial intelligence and linguistics to automatically create text in a human language based on some non-text specification. A growing class of *multimodal generation* systems employ the same basic techniques to automatically generate presentations combining information on several output channels. A related set of applications use similar output-producing components, but apply them to textual input: this set includes machine translation, summarisation, and paraphrasing systems.

The primary task for any such generation system is *choice*: the input tends to be unambiguous, well-specified, and well-formed, but there is often a range of options for creating output based on that input. This contrasts with tasks such as natural-language understanding, where the main consideration is determining the most probable interpretation of input that is often ambiguous, under-specified, and ill-formed. When automatically generating output, choices must be made at all levels, from the initial, high-level selection of the content to be included to the low-level selection of the words, facial expressions, or graphical techniques to use. In many cases, the correct action can be selected by a default rule; that is, successful output (for a particular definition of success) can be produced when the system makes the same choice under all circumstances. However, there are also situations in which making context-sensitive choices—or even randomly varying the output—can produce output that increases user satisfaction, task performance, or some other measure of quality.

In the design of any generation system, there is a trade-off between these two implementation techniques. Making default choices is generally a strategy that is easier to implement and requires less processing at run-time; in contrast, supporting variation requires both that the extra contextual information be represented and that additional rules be created to make use of the context. Choosing which parts of the output are to be hard-wired and which allow for dynamic variation is therefore a vital aspect of the design of any generation system. It is important to ensure that any dynamic choice actually has an impact on the generated output. This can only be assessed by comparing the quality of presentations generated with and without the additional variation.

In this thesis, we explore the nature of this trade-off between representation and processing effort on one hand, and the quality of the resulting output on the other hand. We concentrate on the generation of content in two of the output modalities of an embodied spoken-language dialogue system: the linguistic content of object descriptions, and the conversational facial displays of the animated talking head. We implement a number of techniques for adding variation to both of these output channels. For each implementation, we ask human judges to compare the output generated by the enhanced system to that created by a baseline (no-variation) version of the same system. In several cases, we also evaluate the output using automated metrics and compare the results with those of the human evaluation.

This series of implementations and evaluations allows us to address three related questions about the impact of different types of variation on both human and automated evaluations of generated output; Section 1.1 summarises these questions and discusses how the thesis deals with each. The remainder of this chapter then introduces the tools that we use to address these questions. In Section 1.2, we give an overview of the COMIC multimodal dialogue system, which is the context for all of the implementations and experiments in the thesis. We then present the techniques that we use for the two main tasks of this thesis: incorporating variation into a generation system (Section 1.3) and evaluating the resulting output (Section 1.4). Finally, in Section 1.5, we give an overview of the remaining chapters of the thesis.

1.1 Research questions

As described above, the overall goal of this thesis is to explore the trade-off between the additional representation and processing effort involved in making dynamic choices in a generation system on the one hand, and the quality of the resulting output on the other hand. Concretely, this goal can be broken down into three related research questions: whether users can detect generated output that has been tailored to the current discourse context, and whether they prefer such output; whether users prefer output generated by making a rule-based choice or a weighted random choice; and whether the results of automated metrics agree with the preferences of human judges when evaluating generated output including variation. The remainder of this section discusses each of these questions in detail.

1.1.1 Evaluating the impact of variation

It is often claimed (e.g., Reiter, 1995) that one of the advantages of generating output dynamically is that a system is able to produce more natural and varied output; however, few studies have directly measured the impact of such variability on the perceived quality of the output, or even whether users notice the tailoring at all. All of the human evaluations in this thesis address this question in some way. In each study, we introduce a particular type of variation into an output-generation system, and then test the impact of the implementation via a comparative user evaluation. The experiments fall into two high-level categories. In some cases, we measure whether users are able to determine the intended tailoring based on the generated output; in the others, we ask users to select their preferred version among minimal pairs of output generated with and without the modification. We investigate this question using two presentation modalities: natural-language descriptions of objects and non-verbal behaviour of an embodied agent.

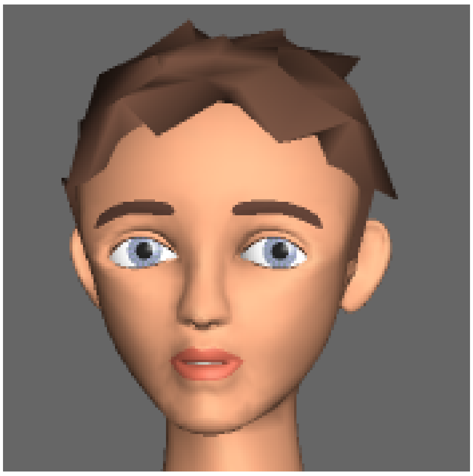
1.1.2 Comparing implementation strategies

At a high level, there are two methods that can be used to make choices when automatically generating output. On the one hand, the system can make use of rules; these rules may be written by hand, or may be derived from a corpus of target outputs. On the other hand—particularly if such a corpus of target outputs is available—the system may instead make a random choice, possibly weighted by the frequency of different options in the corpus. Section 1.3 discusses both of these techniques in more detail. Each of these techniques has been widely used, both in text generation and—in particular—for choosing behaviour for embodied agents. In this thesis, we explore both implementation techniques, both for generating text and—more extensively—for selecting the behaviour of an embodied agent; in each case, we gather human judges' opinions of the generated output.

1.1.3 Comparing evaluation techniques

Recent proposals for shared tasks and common evaluation metrics for generation have made the issue of evaluation a topic of lively debate in the generation community. A range of automated metrics have been used to evaluate generation systems, some of which have been

Figure 1.1: COMIC tile-design description



"Here is another design. There are geometric shapes on the decorative tiles, but the tiles are from the Armonie series. Once again the tiles are by Steuler, but here it is in the classic style."

shown to correlate with human judgements on particular aspects of the output; however, it is still not known to what extent automated measures can be used to supplement or replace human judgements of output quality. The experiments in this thesis provide us with directly comparable results from human-based evaluations and a range of automated metrics on several types of output. Comparing these results gives an indication of the circumstances under which different types of automated evaluation metrics do and do not agree with human judgements of output quality.

1.2 The COMIC multimodal dialogue system

The implementations in this thesis all consist of modifications to the output-generation components of the COMIC multimodal dialogue system. This system adds a multimodal talking-head interface to an existing CAD-style application used in bathroom sales situations to help clients redesign their rooms. The output combines synthesised speech, non-verbal behaviour of an animated talking head, deictic gestures using an on-screen pointer, and direct control of the underlying application. Section 3.1 gives an overview of the components and architecture of the full COMIC system.

We concentrate on those output turns in which the embodied agent describes and compares tile-design options to the user, as those are the turns in which the output-planning process is the most dynamic and the most open to adding variation; Figure 1.1 shows a sample description of this type. At a high level, the output planner uses the classic generation "pipeline" architecture described in Section 2.1.2 to translate tile-design facts stored in a database into the textual and multimodal content of turns like that shown in Figure 1.1. The facial displays of the embodied agent are used to reinforce the messages communicated in the speech, for example by nodding on emphasised words. Section 3.2 gives a detailed description of the process of creating output of this type.

The style of output generated by COMIC is similar to that of many previous and current systems that dynamically generate descriptions and comparisons of objects drawn from a database; in fact, this style of generation goes back to the pioneering TEXT system (McKeown, 1985). Similar recent systems include M-PIRO (Androtsopoulous *et al.*, 2007), which generates textual descriptions of museum objects, and FLIGHTS (Moore *et al.*, 2004), which creates user-tailored descriptions of airline flights. In Section 2.3, we discuss a number of systems of this type in more detail.

Using the non-verbal behaviour of an embodied agent to enhance human-computer interaction is also a technique that is used in a number of systems. The non-verbal channel can be used both to convey information and to help regulate the flow of a conversation by providing social signals, and has been used in systems including REA (Cassell *et al.*, 2001a), Greta (de Carolis *et al.*, 2002), and the virtual snowboarding instructor implemented by Stone *et al.* (2004); Section 2.4.2 summarises research in this area.

COMIC is based on a well-known architecture, uses standard components, and produces output that is similar to that produced by many other systems both in its linguistic content and in its use of the embodied agent. This means that, while the results of the experiments in this thesis are based on modifications to the COMIC output-generation process, they also apply to a wide range of other systems. The issue of generalisability is discussed in Section 3.3 and again in Chapter 10.

1.3 Adding variation to automatically generated output

In this thesis, we use the COMIC output-generation system as the basis for implementing and testing a range of methods for adding variation to generated output. In its basic form, the COMIC generator implements a one-to-one mapping between its inputs and its outputs. This is illustrated in Figure 1.2: given a specific input, a basic generation system will always select the same presentation content. In this section, we present the two basic implementation techniques that we used to add variation to this basic generation process: *rule-based variation* and *stochastic variation*. A range of variation types and implementation techniques are discussed in more detail in Section 2.5.

A basic generation system, as illustrated in Figure 1.2, takes into account only the high-level communicative goal (e.g., *describe design #15*) when choosing how to create its output, and

Figure 1.2: The basic generation process

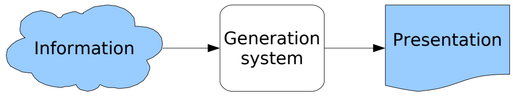
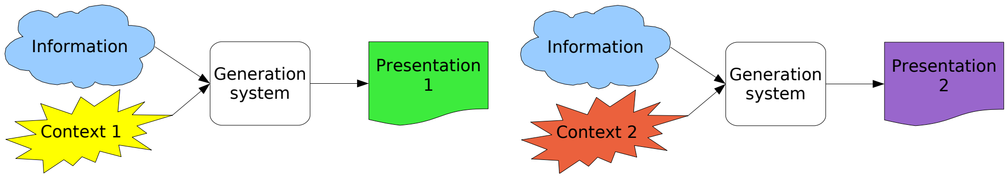


Figure 1.3: Rule-based variation



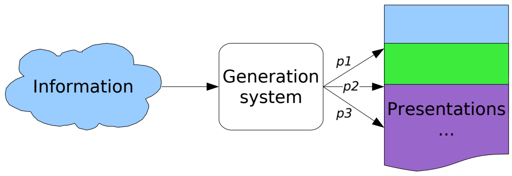
always chooses the same presentation for any given goal. When rule-based variation is introduced, the system also considers the circumstances under which that information is to be presented, as is illustrated in Figure 1.3. The context may include the history of the discourse, a model of the user's preferences or capabilities, or rhetorical, interpersonal, or stylistic goals of the system. Choosing to make use of the additional contextual information is a decision with impact on the overall design of the system: the contextual information must be represented and updated, and extra rules are required to integrate it into the generation process.

In COMIC, two main sources of contextual information are used: the history of the dialogue and a model of the user's likes and dislikes. Rule-based variation is implemented within COMIC and evaluated in the experiments described in Chapters 4, 7,8, and 9.

With rule-based variation, the notion of system input is generalised to include the context in which it is to be presented. If a system is implemented using only rule-based variation, it is still possible to predict the exact output it will give in any given situation, as long as the full context is known. In contrast, stochastic variation adds an element of nondeterminism to the generation process: as illustrated in Figure 1.4,[1] when this technique is used, the system makes a stochastic choice among a range of possibilities in a given situation. Section 2.5.4 describes several systems that have used this implementation technique.

[1] In the figure, p_1, p_2, p_3 represent the probabilities of choosing each option.

Figure 1.4: Stochastic variation



In this thesis, stochastic variation is used when we have a corpus describing a wide range of possible alternatives. Writing deterministic rules to cover these cases would either discard a large percentage of the options from the corpus, or else require a prohibitively large number of rules, so stochastic variation is useful. Two implementations make use of this technique: in Chapter 5, we choose randomly from among the top n paraphrases in order to introduce variation into a sequence of textual descriptions; in Chapters 8 and 9, we use this technique to select the set of facial displays to use in a range of user-model and syntactic contexts.

Stochastic variation is generally used in conjunction with rule-based variation. In the experiment described in Chapter 5, for example, the dialogue history is used both to constrain the syntactic structures that are available and to modify the probabilities of those that remain, and the system then selects from among that revised set. Various aspects of the context are used in a similar way for the weighted face-display selection evaluations in Chapters 8 and 9.

1.4 Evaluating generated output

When experimenting with generation techniques, it is vital to determine whether any implemented modifications make a difference to the generated output. Section 2.6 surveys the techniques that have been used to evaluate the output of a range of generation systems and summarises the current debate regarding the role of shared-task evaluations. In this section, we discuss the specific evaluation methods employed in this thesis.

Two main techniques have been used to evaluate the output of generation systems: automated metrics based on a corpus of target outputs, and techniques involving human judges. If a corpus is available, automated evaluations are simpler to carry out than human evaluations, and do not involve the overhead of recruiting participants and analysing the resulting data. Section 2.6.2 describes a number of generation systems that have been evaluated using automated techniques.

However, as generation is an open-ended task, often alternatives other than those that actually occur in the corpus can also be equally valid options but will tend to be scored lower by automated metrics that assess corpus similarity. In particular, when the system being evaluated is one that aims to produce variation, automated metrics based on corpus similarity will tend to score its outputs lower than one that uses no variation, while several recent studies of text-generation systems (e.g., Stent *et al.*, 2005; Belz and Reiter, 2006) have found that human participants actually prefer outputs that diverge from the "average" corpus outputs. Other metrics can also be computed automatically—for example, Mellish *et al.* (1998) compared the number of facts conveyed in the output of different text-structuring strategies—but again it is not clear how such automated metrics relate to user preferences in a given context. For this reason, although we perform several automated evaluations in this thesis, they are always accompanied by a human evaluation.

Most generation systems are designed to achieve particular communicative or task-based goals or are embedded in larger systems that have such goals. The most complete demonstration of the success of a generation technique is a task-based, comparative evaluation: showing that the system achieves its goals significantly better with the technique enabled than it does with it disabled. Previous task-based evaluations of generation systems include Cox *et al.* (1999), Carenini and Moore (2006), Karasimos and Isard (2004), Elhadad *et al.* (2005), and Di Eugenio *et al.* (2005); the details are given in Section 2.6.1.

However, running this sort of task-based evaluation presents a number of practical problems when evaluating the output components of a multimodal dialogue system such as COMIC. First, to run the full COMIC dialogue system requires a number of computers working together, some with very specific hardware or software. Also, even with state-of-the-art components, the input-processing subsystem of an end-to-end dialogue system—especially when speech recognition is included—can introduce many errors and misunderstandings. If the goal is to study specific adaptations to the system output, there is a danger that the modifications of interest can often be missed in the larger process of attempting to interact with the system; this was largely the case in the one full task-based evaluation of COMIC that was undertaken (White *et al.*, 2005), for example. Finally, for a true task-based evaluation, it is necessary to find participants who can realistically be expected to perform the task at hand; for COMIC specifically, recruiting large numbers of participants who have direct experience with designing and buying bathrooms is non-trivial.

In this thesis, we examine the individual impact of a number of sometimes subtle modifications to a multimodal output-generation process. For the reasons outlined above, we chose not to do task-based evaluations, but instead to ask our participants for direct judgements of the output quality. The judgements include selecting the version of the output that is most appropriate to a given conversational situation, determining the affective content of an utterance, or simply choosing which version they like better. The materials for all of the studies were generated in advance by the full COMIC output system and were then played back or presented as text to the participants, which allowed studies to be run both over the world-wide web and using an output-only version of the COMIC system. This sort of direct-evaluation paradigm has been successfully used in previous studies including Hartley *et al.* (2000) Walker *et al.* (2002), Walker *et al.* (2004), and Belz and Reiter (2006); again, see Section 2.6.1 for the details.

1.5 Overview of the thesis

The body of this thesis is divided into three parts. First, Chapters 2 and 3 together describe the context of this research. Chapters 4–9 then describe a series of implementations and evaluations designed to address the research questions outlined in Section 1.1. Finally, Chapter 10 combines the findings from all of the experimental studies to propose answers to the research questions, and also describes possible extensions to the work described here. The rest of this section gives a more detailed summary of the content of each individual chapter.

Chapter 2 surveys past and current work in a range of relevant fields: the high-level process of generating output, the roles of corpora in generation, the automated generation of descriptions of database objects, the features of embodied language, several methods of adding variation to the generation process, and the main techniques for evaluating the output of generation systems. At the end of this chapter, we describe how the research of this thesis fits in with and contributes to each of these fields.

Chapter 3 then describes in detail the COMIC multimodal dialogue system introduced in Section 1.2, first giving an overview of the architecture and components of the entire system and then concentrating specifically on how dynamic multimodal object descriptions are specified and generated.

The generation subsystem of COMIC is then used as the experimental platform for the main body of the thesis in Chapters 4–9. These chapters describe a series of enhancements to the basic multimodal output-production process outlined in the second half of Chapter 3, each designed to introduce a different type of variation to the generated output. For each implementation, we describe the results of an evaluation in which human judges were asked to compare output generated with and without the implemented variation; in some cases, we also present the findings of an automated evaluation and compare them with the results of the human studies. The first two studies concentrate on the linguistic content of the object descriptions, while the rest focus on the non-verbal behaviour of the embodied agent.

In **Chapter 4**, we describe how the COMIC text-generation process was extended to use information from two different sources: the history of the dialogue and a model of the user's likes and dislikes. We then present two "overhearer"-style evaluations in which participants were asked to select the system output that was more appropriate for a hypothetical user and conversational situation: one study concentrated on the dialogue-history modifications, while the other dealt with the user-preference adaptation. Participants in both studies generally found the versions that were generated with contextual tailoring to be more appropriate for the hypothetical user. However, they sometimes missed adaptations presented in speech that they perceived in the written text, and were not able to distinguish between descriptions intended to be neutral and those intended to be positive.

Next, in **Chapter 5**, we explore the impact of two different techniques of avoiding syntactic repetition in generated text by selecting different paraphrases in the surface realiser: choosing options with *n*-gram scores below the top option, and penalising words from the immediately preceding discourse. We performed evaluated these techniques in two ways. First, we carried out a forced-choice evaluation comparing texts generated with and without these techniques. The results of this study indicate that participants found descriptions generated using these techniques better written and less repetitive than those generated without them, and did not report any difference in understandability. Second, we used automated metrics to assess the impact of both of these techniques on several features of the generated text. These metrics show that both techniques tended to increase variability of the texts as measured by edit distance and to decrease the corpus-based *n*-gram scores, and that increasing the threshold score below which options could be selected also tended to introduce more dispreferred substrings.

In Chapters 6–9, we turn our attention to the other major modality of the output generated by COMIC: the conversational facial displays of the animated talking head. **Chapter 6** describes the process of collecting and annotating a corpus of facial displays based on the non-verbal behaviours of a single speaker reading a range of sentences in the COMIC domain. At the end of the chapter, we analyse the influence of a number of contextual factors on the facial displays used by the speaker.

The single factor with the most influence on the recorded speaker's facial displays was the user-preference evaluation, which affected the distribution of all of the facial displays. In **Chapter 7**, we describe a simple rule-based method of selecting facial displays based on the user-model evaluation. We then describe two studies designed to gathered user responses to videos generated using this rule: the first assessed the recognisability of the displays, while the second measured whether consistency between the verbal message and the facial displays was preferred. The participants in the first experiment were generally able to identify the intended user-model polarity of a sentence based on the accompanying displays, although again—as in Chapter 4—they tended to confuse outputs intended to be neutral with those intended to be positive. In the second experiment, the participants generally preferred outputs with facial displays consistent with the linguistic message over those with inconsistent displays.

The generation system described in Chapter 7 used a simple corpus-derived rule to select the facial displays to accompany speech. In **Chapter 8**, we present two implementation techniques that make direct use of the corpus data to choose the facial displays depending on the context: always choosing the majority corpus option, or making a weighted choice among all of the possibilities. The facial-display schedules generated by these two strategies were compared to each other and to those generated by the rule-based strategy from Chapter 7 using a range of automated metrics and through human evaluation. The results of the automated measures vary. On metrics based on corpus similarity, the majority-choice strategy scored the highest and the rule-based strategy the lowest. On the other hand, on metrics that count the number and range of facial displays generated, the output of the weighted-choice strategy was most similar to the corpus, while the rule-based strategy was the least similar. Finally, when the distribution of facial displays was measured via standard deviation, the output of the majority-choice strategy was most different from the corpus. In a human evaluation, the judges strongly preferred the output of the weighted-choice strategy to that of the majority-choice strategy, while the judges in a second study showed no significant preference between the weighted-choice and rule-based strategies; on both studies, there was a weakly significant preference for resynthesised versions of the original corpus displays over any of the generated versions.

The final experimental study in **Chapter 9** brings together materials and techniques from several of the previous chapters. In this study, we reused the materials from the evaluation of the user-preference-based textual tailoring from Chapter 4, but added the animated talking head to the presentation. Just as in Chapter 4, judges were asked to identify output that was correctly tailored to the preferences of a hypothetical user. We compared the influence on this task of face-display schedules generated by both the rule-based and the weighted-choice strategies. The results demonstrate that participants who used the rule-based schedules performed as well on this task as those from Chapter 4 who used the speech-only presentation, while the performance of participants who used the weighted-choice schedules was significantly worse.

In **Chapter 10**, we consider the results of all of the studies described in this thesis and draw conclusions regarding the questions set out in Section 1.1. We also discuss the implications of these results for other generation systems, and propose possible extensions to this research. In brief, the responses to the research questions are as follows. (1) The human judges were able to perceive contextual tailoring in generated output except in cases where other aspects of the output obscured it or the variation was not marked, and also preferred output that exhibited a wider range of output possibilities to output that was drawn from a narrower selection. (2) For textual output, a system that selected options other than the top one from a corpus was preferred over one that selected the top option. For embodied-agent output, the results were mixed. As with text, users clearly disliked output generated by taking the highest-probability option based on a corpus. When the choice was between simple corpus-derived rules and corpus-based weighted choice, they had a mild subjective preference for the weighted outputs; however, the weighted outputs decreased their ability to identify contextual tailoring in the speech, while the rule-driven displays did not. (3) Automated metrics based on simple corpus similarity favour generation strategies that do not diverge far from the average corpus examples, which are exactly the strategies that human judges tend to dislike. Automated metrics that measure other properties of the generated output correspond more closely to users' preferences.

Chapter 2

Background

“But ye gotta know *where* ye’re just gonna rush in.
Ye cannae just rush in *anywhere*. It looks bad,
havin’ to rush oout again straight awa’.”

Terry Pratchett, *The Wee Free Men*

TO ADDRESS the research questions of this thesis, we modify the generation process of a multimodal dialogue system to incorporate variation into two of the output channels (textual descriptions and embodied-agent behaviours), using a range of implementation strategies, and measure the impact of each modification both by gathering the judgements of human users and by computing automated evaluation metrics. This set of tasks draws from on results and techniques from a number of relevant research areas; in this chapter, we discuss the existing work and open issues in each of these areas.

We begin in Section 2.1 with an overview of the process of automatically generating dynamic output: we describe the input resources that are normally used by such a system, present common architectures that have been used, and end with a discussion of the issues specific to the generation of coordinated content across multiple output channels. In Section 2.2, we turn our attention to the increasingly important role of annotated corpora in generation. We first describe the different ways that data from a corpus has been used to help make decisions in generation systems, and then summarise the state of the art in the currently active field of multimodal corpora.

In the next two sections, we concentrate specifically on the types of output that are produced by the multimodal dialogue system that is the focus of the experiments in this thesis. In Section 2.3, we survey the task of describing and comparing objects drawn from a database, which is a style of generation that dates back to some of the earliest text-generation systems.

In Section 2.4 we discuss the use of non-verbal behaviours in human-human communication and describe how those behaviours have been reproduced in embodied artificial agents.

After that, we present the approaches that have been taken to the two primary tasks of this thesis. In Section 2.5, we describe the main sources and techniques that have been used to incorporate variation into the output of a generation system. Section 2.6 then surveys current approaches to evaluating the output of generation systems, both human evaluations and automated metrics, and summarises the current debate regarding the role of shared-task evaluations.

At the end of this chapter, in Section 2.7, we summarise the research described in this section and describe how the experiments in this thesis both fit in with and extend the existing work in these areas.

2.1 The generation process

Recall that a generation system is a computer system that transforms its inputs into an output presentation that includes linguistic content. Generation—particularly of text, but also of various forms of multimodal output—is a field that has been well-studied for several decades, and enough systems have been built to address this task that it can be discussed in fairly abstract terms. In Section 2.1.1, we first describe the types of goals and knowledge resources that can be used to specify the input to a generation system. We then discuss common architectures for generation systems in Section 2.1.2. Finally, in Section 2.1.3, we give an overview of the specific issues that arise when generating coordinated output across multiple output channels.

2.1.1 Input specification

In theory, generation is the inverse of understanding, and it is tempting to use the same representations for the output of an understanding system and the input to a generation system. However, in practice the needs of the two system types are very different: a comprehension system often extracts less information from a presentation than a generation system would require to create it. In a much-quoted analogy, Wilks (1990) compares comprehension to the task of counting from zero to infinity, while generation is compared to counting from infinity to zero.

The exact nature of the input to a generation system therefore tends to be tailored to the specific needs of this task, and is often application-specific. The following is one characterisation of the distinct components of the input to a generation system (Reiter and Dale, 2000):

Communicative goal The purpose of the output to be generated. Note that this is distinct from—and more specific than—the overall purpose of the generation system; it describes the goal for a specific piece of output.

Knowledge source The information about the domain that is available to the system, typically encoded in one or more databases and knowledge bases.

User model A characterisation of the audience for whom the output is to be generated.

Discourse history A record of what has previously been communicated to the user.

The communicative goal is the primary input to the system, while the other information sources provide information that can be used to help make choices while creating output to satisfy the goal. Not every system makes use of all of these information types: for example, not all systems incorporate a model of the user, and if a system is designed to generate stand-alone outputs it has no need to keep track of the discourse history.

As Dale (1993) points out, there is an additional factor that has a significant influence on the output produced by a generation system: the implicit assumptions hard-wired into the system's processing. It is these assumptions that determine what levels of variation are possible in a particular system. For example, if a text-generation system never considers any tense other than the present, or an information-graphics system always produces bar charts, then there is *a priori* no opportunity for any dynamic choice on those dimensions. The process of designing a generation system is largely one of choosing which of the decisions are hard-coded and which allow for dynamic choice, and how.

2.1.2 Generation architectures

In their survey of natural language generation (NLG), Reiter and Dale (2000) outlined a high-level "pipeline" architecture that is common to many systems that generate natural-language texts. As shown in Figure 2.1, the pipeline architecture casts a generation system as a transducer that converts machine-readable information into human-readable presentations through a series of three steps, where the output of each step is the input to the next. The steps are as follows:

Content planning Selecting the domain-specific information to be presented and giving it a high-level rhetorical structure.

Microplanning Mapping the content plan into specifications for the output components.

Surface realisation Making specific decisions in each output component and producing the actual output.

Figure 2.1: The generation pipeline (Reiter and Dale, 2000)

As a concrete example of these steps, consider a system designed to generate textual summaries of time-series data. The input to the system would be the numerical data, and possibly a particular communicative goal such as *describe the recent trend*; the content-planning component would analyse the data and choose the relevant features to include; the microplanning component would structure the selected content into sentences and select the lexical items to be used; while the surface-realisation component would create grammatical sentences with fully-inflected words to present to the user.

The divisions between the different steps in the pipeline are not always as clear-cut as in the above example and do not always map cleanly onto components of a specific generation system. When Cahill *et al.* (1999) analysed a number of existing NLG systems and attempted to map them onto a similar architecture, they found that quite often the dividing lines between component tasks were different between systems, making it difficult to specify a detailed functional description of any individual component.

As well, in an actual implementation, the data flow may be more complex than the unidirectional pipeline shown in Figure 2.1: in some cases, processing may also flow from "downstream" components back to those earlier in the pipeline. For example, if a system must produce a textual document of a specific length, it is not until the surface realisation stage that the length can be measured. If the document is then found to be too long or too short, the realiser might send a message back to the content planner to adjust the amount of content as necessary. Bontcheva and Wilks (2001) and Reiter *et al.* (2003) both used this technique to create texts under such space constraints, for example.

An alternative view of the generation process is provided by the Reference Architecture for Generation Systems (RAGS) (Mellish *et al.*, 2006). Instead of focusing on the tasks, this architecture instead provides a language for describing the interfaces between components. The goal is to facilitate the sharing of data, components, and techniques by providing a common vocabulary and reference point, thereby allowing scientific discussion to be focused on genuine points of theoretical disagreement.

RAGS provides a *data model* that describes NLG data structures at two levels, based on current practice in implemented systems. The high-level specification divides data into six basic types with different linguistic roles: Conceptual, Rhetorical, Document, Semantic, Syntactic, and Quote. These levels are defined in terms of abstract type definitions specifying the structure of the representation, independent of any particular implementation. The low-level "objects and arrows" model provides a reference implementation of the high-level type definitions that formally specifies the data states that can be communicated between NLG modules during the generation process.

RAGS also includes a proposed XML syntax that can be used to store data sets offline (Cahill *et al.*, 2001), again with the goal of facilitating sharing and re-use. Mellish *et al.* (2006) defined a possible set of modules for frequently-used operations in an NLG system, specifying the input and output of each in terms of RAGS data structures: for example, the input to a lexical-choice module is a semantic representation, while the output is a syntactic representation.

2.1.3 Multimodal generation

Our definition of a generation system is one that creates output *including* language; in this section, we discuss the issues that arise when coordinated output is created on several output channels. Multimodal generation can take several forms: for example, the types of output described at a recent workshop on the topic (Theune *et al.*, 2007) include embodied conversational agents (Marsi and van Rooden, 2007), illustrated documents (Bateman and Henschel, 2007), and information graphics with captions (Habel and Acartürk, 2007).

The selection of behaviours for embodied agents is addressed in detail in Section 2.4.2. In this section, we give an overview of the issues in multimodal generation as a whole, based on the summaries of Maybury (1995) and André (2000, 2003). There are two main additional, related issues that arise when creating output combining content on several channels: *modality selection* and *output coordination*.

2.1.3.1 Modality selection

Modality selection[1] is the task of assigning parts of the presentation to the available output channels. Information that can be used to help make this choice includes the output channels, the information to be presented, the goals of the presenter, the characteristics of the user, the intended user task, and any limitations on available resources.

One possibility for modality selection is to choose the content on each channel independently: if language is a primary output channel, the desired content can be expressed linguistically, with the other modalities providing supporting content as they are able. However, this simple selection technique does not take advantage of the ways in which content on the different output channels can be complementary. Piwek *et al.* (2005) confirmed that this is not the method that graphic designers generally use in practice; instead, they plan the parts of a

[1] This task is often called *media allocation*.

document together, and may even create the graphical elements before any text is written. When it comes to embodied behaviour, there is also evidence (e.g., de Ruiter, 2007) that speech and body language together come from a single source, rather than being planned independently.

To create high-quality output, therefore, a multimodal generation system should select the output to be presented on its output channels in a coordinated process. Early multimodal-generation systems (e.g. Feiner and McKeown, 1991; André and Rist, 1993) relied on a classification of the input data, and used media-allocation rules to map from information types and communicative functions onto media classes. Arens *et al.* (1993) divided the knowledge that is needed for such decisions into four categories: the characteristics of the output channels, the features of the information to be presented, the goals and characteristics of the producer, and the characteristics of the perceiver and the communicative situation.

More recent multimodal systems view the presentation more holistically, reasoning about the semantics of the content on all of the output channels to create a coordinated, high-quality presentation. This coordination goes beyond explicit textual references to the graphics (saying *Figure N* or *on the left*, for example), although that is an important component (e.g., Wahlster, 2006). The content selected for one channel may constrain the content that can be chosen on others. For example, if a figure demonstrates a procedure, then less text is needed to describe the steps; similarly, the content of a document and its typographical layout have a mutual influence (Piwek *et al.*, 2005).

A particularly active task in modality selection at the moment is the generation of multimodal referring expressions: creating a description of a certain object that uniquely identifies it in a visual context accessible to both speaker and hearer. The generated reference may include one or both of linguistic description (e.g., *the blue cube*) and multimodal behaviours (highlighting, flashing, gestures of an embodied agent), and the content selected for each channel can influence the selection of content for the other. Current approaches to this task (Kranstedt and Wachsmuth, 2005; van der Sluis, 2005; Piwek, 2007) aim to reproduce the behaviour of humans producing referring expressions in controlled contexts.

2.1.3.2 Output coordination

Output coordination is the task of ensuring that the content of a multimodal presentation is internally coordinated, both spatially and temporally. Most systems that do physical layout of presentations use some form of constraint satisfaction. The constraints may be imposed top-down, as part of the overall structure of the presentation (Bateman *et al.*, 2001; Power *et al.*, 2003a), or they may be bottom-up restrictions from the individual generators. In some systems, constraints from the presentation planner may cause revisions of decisions made in earlier modules of the planner. For example, if the selected content cannot be laid out due to space constraints, the content selection might be revised (Reiter *et al.*, 2003); the generated text can also be modified to make explicit references to the multimodal content, as described above (Piwek *et al.*, 2005).

In a presentation incorporating temporal modalities such as speech or animation, an important part of producing coherent output is ensuring that the various parts of the presentation occur at the right time with respect to one another. In practice, in most current systems, this largely amounts to making sure that other components of the presentation take place at the right time with respect to the speech output; that is, it is the speech that largely determines the temporal behaviour of the presentation.

To coordinate behaviours with synthesised speech, two possible approaches can be taken (Cassell *et al.*, 2001c); which one is chosen in a given system depends on the capabilities of the speech synthesiser that is used. With speech synthesisers that produce real-time events as the audio is produced (e.g., AT&T NaturalVoices[2]), the animation system may compile a set of event-triggered rules to govern the behaviour generation, and then create the animation reactively during the presentation. If the synthesiser cannot produce such real-time events—or if the animation cannot be prepared on the fly—the system must instead obtain an estimate of word and phoneme timings in advance, and then construct the full animation schedule prior to execution.

2.2 Corpora and generation

In recent years, the increasing availability of large textual corpora, both annotated and unannotated, has resulted in an explosive development of language-processing techniques that make direct use of the data represented in a corpus. The areas where data-driven techniques have been successful include machine translation, part-of-speech tagging, parsing, chunking, and summarisation (Manning and Schütze, 1999). In the last decade, researchers in Natural-Language Generation (NLG) have now also begun to make use of such techniques.

This section describes the state of the art in two related fields of research. In Section 2.2.1, we describe how corpora have been used as a resource for decision-making in a range of generation systems. In Section 2.2.2, we then summarise the growing amount of work that is currently being done on collecting and using multimodal corpora.

[2] `http://www.naturalvoices.att.com/`

2.2.1 Data-driven generation

In the view of natural-language generation (NLG) described by Reiter and Dale (2000), the main use of a corpus was as a guide for the developers of the system: the texts in a corpus were seen as target outputs to help in specifying the system, but there was no notion of using the texts themselves directly in the system. However, in recent years, there has been an increasing trend towards applying statistical techniques developed in other areas of computational linguistics directly to the corpus data. The proceedings of a workshop on the use of corpora in NLG (Belz and Varges, 2005) include both systems that use corpora as part of their decision-making process and those that use them to evaluate their output. In Section 2.6.2, we survey how corpora have been used for evaluation; in this section, we concentrate of the use of corpora for decision making.

One of the first generation systems to exploit corpora directly was Nitrogen (Langkilde and Knight, 1998a,b). Nitrogen works in two stages: first, it maps its semantic inputs into word lattices, and then it uses *n*-grams derived from text corpora to search through a lattice to find the best-scoring realisation. The successor system, HALogen (Langkilde-Geary, 2002), adds a fuller treatment of syntax to the lattice-generation process and makes other modifications to allow the system to scale for broader coverage and to allow an application to have finer control over the output if necessary. Kipp (2004) used a similar lattice-based overgenerate-and-filter method to select gestures to accompany speech, using individualised "gesture profiles" derived from annotated corpora of skilled public speakers; this system is discussed further in Section 2.4.1.

Other text-generation research has also made direct use of corpora in a variety of ways. Oberlander and Brew (2000) proposed a modification to the Nitrogen architecture which uses fluency criteria such as sentence length to guide the search through the word lattice. Bangalore and Rambow (2000a,b) described FERGUS, a text generator that integrates *n*-gram models with an XTAG grammar and a tree-based stochastic model derived from the Penn Treebank, and demonstrated empirically (Bangalore *et al.*, 2000) that the use of the tree-based model improves the generation accuracy of the system when measured against a corpus.

Varges and Mellish (2001) described an application of instance-based learning methods to the task of natural language generation in the context of the IGEN generation system. IGEN uses a modified chart-based realisation algorithm that integrates a ranking on the edges based on their similarity to a corpus of instances; this ensures that the system is able to find a highly-ranked candidate without needing to explore the entire search space.

The OpenCCG surface realiser (White, 2005, 2006b) uses a similar chart-based realisation algorithm; however, it ranks its edges using *n*-gram precision scores derived from a corpus of target outputs, rather than comparing the edges against the examples directly. OpenCCG incorporates a number of efficiency measures (White, 2006b) to help guide the search, including chunking of the input logical forms, pruning of the search space, and formulating the realisation process as an anytime, best-first algorithm. Isard *et al.* (2006) used the OpenCCG realiser as the basis for a corpus-based method of adding personality to generated texts.

*p*CRU generation (Belz, 2006) is a generation framework that combines a probabilistic generation methodology with a comprehensive model of the generation space, where probabilistic choice informs choices during the generation process. *p*CRU first implements a base generator using a set of generation rules and argument and relation-type hierarchies. Probabilities are associated with generation rules by training on an unlabelled corpus. The generator can be run in one of three modes. In *greedy* mode, it chooses the single highest-probability rule in every case; in *Viterbi* mode, it searches the full generation forest and chooses the overall highest-probability result; while in *greedy roulette* mode, it chooses among generation rules stochastically, using the corpus probabilities.

2.2.2 Multimodal corpora

A multimodal corpus is an annotated collection of coordinated content on communication channels such as speech, gaze, hand gesture, and body language, and is generally based on recorded human behaviour.[3] Researchers in this area have increasingly been coming together to share raw and annotated data, as well as techniques and tools for annotation and analysis. A series of workshops on multimodal corpora (Maybury and Martin, 2002; Martin *et al.*, 2004, 2006) have seen papers presented describing corpora and their applications in areas including meeting analysis, hand gestures, multimodality during conversation, and multimodal human-computer interaction.

The normal method for annotating a multimodal corpus is to annotate each of the individual communication modalities on its own layer and to make explicit or implicit links between the layers. Standard tools for this type of annotation include Anvil (Kipp, 2004), NXT (Carletta *et al.*, 2005), and ELAN (Hellweig and Van Uytvanck, 2006). The types of data that are annotated depend both on the corpus and the intended applications, and may range from low-level time-stamped motions to high-level discourse structures. A range of annotation schemes have been used; Wegener Knudsen *et al.* (2002) provide a survey.

The AMI meeting corpus (Carletta, 2006) is a typical example of a large multimodal corpus. The raw data for this corpus consists of 100 hours of recorded multi-party meetings, including full video and audio recordings of all participants, with fully-transcribed and time-stamped speech. The data has been annotated on the following levels: dialogue acts, topic segmentation, abstractive and extractive summaries, named entities, individual actions and gestures,

[3] Although Ech Chafaï *et al.* (2006) used a corpus based on Tex Avery cartoons.

person location, focus of attention, emotional content, and argumentation structure. Many of these levels are linked directly to segments of the transcript, while others—such as gestures—are marked with starting and ending times.

At the moment, multimodal corpora are built and used mainly for descriptive purposes such as analysis and summarisation. For example, the primary applications of the AMI meeting corpus include human-human communication modelling, multimedia indexing and retrieval, and meeting structure analysis and summarisation. Most papers in Martin *et al.* (2006) describe such applications; however, multimodal corpora have also been used for generating output, particularly for embodied conversational agents. Several such applications are described in Section 2.4.2.

2.3 Generating descriptions of database objects

The preceding sections gave a high-level overview of the generation process. In this section, we focus specifically on the style of generation that is implemented in the COMIC system that is used as the basis for experimentation in this thesis: generating text to describe and compare objects whose properties are drawn from a database. This is a task that has been addressed widely, from the earliest text generation systems to current multimodal dialogue systems.

One of the first systems to address this type of generation was TEXT (McKeown, 1985). This pioneering natural-language generation system created descriptions and comparisons of objects in a database of military weapons and vehicles. The generation process was based around *schemata*: generation scripts that specify both the information to include and how it should be presented. A schema also indicates the discourse goals that it can be used to satisfy. Schemata are constructed from *rhetorical predicates* representing the types of utterances in the text—attributive, analogy, or contrast, for example. Schemata contain mandatory and optional components, and some components can be repeated if necessary. In the TEXT system, four basic schemata were defined: *identification*, *attributive*, *constituency*, and *compare&contrast*.

The TEXT generation process proceeded as follows. First, based on the incoming question, the system selected the set of schemata that can be used for the answer and gathers a pool of relevant knowledge from the database. The type of information in the pool was used to select the appropriate schema in cases where more than one can fulfil the goal; for example, when a large amount of information was available about an entity, it was defined using the *identification* schema rather than the *constituency* one. The relevant knowledge was then used to fill the schema, using the semantics associated with each part of the schema to select the appropriate information. A focusing mechanism ensured that, whenever there was a choice, the proposition that tied in most closely with the preceding discourse was selected. When the schema was filled, it was passed to the tactical component to be translated into English. (2.1) shows a sample TEXT output comparing an ocean escort with a cruiser (McKeown, 1985):

(2.1) The cruiser and the ocean escort are surface ships. Ocean escorts have a DISPLACEMENT between 3400 and 4100. All ocean escorts in the ONR database have REMARKS of 0, FUEL TYPE of BKNR, FLAG of BLBL, MAST HEIGHT of 85 and PROPULSION of STMTURGRD. Ocean escorts carry between 2 and 22 torpedoes, 16 missiles, and between 1 and 2 guns. A ship is classified as an ocean escort if the characters 1 through 2 of its HULL NO are DE. Cruisers have a PROPULSION of STMTURGRD and a LENGTH between 510 and 673. All cruisers in the ONR database have REMARKS of 0 and FUEL TYPE of BKNR. Cruisers carry between 8 and 42 torpedoes, between 4 and 98 missiles, and between 1 and 4 guns. A ship is classified as a cruiser if the characters 1 through 2 of its HULL NO are CL or the characters 1 through 2 of its HULL NO are CG. The ocean escort, therefore, has a smaller LENGTH and a smaller DISPLACEMENT than the cruiser.

Many subsequent systems have addressed similar generation tasks, in a variety of domains and using a range of techniques. TAILOR (Paris, 1988), for example, generated descriptions from a database of patent abstracts, customised to the properties of the user, while ROMPER (McCoy, 1988) generated texts correcting user misconceptions about object properties in multiple domains; these systems are both discussed in more detail in Section 2.5.2. Other systems in the same general area include GOSSIP (Carcagno and Iordanskaja, 1993), which generated descriptions of user operations in a computer centre; MIGRAINE (Mittal *et al.*, 1994) and PIGLET (Binsted *et al.*, 1995), which created user-tailored health-education information; ADVISOR II (Elhadad, 1995), which provided advice on selecting undergraduate computer-science courses; and IDAS (Reiter *et al.*, 1995), which created technical documentation based on a representation of the object being documented.

One large class of systems in this area consists of those that generate intelligent descriptions of objects for educational purposes, taking into account the preceding discourse (see Section 2.5.1 for more on uses of discourse history in generation). Systems of this type include PEBA-II (Milosavljevic, 1999), which acted as an intelligent online encyclopedia, as well as three systems designed to generate descriptions of museum objects: POWER (Dale *et al.*, 1998), ILEX (Mellish *et al.*, 1998; O'Donnell *et al.*, 2000) and M-PIRO (Isard *et al.*, 2003; Androtsopoulous *et al.*, 2007).

The focus in PEBA-II and POWER (an adaptation of PEBA-II to the museum domain) was on the generation of intelligent comparisons among the objects as they are presented, in order to increase the coherence and understandability of the generated text. ILEX had a similar architecture, but added the notion of a *system agenda*: in addition to responding to user requests, ILEX also had its own didactic goals such as communicating information that the museum curators have labelled as important. M-PIRO was largely based on ILEX, but added a high-level content authoring tool and was able to generate descriptions in multiple languages;

it is currently being redeveloped into a commercial prototype.[4] The following text generated by M-PIRO (Karasimos and Isard, 2004) gives an example of the text generated by systems of this type; note that this text would normally be accompanied by an image of the exhibit being described:

(2.2) This exhibit is a stamnos. Unlike the previous vessels, which were created during the archaic period, this stamnos was created during the classical period. It shows Dionysus (centre) being garlanded by maenads in a state of ecstasy. One maenad (left) is filling a skyphos with wine, another (right) is playing a drum. This stamnos was decorated by the painter of Dinos with the red figure technique and is made of clay.

Another class of similar generation systems are "recommender" systems: these include GEA (Carenini and Moore, 2006), MATCH (Walker *et al.*, 2004), and FLIGHTS (Moore *et al.*, 2004). All of these systems generated descriptions and comparisons of options—real estate, restaurants, and airline flights, respectively—that were tailored to the user's likes and dislikes in addition to the discourse context. We discuss this type of tailoring in detail in Section 2.5.2.2.

GEA generated evaluative arguments tailored to user preferences in the domain of real estate, in the context of a graphical data-exploration environment. When generating a description of an option, the user preferences influenced the features that were included, the ordering of those features, and the use of scalar adjectives and adverbs in the text. MATCH extended GEA's approach to generate tailored spoken descriptions and comparisons of restaurants in New York City; FLIGHTS generated similar descriptions of flight options and uses the user-preference information at all levels of the generation process, from content selection to appropriate intonation. The following is a sample output from FLIGHTS (Demberg and Moore, 2006), tailored to a user who prefers to fly business class and to fly direct:

(2.3) You can fly business class on KLM, arriving at four twenty p.m., but you'd need to connect in London. There is a direct flight on BMI, arriving at four ten p.m., but it has no availability in business class.

2.4 Embodied language

In the preceding section, we described a number of systems that have addressed one of the presentation modalities that we investigate in this thesis: generating descriptions of database objects. This section addresses the other main output channel that we consider: namely, the selection of non-verbal behaviours for an embodied conversational agent. In Section 2.4.1, we first provide an overview of the use of non-verbal behaviour in human interaction. Section 2.4.2 then describes the mechanisms that have been used to generate human-like non-verbal behaviour for embodied artificial agents.

[4] http://www.ltg.ed.ac.uk/methodius/

2.4.1 Non-verbal behaviour in humans

There is no longer any question that the production of language and its accompanying non-verbal behaviour are tightly linked (e.g., Bavelas and Chovil, 2000; McNeill, 2000b; Kendon, 2004). The communicative functions of body language listed by Bickmore and Cassell (2005) include conversation initiation and termination, turn-taking and interruption, content elaboration and emphasis, and feedback and error correction; non-verbal behaviours that can achieve these functions include gaze modification, facial expressions, hand gestures, and posture shifts, among others.

While there is general agreement that language and gesture are derived from a common source, there are alternative hypotheses for how language and gesture are actually produced in practice. McNeill (2000a), for example, proposed that language and gesture are produced simultaneously as expressions of the same underlying idea (called a *growth point*), while de Ruiter (2007) proposed that content on the two channels is planned together but executed separately. Although the content expressed on the two communicative channels can overlap, de Ruiter (2007) argued that this information is never truly redundant: even when the propositional content is formally identical across the two channels, the mere fact that both channels are used is itself a communicative act.

In the remainder of this section, we discuss findings from human studies of four specific functions of non-verbal behaviour that have been well studied: deictic pointing, iconic gestures, expressions of affect, and visual correlates of prosodic emphasis.

Deictic pointing gestures generally co-occur with deictic speech such as *this* and *that*. There is evidence that, not only do deictic expressions require a pointing gesture to be resolved, but the inverse is also true: de Ruiter (2007) describes a task in which, when participants were forbidden to speak, the rate of pointing gestures also dropped significantly. Pointing is used frequently: in one study, Piwek (2007) found that nearly half of the referring acts in a corpus included a pointing gesture, and speakers tended to point more frequently when the object was not in focus or was important for the current task. The type of pointing can also vary depending on the conversational and pragmatic context: the shape of the gesture—and even the body part used to make it—have been shown to vary depending on factors including the proximity of the intended referent (Wilkins, 2003) and whether the hearer is assumed already to be aware of the location being indicated (de Ruiter, 2007). Kranstedt *et al.* (2006) introduced the concept of a *pointing cone*, which represents the focused area of a finger pointing gesture, and used this concept to define a formal semantics of multimodal referring expressions.

Iconic gestures are gestures whose form resembles their referent in some way, and that are created on the fly by a speaker and have no predefined mapping from form to meaning: they differ from gestures such as "thumbs up" in that they have no inherent meaning and therefore cannot be interpreted without reference to the accompanying speech. For example, an iconic gesture might indicate the direction or speed of a motion or the shape of a relevant landmark in an instruction-giving context. Goldin-Meadow *et al.* (1993) and McNeill (2000a) proposed that these properties mean that a speaker's use of iconic gestures provides a direct means of observing the mental representations active in their mind. On the other hand, de Ruiter (2007) argues that, despite the similarity, the transformations that take place between the mental representation and the concrete gesture are sufficiently numerous and complex that the underlying mental representation is obscured. Lascarides and Stone (2006) proposed a formal semantic representation of the relationship between speech and iconic gesture. Their analysis first produces an underspecified representation of the gesture meaning; this meaning is then resolved by determining the appropriate rhetorical relation to link the gesture and the speech, using mechanisms drawn from dynamic semantics (Asher and Lascarides, 2003).

Non-verbal behaviour—in particular, facial expressions—also has a large role to play in the perception of affect. The characteristic static facial expressions accompanying Ekman's "basic emotions" (Ekman, 1999) such as happiness and disgust have been well studied. Other affective states have been studied, as have displays that evolve over time; Cowie *et al.* (2001, pp. 58–66) gave a survey of research into the relationship between emotions and facial expressions. Krahmer and Swerts (2005) studied a range of audiovisual signals of uncertainty in adults and found that delays, filled pauses, eyebrow raises, high intonation, and "funny faces" (i.e., marked facial expressions) were all associated with less certainty on the part of the speaker. When the videos from this experiment were played back to a separate group of participants, the degree of certainty was reliably identified. Cunningham *et al.* (2005) manipulated video sequences to determine the components of the face that are most relevant to detecting facial expressions, and found that different expressions rely on different parts of the face: agreement and disagreement rely on rigid head motion, for example, while the mouth is critical to detecting happiness. Other studies have investigated non-verbal signals of deception: Ekman *et al.* (1999) and Rehm and André (2005) both found that judges can be good at detecting these signals. More generally, Buisine *et al.* (2006) investigated signals of "blended" emotions, in which signals of more than one emotion are simultaneously expressed (e.g., disappointment masked by joy): the participants in this study were often able to identify all components of the emotion, especially when other signals such as speech were included.

A range of behaviours have been observed to co-occur with the accented parts of an utterance. Ekman (1979), Cavé *et al.* (1996), and Flecha-García (2006) all noted that eyebrow motions were more frequent on accented words; Graf *et al.* (2002) and Keating *et al.* (2003) observed that head motions occurred more often with prosodic accents; while Erickson *et al.* (1998), Keating *et al.* (2003), and Dohen *et al.* (2004) all observed greater amplitude mouth movements. In a series of studies, Swerts and Krahmer have demonstrated that congruent speech prosody and visual cues are preferred to conflicting cues, that correct facial displays enhance judges' ability to perceive stress, and that the upper part of the face and the left side are the most relevant for perceiving the intended prosody (Krahmer and Swerts, 2004; Swerts and Krahmer, 2004, 2007). Thompson and Russo (2006) also found that the facial expressions of singers affected hearers' perception of the music. The information structure of an utterance can also affect non-verbal behaviour: Cassell (2000) found that gesturing behaviour tends to occur largely in the rhematic part of an utterance (i.e., the new or interesting part).

2.4.2 Non-verbal behaviour for embodied conversational agents

An Embodied Conversational Agent (ECA) is a computer interface that is represented as a human body, and that uses its face and body in a human-like way in conversation with the user (Cassell *et al.*, 2000; Pelachaud and Ruttkay, 2004). The main benefit of an ECA as a user-interface device is that it allows users to interact with a computer in the most natural possible setting: face-to-face conversation. However, to take full advantage of this benefit, the conversational agent must produce high-quality output, both verbal and non-verbal. Non-verbal behaviour has two main aspects: motions such as beat gestures and emphatic facial displays that correspond directly to the structure of the speech, and other behaviours such as emotional facial expressions that are related to the pragmatic context.

In the preceding section, we described how a number of non-verbal behaviours are used by humans in conversation. To select behaviour for an ECA, it is necessary to translate the general findings from observing humans into concrete selection strategies to be used in the implementation.There are two main implementation techniques that have been used for making this decision. In some cases, the recorded behaviours are analysed by hand and rules are created to make the selection; in others, recorded human data is used directly in the decision process. These two techniques parallel the two main ways in which corpora have been used in generation systems (Section 2.2.1). In the rest of this section, we describe systems that have made use of each of these implementation techniques.

Several implementations have based their selection of non-verbal behaviour on rules derived from studying human non-verbal behaviour. (Cassell *et al.*, 2001a) used a unified, incremental generation approach to plan the text and gesture content of a turn for the REA agent. Their implementation selected the gestures and facial expressions to accompany the text using heuristics derived from previous studies of typical North American non-verbal displays (e.g. Chovil, 1991). (Poggi and Pelachaud, 2000; de Carolis *et al.*, 2002) concentrated on producing appropriate *affective* rule-based non-verbal behaviour for the Greta agent; for example, the agent might look sad when delivering bad news to the user. Again, the rules to map from emotional states to facial displays were based on the literature on facial expression of emotion. André and colleagues have used similar techniques to implement agents that display different politeness strategies in both speech and gesture (Johnson *et al.*, 2005), and that show by their body language when they are lying (Rehm and André, 2005), while Egges *et al.* (2004) and Meerbeek *et al.* (2006) have implemented embodied agents that express different personalities through body language and interaction styles. The embodied head in the IMIX information-retrieval system (Marsi and van Rooden, 2007) used rules to generate facial displays that indicate certainty or uncertainty, based on the findings of Swerts and Krahmer outlined in the preceding section. Kranstedt *et al.* (2006) used findings from their study of human deictic pointing gestures (discussed in the preceding section) to help select the gesturing behaviour of the Max embodied agent.

All of the agents described above based their selection of non-verbal behaviours on rules derived from large-scale studies of human behaviours, but none used the recorded human behaviour directly. This means that they tend to produce "average" behaviour from a range of speakers, and cannot easily be adapted to produce specific personality and stylistic effects. To address this, other systems have instead made direct use of the recorded behaviours of a single speaker. In some cases, motion-capture techniques are used, while in others a recording is annotated by hand and the annotated behaviours are then used. Motion capture is able to produce more naturalistic output than a rule-based system, and can also easily model a single individual; however, capturing the motion requires specialised hardware, and the agent must be implemented so that it can exactly reproduce the human motions. Annotating a video corpus can be less technically demanding than capturing and directly re-using real motions, especially when the corpus and the number of features under consideration are small.

Motion capture has been used in several systems. Stone *et al.* (2004), for example, recorded an actor performing scripted output in the domain of the target system—an instructor character for a video game. They then split the recordings into coherent phrases and annotated each phrase with domain-specific semantic and pragmatic information such as the move that the player attempted and what the result was. To generate behaviours for the agent, they used a template grammar to select phrases and combined them into a performance specification, which was then played back on the agent. Cunningham *et al.* (2004) and Shimodaira *et al.* (2005) used similar motion-capture techniques to create the appearance and facial displays for talking heads, and Hofer (2006) proposes using a similar techniques. The motions of the latter two systems are derived directly from the features of the synthesised speech, using mechanisms similar to those used in unit-selection speech synthesis (Hunt and Black, 1996).

Using an annotated corpus has also been used to select behaviour for several agents. For example, Cassell *et al.* (2001b) used this technique to choose posture shifts for the REA agent. Speakers were first recorded giving "pseudo-monologues" describing each room of their home and giving directions between two well-known landmarks. The recordings were transcribed and annotated for discourse segment boundaries, turn boundaries, and posture shifts. Probabilities derived from this corpus were then used to select the posture shifts to be included in the output. More recently, Kipp (2004) used a similar technique to generate agent gestures, as follows. He first annotated the gesturing behaviour of a range of skilled public speakers, and then derived individual “gesture profiles” representing the handedness, timing, and function of gestures through estimated conditional probabilities. To generate speech and gestures, he first used heuristic rules to specify possible gesture locations on semantically-annotated text, and then used the gesture profile to add and filter gestures to create a final schedule.

2.5 Sources of variation

The implementations in this thesis all deal with incorporating variation into the output of a generation system, using different implementation techniques and a range of knowledge sources. In this section, we survey the various ways in which dynamic choice has been used in generation systems. We first concentrate on three main sources that have been used to drive rule-based variation in output: the discourse history (Section 2.5.1), a model of various aspects of the system user (Section 2.5.2), and rhetorical and stylistic goals (Section 2.5.3). Finally, in Section 2.5.4, we describe systems that have used random choice to introduce variation into their output.

2.5.1 Discourse history

The ability to modify the content of a generated presentation to take into account previously-generated output is often cited as one of the main advantages of choosing generation over canned text. Keeping track of the discourse history allows a generation system both to avoid repeating information that has already been communicated and to choose methods of presenting new information that best relate it to the preceding content. This is a source of generation knowledge that dates back to the earliest generation systems—for example, McKeown’s TEXT system (McKeown, 1985) (see Section 2.3) maintained a model of the preceding discourse when generating its descriptions, and used constraints on the shift of focus of attention to select the information that fits in best with the preceding discourse. It is rare to find a generation system that does not take into account the discourse history in some way, unless the goal is to generate one-off messages (e.g., when generating the initial sentence of newswire stories (Varges and Mellish, 2001)).

Discourse-history information is often used when choosing referring expressions, either textual or multimodal. In the classic incremental algorithm for referring-expression generation (Dale and Reiter, 1995), which adds attributes to a description in a fixed order until all distractors have been ruled out, the only relevant factors are the attributes of the target object and the distractors. Krahmer and Theune (2002) extended this algorithm by adding salience weights to all of the objects in the domain and modifying the algorithm so that it stops when the target object is the most salient object with the proposed description. Salience weights are computed based on the function of the object in the preceding discourse (e.g., focus objects have higher salience than other objects in a sentence), and decrease gradually over time. In a human evaluation, participants generally preferred the expressions generated by this extended algorithm to those generated by the basic incremental algorithm.

The above work describes using the preceding discourse to decide how to refer to a single entity. The discourse history can also be used to help structure entire texts. Kibble (1999) described how constraints for centering theory (Grosz *et al.*, 1995), which defines rules on coherence and focus transition across sentences, can be used to help make text-planning decisions in NLG. Karamanis (2004) implemented a number of centering-based metrics of text ordering and compared them empirically; he found that the simplest metric provided a very strong baseline, but that modifying it to include constraints on *PageFocus*—the main entity of the whole document—produced better results.

A major emphasis of the generation in the museum-guide-style systems discussed in Section 2.3 (Milosavljevic, 1999; O'Donnell *et al.*, 2000; Androtsopoulous *et al.*, 2007) was to present new objects by making explicit comparisons and contrasts with objects and concepts from the discourse history. This involves choosing appropriate referring expressions, selecting appropriate material for the comparison, and producing coherently structured texts. Karasimos and Isard (2004) found that incorporating comparisons into the output increased learning performance on the M-PIRO system. McCoy (1988) used similar techniques to describe objects in the context of the preceding discourse.

Moore and Paris (1993) pointed out that, in order to participate in a dialogue, a system must be able to reason about its own previous utterances, as well as those of the user. This allows the system to respond to (possibly vaguely articulated) follow-up questions from the user, to clarify misunderstandings, and to elaborate on prior explanations. For the system to be able to do this, more than just the surface content of turns must be recorded in the history: the system must also have access to the full propositional content and the reason that the content was selected in order to be able to respond intelligently and appropriately.

2.5.2 User model

Like the discourse history, a model of the user is also very often used as a resource for decision making in a generation system. Kobsa and Wahlster (1989) and, later, Zukerman and Litman (2001) give overviews of the various ways in which user models have been used to help make generation decisions. Two main types of information about a user have been employed in generation systems: the user's expertise or interests, and their likes and dislikes. In this section, we present several systems that have made use of each of these information types to help make generation decisions. Note that many systems where the primary emphasis is on modifying the style of the output or on meeting specific rhetorical goals also make use of information about the target user in addition to other knowledge sources. Several systems of this type are described in Section 2.5.3; in this section, we concentrate on systems where the user model is the primary knowledge source for content selection and other factors.

2.5.2.1 Expertise and interests

A number of systems have taken into account information about the user's knowledge and interests to create tailored output. The TAILOR system (Paris, 1988), for example, generated descriptions of objects from a database, tailoring the description to the user's level of expertise. The expertise level affected the output in two ways: novice users were presented with more detail than were expert users, and the system also used different explanation strategies for the two types of users. For a novice, an explanation was arranged around tracing the process that allows an object to perform its function, while for an expert the system instead generated a description based on the components of the object; these strategies were derived from studying sample human-written descriptions designed for both types of readers. The user model stored the basic concepts and specific artifacts understood by the user. For example, a user might have good knowledge of how a motor works but little knowledge about elevators, so a description of the latter should take into account their knowledge of the former.

McCoy's (1988) ROMPER system generated texts in multiple domains to address two types of object-related misconceptions: misclassification of an object, and assignment of attribute values to an object that differed from those in the system's domain model. The user model explicitly represented the user's beliefs regarding object attributes (e.g., a belief that a whale is a fish) and selected discourse strategies designed both to correct the error and, where possible, to give support by explaining reasons that the user might have held the incorrect belief. The user model was "highlighted" as the discourse proceeds to indicate the concepts that had been discussed so that correct inferences could be drawn.

Other systems that represented the user's capabilities and interests include those of McKeown *et al.* (1993), who endeavoured to use words in the user's vocabulary in explanations; Zukerman and McConachy (1993), who generated concept descriptions that address anticipated erroneous inferences and omit easily inferred information, tailored to the user's capabilities; Milosavljevic and Oberlander (1998), who tailored the content of descriptions to the user's interests and skill level; Zinn *et al.* (2005), who implemented a tutoring system that dynamically modelled the student's understanding of the problem domain; and Carroll *et al.* (1999) and Williams and Reiter (2005), both of who generated texts designed for readers with low literacy skills.

A number of healthcare-related generation systems use information from the patient's medical record both to determine the most relevant information and to decide how to present it: Åhlfeldt *et al.* (2006, ch.5) provide a recent survey. The PIGLIT system (Binsted *et al.*, 1995) provided hypertext explanation of a patient's medical record, tailored using data that could be drawn directly from the record such as symptoms and length of illness; Cawsey *et al.* (2000) later modified the system to generate texts for patients with cancer. MIGRAINE (Mittal *et al.*, 1994), HealthDoc (DiMarco *et al.*, 1995) and STOP (Reiter *et al.*, 2003) used similar methods to generate patient-education materials about other topics. Systems such as PERSIVAL (McKeown *et al.*, 2001), CLEF (Hallett and Scott, 2005), and TAS (Elhadad *et al.*, 2005) that generated texts designed to be used by medical staff also used similar techniques to determine relevance and presentation strategies based on patient characteristics and user queries.

2.5.2.2 Likes and dislikes

One of the earliest generation systems to make use of a model of user preferences was Rich's (1979) Grundy system, which provided book recommendations tailored to a user's preferences. The system asked a series of questions to obtain an initial user model, and then updated this model based on the user's responses to suggestions during the course of the (typed) interaction. The user models were based on stereotypes, which were represented as typical combinations of attributes that would be activated by triggers in the conversation.

Since that original work, many other generation systems have made use of explicit models of the user's likes and dislikes to help specify the output. For example, The ADVISOR II system (Elhadad, 1995), which generated advice on choosing computer-science undergraduate courses, stored the student model as a set of slots, including factors such as previous courses taken, interests, and skills. It then used this information along with knowledge of the available courses and of argumentation strategies to generate tailored advice-giving paragraphs.

Several recent systems have represented the user's preferences as an Additive Multiattribute Value Function; these models are based on the notion that, if anything is valued, it is valued for multiple reasons, where the relative importance of the reasons may vary for different users. An AMVF-based user model is able to sort a number of options into a preference order, and to indicate which of the features of each option contributed the most to its positive or negative rating; both of these types of information are useful for tailoring generated output to a user. AMVF-based user models have been used in a variety of systems and domains, including the "recommender" systems listed in Section 2.3 (Carenini and Moore, 2006; Walker *et al.*, 2004; Moore *et al.*, 2004). All of these systems used the user-model information both to choose the content to present to the user and to structure it appropriately. More details of AMVFs are given in Section 4.1.

User-preference representations other than AMVFs have also been proposed. Carberry *et al.* (1999), for example, represented similar information for use in collaborative consultation dialogues, using a more ad-hoc representation; they also proposed a method for updating preferences during the course of an interaction. CoGenTex's commercially-deployed *Recommender*[5] system also represented the user's preferences in a similar manner.

2.5.3 Rhetorical and stylistic goals

The preceding two sections described how information from two external knowledge sources can be used to influence the output produced by a generation system. We now examine the effect on the output of the goals of the generation system itself: for example, the rhetorical goals and specifications of the desired style to use, or the personality to project. This type of goal interacts with the user-model information presented in the preceding section: the style of the text may vary based on characteristics of the user, as in the texts generated for laypeople, funders, and computational linguists by DiMarco and Foster (1997); user characteristics can also influence factors such as the graphical techniques to be used and the modality-selection choices (Rist, 2005). In this section, we focus on implementations where stylistic and pragmatic variation is the primary goal, as opposed to user tailoring.

The foundational work in this area is Hovy's PAULINE system (Hovy, 1988). The goal of PAULINE was to demonstrate that taking into account pragmatic information enables a range of generation decisions to be made that are otherwise not decidable. The system generated texts describing a series of events, using pragmatic information including the speaker's goals for the hearer (e.g., changing the hearer's knowledge or affective state), goals regarding the speaker-hearer relationship (e.g., modifying the status or distance of the participants), the desired conversational setting (formality, time pressure), the properties of the speaker and the hearer (e.g, knowledge of and opinion towards the topic), and the relationship between the participants (friends, social status, like or dislike).

[5] `http://www.cogentex.com/solutions/recommender/index.shtml`

PAULINE used all of these constraints as a resource for decision-making at all levels of the generation process, from content selection through to lexical choice. Its implementation combined planning and realisation into a single interleaved process. For a given input, the planner assembled an initial, incomplete specification of the output and sent it to the realiser; when the realiser needed more specific information at a particular choice point, it in turn called the planner again to retrieve a solution. The following examples (Hovy, 1988, pp. 7–8) show excerpts of two descriptions generated by PAULINE of the same sequence of events. (2.4) is intended to represent a Kennedy supporter speaking at a medium level of formality, while (2.5) corresponds to a Carter supporter giving a formal speech.

(2.4) Kennedy narrowed Carter's lead by getting all of 21850 votes in the primary in Michigan. In a similar case, Carter decreased Udall's lead in a primary in 1976, and he easily trounced Udall to be nominated by 2600 delegates. I am glad that presently Kennedy is closer to getting the nomination than before.

(2.5) I am pleased to inform you that Carter has improved his chances of winning the nomination. At the present time, Carter has many more delegates than he had in the past; also, Carter has many more than Kennedy does.

The factors addressed by Hovy have also been used in other generation systems to drive variation. For example, several systems have generated output to project specific personalities. The goal of the CRaG system (Isard *et al.*, 2006) is to generate dialogue contributions that project specific personality traits, varying features including lexical choice and the degree of alignment between the two dialogue participants. Walker *et al.* (1997a) based choices on the semantic content, syntactic form, and acoustical realisation of spoken utterances on rules that produced different personalities and interpersonal situations. Many embodied-agent systems also aimed to project personality and other affective factors, either by basing their decisions on the behaviours of specific individuals or by following rules derived from the study of characteristic human behaviour; examples of such systems are discussed in detail in Section 2.4.2.

In addition to rhetorical goals, notions of style and genre have also been used to make generation decisions. The ICONOCLAST system (Power *et al.*, 2003b) generated patient-information leaflets: material inserted into a medicine packet which explain features such as ingredients, side-effects, and dosage instructions. The document author defined the high-level content of the leaflet using a *WYSIWYM* ("what you see is what you mean") editing style. The author could also control a range of stylistic parameters on sliding scales, including paragraph and sentence length, the frequency of connectives, passives, pronouns, semicolons, and commas, the technical level, and the use of vertical lists. These parameters provided the system with soft constraints, which interacted with hard constraints on syntax and rhetorical structure to guide the generation of texts from the authored content, using a generate-and-test approach.

Bateman and Henschel (2007) proposed a notion of genre for multimodal documents, characterised by collections of features at several levels of description, including the rhetorical structure and the page layout. In fact, the relationship between the content at these two levels defines a genre. Since the layout of a document is often distinct from its sequential order, the process of mapping between these levels involves selectively breaking apart rhetorical relations based on *break conditions* that draw on a range of sources including the semantic content and the structural properties of the rhetorical relation.

2.5.4 Stochastic choice

Often, even in a rule-based generation system, developers incorporate an element of random choice into the output-generation process; this is generally seen as a self-evidently positive feature. For example, when describing the M-PIRO museum-exhibit description system discussed in Section 2.3, Isard *et al.* (2003) state that the ability to create multiple expressions corresponding to a database fact "lets us vary the way in which the system expresses the same fact." Similarly, when the system described by van Deemter *et al.* (2005) had more than one option for realising a sentence, it chose one at random; this implementation was chosen "to maximise the *variety* of sentences produced by the system" (emphasis original).

Stochastic variation can also be implemented in a data-driven generation system. As described in Section 2.2.1, such systems normally use statistical techniques to choose the highest-ranked available option for realising a given input. There are two ways that stochastic variation can be introduced into this process: either the weights are modified, or else the selection process is adapted to select something other than the single top-scoring realisation. Stone *et al.* (2004) used the former technique when selecting behaviours for the embodied agent mentioned in Section 2.4.2: the scores were perturbed slightly to choose from among low-cost (i.e., highly-ranked) utterances. Belz (2006) used the latter technique in the greedy roulette version of *p*CRU described in Section 2.2.1, which selected among generation rules using probabilities drawn from a corpus rather than choosing the most probable rule in every case.

2.6 Evaluating generation systems

Evaluating the success of a generation system is known to be a difficult task: as pointed out by Mellish and Dale (1998), the issues include defining the input and output, choosing what to measure, selecting a control or baseline for comparison, obtaining adequate training or test data, and dealing with disagreement of human judges. All of these problems are more serious than the corresponding problems in evaluating, for example, a natural-language understanding system: generation is a more open-ended task, so the criteria for success are therefore more difficult to define.

In the current state of the art (cf. Belz and Reiter, 2005), there are two main strategies for evaluating the output of a generation system. On the one hand, the quality of the system output may be assessed by measuring the behaviour or preferences of humans in response to that output. On the other hand, an evaluation may also be carried out automatically, using a collection of "gold standard" target outputs. In the following sections, we outline the main techniques, advantages and disadvantages of these two evaluation strategies and describe a number of systems that have been evaluated using each. At the end, in Section 2.6.3, we discuss an aspect of evaluation that is the current subject of lively debate in the natural language generation community: the feasibility of a common evaluation framework for NLG.

2.6.1 Human evaluation

Most generation systems are designed to achieve a particular communicative goal, or are embedded in larger systems that have such a goal. Depending on the system, these goals may include data presentation (Roth and Mattis, 1990), persuasion (Carenini and Moore, 2006), education (Zinn *et al.*, 2005), social engagement (Bickmore and Cassell, 2005), or simply amusement (Ritchie, 2005), among others. The most complete demonstration of the success of a generation system or technique is a task-based, comparative study: that is, demonstrating that the system achieves its goals significantly better when the advanced generation components are enabled than when they are not.

Task-based studies of this sort have been used to evaluate a range of generation systems. Carenini and Moore (2006), for example, demonstrated that tailoring the content of evaluative arguments to the likes and dislikes of a user significantly affected that user's likelihood of selecting an option that was presented. Di Eugenio *et al.* (2005) showed that users learned more from an intelligent tutoring system when it made use of aggregation in its output; Karasimos and Isard (2004) found similar results with a system that generated textual descriptions of museum artifacts. The doctors studied by Elhadad *et al.* (2005) were able to find information more quickly in a summary of clinical studies when the summary was tailored to the specific patient characteristics; they also reported higher levels of satisfaction.

Some task-based evaluations have failed to find any overall impact of the generation techniques. Cox *et al.* (1999) did not find any significant difference in learning outcomes between system versions that did and did not use dynamic, adaptive generation techniques, although there was some evidence that participants using the dynamic system reached the same learning level quicker. In a large-scale clinical trial (over 2000 smokers), Reiter *et al.* (2003) found that personalised "stop-smoking" letters did not have any greater effect on recipients than the non-tailored control letters.

A full task-based evaluation of this sort can be logistically difficult to run in practice, especially during development. The requirements include a fully-working system that can successfully be used by naïve users, a baseline version of that system for comparison, and a sufficiently large subject pool who can realistically be asked to perform the relevant task. An alternative human evaluation technique is to ask judges directly to assess the quality of the generated output. If the system is interactive, this can be implemented by asking for judgements in the middle of a session; judgements can also be gathered by a questionnaire after the fact. Often, a questionnaire is used in conjunction with a task-based evaluation: in fact, all of the task-based studies cited above used a questionnaire in addition to the task-based assessment.

Having humans directly judge output quality has fewer technical requirements than a task-based evaluation. For example, it can be carried out using pre-generated outputs from a partially-working system. Also, in the case of a dialogue system, it can alleviate issues to do with unreliable or slow input processing which might otherwise overwhelm the effect of the output-generation techniques of interest (e.g., White *et al.*, 2005). Another advantage is that the participants do not necessarily have to be experts in the target task; if an "overhearer" paradigm is chosen (cf. Clark and Schaefer, 1992), the participants can make judgements on the quality of a simulated interaction between the system and another user.

Using this technique, Yeh and Mellish (1997) compared techniques for generating anaphora in Chinese by asking native speakers to judge generated texts and found that the rules that they used were generally successful. Binsted *et al.* (1997) measured the quality of generated jokes by asking children to compare them to both human-written jokes and non-joke texts; they found that the generated texts were recognised as jokes, and that the highest-quality generated jokes were indistinguishable from human-written jokes. Hartley *et al.* (2000) evaluated the acceptability and grammaticality of generated multilingual technical documentation; the judges generally found the output to be of comparable quality to human-written technical texts. Walker *et al.* (2002) found that the output of a trainable sentence planner was judged to be better than the output of several hand-crafted systems when participants were asked to make Likert-scale judgements while reading simulated interactions between a user and the system; in a similar study (Walker *et al.*, 2004), participants preferred restaurant recommendations that were tailored to their own user preferences over those that were tailored to the preferences of some other user. Belz and Reiter (2006) asked judges, both experts and novices, to rate the weather forecasts created by a range of NLG systems, as well as hand-written output; in some cases, those judges preferred the generated output over the corpus texts.

The results of this sort of evaluation are less conclusive than those of a task-based evaluation: while there is often a correlation, users' subjective preferences between interfaces do not always agree with task performance (Nielsen and Levy, 1994; Oviatt, 1999); also, overhearer-style evaluations generally measure perception rather than behaviour (Whittaker and Walker, 2005). However, demonstrating perception is in some ways a necessary prerequisite to task-based evaluation: if participants do not notice any difference between output generated in different ways, it is unlikely to make any great difference to their task performance.

Other evaluation techniques have been used that involve human judges more indirectly. For example, Sripada *et al.* (2005) analysed the post-edits made by human experts on the output of a weather-forecast generation system. One of the participants in a study of summaries of dive-computer data (Sripada and Gao, 2007) also gave unprompted suggestions for rewriting the generated text. The output of a system designed to generate texts for readers with low literacy was evaluated by having participants read the generated texts aloud (Williams and Reiter, 2005); participants made 9% fewer errors when reading the tailored texts. When it is applicable, this sort of indirect evaluation can yield useful results; however, not all systems are amenable to being evaluated in this way.

Another possibility is to use direct physiological measures of the participants to assess their responses to generated output. For example, Prendinger *et al.* (2005a) used eye-tracking to compare interfaces with and without embodied agents; Prendinger *et al.* (2005b) measured stress levels through galvanic skin response; while Bailenson *et al.* (2004) used interpersonal distance in a virtual environment to evaluate users' responses to an embodied agent. This type of evaluation is most common for interfaces involving embodied agents.

2.6.2 Automatic evaluation

The most complete evaluation of a generation system is one that measures human performance or satisfaction with the generated output, as described in the preceding section. However, this type of evaluation is not always feasible or appropriate. For example, stochastic generation systems must be evaluated frequently during the course of development; also, recruiting sufficient human participants to carry out a study can be time-consuming and expensive, even for non-task-based studies where the quality of the output is judged directly.

It is also possible in such cases to use automatic metrics to assess the quality of generated output. The goal for this sort of evaluation is to find some metrics that can easily be computed automatically and that can also be shown to correlate with human judgements of quality. Such metrics have been introduced in other fields, including PARADISE (Walker *et al.*, 1997b) for spoken dialogue systems, BLEU (Papineni *et al.*, 2002) for machine translation, and ROUGE (Lin, 2004) for summarisation. No single automated evaluation technique has been found that can serve this purpose for generation systems as a whole—although, as discussed below, the feasibility of such a shared evaluation metric is a current topic of discussion. A number of systems have been successfully evaluated automatically using more specific methods tailored to the needs of a particular system.

Bangalore *et al.* (2000) used this type of evaluation in the development of their FERGUS realisation module. They describe several metrics that they used for baseline quantitative assessment of their system during development: all of the metrics compared their outputs directly to the corpus on which it was trained, using either the surface strings or the syntactic trees. They found that the metrics correlated well with human judgements of quality on word ordering.

White (2004) evaluated the accuracy and speed of the OpenCCG surface realiser through cross-validation on a corpus of target texts, with two scoring functions: the number of times the top realisation candidate exactly matched the target, and the *n*-gram precision score of the top candidate (using a method based on BLEU). He compared OpenCCG under a number of configurations with two baseline systems and a "topline" system that used only *n*-grams from the target sentence in the search. On both test grammars, OpenCCG out-performed both baselines and did nearly as well as the topline. This cross-validation setup also permitted the individual contribution of each of the efficiency methods that was implemented to be assessed; all of the methods were found to make a significant contribution to the speed of the system. Cross-validation precision and recall have also been used by Marciniak and Strube (2005) and Wan *et al.* (2005) to evaluate the output of their text-generation systems.

Karamanis and Mellish (2005) presented corpus-based methods for evaluating information-ordering systems, distinguishing between two types of evaluation metrics: *distance-based* and *search-based*. Distance-based metrics are those that estimate how close a proposed ordering is to the actual ordering represented in the corpus: the precision and recall measures above are classic examples. Search-based metrics instead compare selection strategies by measuring how highly each scores the gold-standard examples from the corpus: that is, they penalise a system proportionally to its failure to promote the gold standard as the best option. Search-based metrics are only suitable to use on systems that operate nondeterministically, and can be used to compare a range of selection strategies as was done by Karamanis (2004).

While most automated evaluations of generated output have been based on similarity to a corpus of known-good examples, some metrics that score inherent properties of the output itself have been used. Mellish *et al.* (1998), for example, performed a pilot experiment in which the output of three stochastic text-structuring methods was compared by counting the number of facts conveyed and the count of elaboration and non-elaboration relations used. Walker (2005) proposed an alternative mechanism for making use of corpus data in an evaluation: if each of the corpus examples is associated with some reward function (e.g., the subjective user evaluation or a measure of task success), then machine-learning techniques such as reinforcement learning or boosting can be used to train the output planner. The key to these methods is that they provide a mechanism for associating scalar evaluation metrics with the different possible output choices, inducing a ranking. Evaluation is then a matter, not of comparing possible outputs directly against specific corpus examples, but rather of using the learned evaluation function to give a score to the different possibilities.

It is crucial that any automated evaluation metric of output quality be grounded in the results of a human evaluation. Reiter and Sripada (2002), for example, found that the data in the corpus they used had a great deal of variability, and the sentences in it were actually not scored particularly highly by all of the human judges, meaning that those texts should not necessarily be considered a gold standard for the generation system. Another issue, especially in a generation system that aims to incorporate variation, is that output that differs in any way from the specific examples in the corpus is, by definition, penalised by any corpus-similarity metric. These metrics tend to favour strategies that choose "average" options; however, several studies have found that human judges tend to prefer output that incorporates such variation (Stent *et al.*, 2005; Belz and Reiter, 2006).

2.6.3 Shared-task evaluation for NLG

In a shared-task evaluation, different approaches to a well-defined problem are compared based on their performance on the same task. Recently, proposals have been made that some such evaluation framework should be created for natural-language generation. This is currently a topic of lively discussion within the community (e.g., Belz and Dale, 2006; White and Dale, 2007): even among those who believe that such an evaluation would be desirable, there is no consensus on what the task should be or what evaluation metrics should be used, while there are also those who question the utility of any effort at all in this direction.

Reiter and Belz (2006) strongly advocate moving in this direction, and present an initial concrete proposal for a shared evaluation framework. The main benefit that they see is that such a framework would allow different NLG evaluation techniques to be compared by correlating the results for different systems that are entered. They propose to surmount the problem that few generation systems have comparable functionality by supplying sets of raw data, intermediate syntactic and semantic representations, and corresponding human-written texts in three domains: weather forecasts, statistical summaries, and nursing reports. Participants would then submit systems based on this data, and their outputs would be evaluated using a range of the techniques described in the preceding sections: automated comparison against the corpus, direct human judgements of quality, and task-based evaluations.

On the other hand, Scott and Moore (2006) question the utility of such an evaluation, giving a range of "reasons to be cautious" which fall into two main categories: questioning the appropriateness of shared tasks for NLG, and re-examining the results of shared-task evaluations in other fields. They point out that NLG is a heterogeneous field with a wide range of applications and techniques, meaning that defining standard inputs, outputs, and success criteria is at least difficult, and potentially impossible; they also fear that standardising in this way can stifle scientific progress. As well, they criticise standard-task evaluations that have been used in other fields, arguing that metrics such as BLEU are now being criticised (Callison-Burch *et al.*, 2006), that the promised modularity that was supposed to come from these evaluations never materialised, and that the reason these other evaluations became popular was the large amount of government funding (which does not appear to be available in the NLG case). They agree that NLG evaluation methods need to be examined and that there is a need for a common language for comparing systems; however, they argue that the way forward is through theory-neutral architecture proposals such as RAGS (Section 2.1.2), and that the NLG evaluation community should concentrate on improving methods for human evaluation studies.

Both of the papers summarised above were presented at a special session on comparative evaluation (Belz and Dale, 2006) at the 2006 Natural Language Generation conference. A recent follow-up workshop on the topic (White and Dale, 2007) contains position papers expressing both viewpoints. This remains a hot topic of discussion in the NLG community.

2.7 Summary

This chapter has presented a range of existing systems and techniques in each of the research areas relevant to the work described in this thesis. In this section, we briefly describe how the thesis work fits into each of these areas.

In terms of the generation process outlined in Section 2.1, all of the experiments in this thesis are based around the output-generation components of the COMIC multimodal dialogue system. We make use of all four of the input components outlined in Section 2.1.1: the communicative goal is given by the dialogue manager, while the information about the objects to describe is stored in a database. In the baseline output-generation process, the dialogue history and user-preference model are not used, but the enhanced versions of the output-planning process also use these input sources.

COMIC uses a standard pipeline-based process to create its multimodal output; this process is described in detail in Section 3.2. To create the multimodal components of its output (behaviours of the talking head and gestures with an on-screen pointer), it first creates the spoken content of a turn and then uses the contextually-annotated text plan to select supporting multimodal content. As described in Section 3.2.4, modality selection is essentially hard-wired into the rules that choose multimodal content based on the text plan. Output coordination is implemented by first preparing the spoken output and then using the word and phoneme timings to create schedules for the other modalities, as described in Appendix B.

We make use of corpora in several of the ways outlined in Section 2.2. The basic COMIC output-generation uses the OpenCCG surface realiser, which—as described in that section—uses n-gram models to guide its chart-based search for a good realisation. In Chapter 5, we modify this corpus-based process to add variation into the generation process, and also use corpus-based methods to evaluate the generated output automatically. In Chapter 6, we describe the recording and annotation of a multimodal corpus of facial displays in the COMIC domain, and the experiments in Chapters 7–9 make use of this corpus data both in the generation process and for automated evaluation.

The COMIC output-generation process follows in the style of the educational and recommender systems described in Section 2.3: the information about the objects that it describes is also stored in an ontology, and the texts that it generates strongly resemble those created the systems in both of these groups. The process of generating output in COMIC is described in Section 3.2 and is used in all of the experiments in Chapters 4–5 and 7–9, as well as to create the recording scripts for the corpus described in Chapter 6.

An embodied talking head is used in all of the experiments in Chapters 7–9. This agent displays two of the non-verbal behaviour types described in Section 2.4.1: affective behaviour (based on the user-preference evaluation), along with facial displays associated with the prosody of the sentence. We base the motions of the talking head on a single-speaker corpus of facial displays in the domain of COMIC; the details of this corpus are given in Chapter 6. To select facial displays, we use both of the implementation strategies described in Section 2.4.2: using rules derived from the corpus data, and using the corpus data itself to make the selections. In Chapters 8 and 9, we compare these two strategies.

We make use of all of the variation sources described in Section 2.5 except for rhetorical and stylistic goals (Section 2.5.3). We use the discourse history in two implementations: in Section 4.3, we implement rule-based dialogue-history tailoring in the text-generation process, while in Chapter 5 we modify the generation process to avoid repeating content from the previous turn. A model of the user's preferences is used in several of the evaluations: in Section 4.4, we add user-model tailoring to the text-generation process, while in all of the talking-head experiments in Chapters 7–9 we use the user-model evaluation as one of the primary factors in the selection of facial displays. Finally, in both Chapter 5 and Chapters 8–9 we experiment with using stochastic choice to add variety to the output.

To evaluate the output generated by all of the modified versions of the COMIC system, we make use of several of the evaluation techniques described in Section 2.6. In Chapters 4 and 9, we use "overhearer"-style evaluations in which participants judge the quality of system outputs in the context of a simulated interaction; in Chapters 5, 7, and 8 we ask participants directly to make judgements on outputs presented in isolation; while in Chapters 5 and 8 we compute a variety of automated metrics on the generated output.

Chapter 3

The COMIC multimodal dialogue system[*]

> [I]n practice, even fiddling with one parameter [of a generation system] may well consume a PhD, and a large part of life thereafter.
>
> Dale (1993)

ALL OF THE implementations and evaluations in this thesis are based on modifications to the output-generation components of the COMIC multimodal dialogue system; this chapter describes the COMIC system in detail. In Section 3.1, we first survey the architecture and components of the full COMIC system. In Section 3.2, we then give a more detailed description of how dynamic tile-design descriptions are generated by COMIC, as this is the process on which all of the subsequent experiments in this thesis are based.

The development of COMIC was a collaborative effort among all of the partners in the project. Specific components where the author was largely responsible for design and implementation include the fission module (Section 3.1.5.1) and the majority of the output-planning process outlined in Section 3.2, along with the ViSoft emulator (Section 3.1.2.1), the user-model manager (Section 3.1.4.4), the speech-synthesis module (Section 3.1.5.3), and the Greta- and RUTH-based talking-head modules (Section 3.1.5.4).

[*]Parts of this chapter are adapted from Foster and White (2004); Foster *et al.* (2005).

Figure 3.1: Components of the COMIC demonstrator (tile-browsing phase)

(a) Talking head

(b) Design application

Figure 3.2: Sample COMIC input and output

User	Tell me about this design *[circle Alt Mettlach]*
COMIC	*[Nod]*
	Okay!
	[Change screen so selected design is shown in room]
	[Look at screen]
	THIS DESIGN is CLASSIC. *[circle tiles]*
	It uses tiles from the ALT METTLACH collection by VILLEROY AND BOCH. *[point at manufacturer name]*
	As you can see, the colours are DARK RED and OFF WHITE. *[point at tiles]*

3.1 Overview of COMIC

COMIC ("COnversational Multimodal Interaction with Computers")[1] was an EU IST Fifth Framework project combining fundamental research on human-human interaction with advanced technology development for multimodal conversational systems. The project ran from March 2002 through February 2005.

The multimodal dialogue system built during the course of the project added a dialogue interface to a CAD-like application used in bathroom sales situations to help clients redesign their rooms. The input channels includes speech recognition, along with handwriting and pen gestures provided either on a tablet display or with a mouse; the output combines synthesised speech, non-verbal behaviour of a talking head, deictic gestures using an on-screen pointer, and direct control of the underlying application. To run the full COMIC system requires at least two computers, one for the talking head and one for the design application; it can also run distributed across more computers if necessary.

There are four main phases when interacting with COMIC. First, the user describes the blueprint of their own bathroom, using a combination of speech input, pen-gesture recognition and handwriting recognition. Next, the user chooses a layout for the sanitary ware in the room. After that, the user is able to browse through a range of tiling options for the bathroom. Finally, the user is given a three-dimensional virtual tour of the finished bathroom.

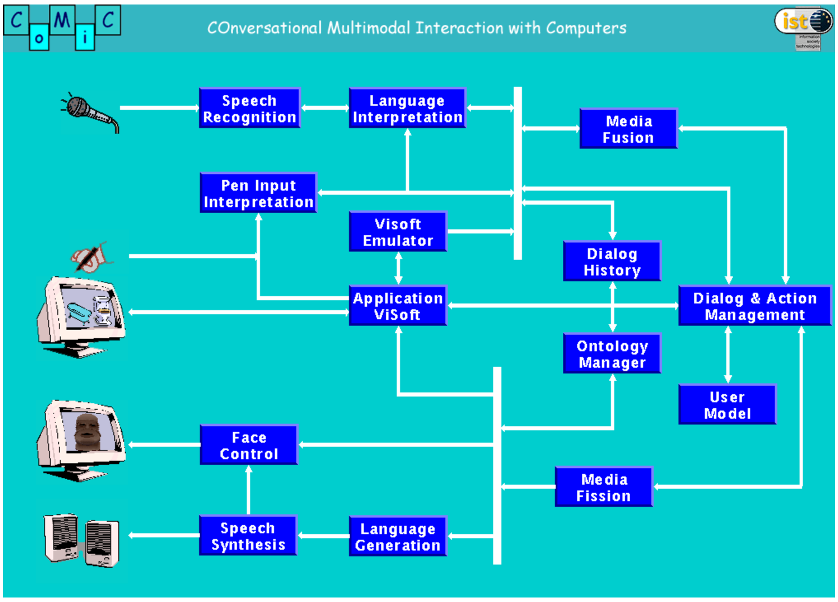
Figure 3.1 shows examples of the talking head and the design application, in tile-browsing mode, while Figure 3.2 shows a sample user request and system response from the tile-browsing phase. The sample output includes content in all of the modalities; the small capitals represent pitch accents in the synthesised speech, while the indicated circling and pointing gestures are produced using the on-screen pointer and are synchronised with the text that they appear next to. Appendix A contains a transcript of a complete interaction with the COMIC system, including screenshots of all phases, and provides a link to a video of the interaction.

3.1.1 Architecture and inter-module communication

Figure 3.3 shows the overall architecture of the COMIC demonstrator. In the centre of the diagram is the tile-design application around which the dialogue is based. The general data flow in the system is clockwise from the top left of this diagram. First, the speech recognition and pen-input interpretation modules perform initial processing on the input from the user; linguistic content from both of these modules is then translated by the language interpretation module into semantic forms. The fusion module then considers input from all sources and

[1] http://www.hcrc.ed.ac.uk/comic/

Figure 3.3: Architecture of the COMIC dialogue system



resolves cross-modal references as necessary to produce a unified, modality-independent representation of the user's request. The dialogue manager then selects the appropriate system response to this input, using information from knowledge sources including the dialogue history, an ontology of tile designs, and a model of user preferences. The modality-independent specification of the output is then sent to the fission module, which selects content for each of the output modalities and coordinates the output. The language generation, speech synthesis, and face control modules are then used to produce concrete output according to the fission-module specifications.

COMIC is implemented in the MULTIPLATFORM communication architecture (Herzog *et al.*, 2004). This architecture has also been used in several other dialogue-system projects, including VerbMobil (Wahlster, 2000), SmartKom (Wahlster, 2006), and IMIX (op den Akker *et al.*, 2005). MULTIPLATFORM is able to run systems consisting of a number of modules, possibly distributed across a number of heterogeneous computers. It is based around a publish-subscribe paradigm; that is, modules publish data on named pools (i.e., communication channels), and any module is able to subscribe to any pool to receive data sent on that channel. The full COMIC system incorporates 36 pools. The majority of the pools are used for direct communication between two modules—for example, there is a pair of pools used for queries to and responses by the user-model manager (Section 3.1.4.4). There are also pools to which a number of modules subscribe and publish: the most widely used pool is the one that controls turn-taking across the system.

3.1.2 Design application

The core of COMIC is a module based on a bathroom-design application from ViSoft,[2] a German software company. This module is labelled *Application ViSoft* in Figure 3.3 and is located at the centre of the diagram. Appendix A provides a number of screenshots of the application.

The design-application module interacts with the rest of the system in a number of ways; some of these are active in all of the phases of the interaction, while others are specific to particular phases. In all phases, it provides the following capabilities:

- When input is active, the design application processes the movements of the pen on the tablet (or the mouse on the screen), leaves ink trails on the screen, and sends the resulting pen streams to the pen-interpretation module for input processing.
- It publishes information about the objects displayed on its screen every time the screen state changes, both so that the pen-interpretation module can correctly process selection and erasing gestures and so that the output system can point accurately at the on-screen objects.
- In response to commands from the dialogue manager and the fission module, the design application changes what is displayed on its screen and switches among the different phases.

The following are the phase-specific commands supported by the design application:

- In the blueprint-specification phase, the design application displays “beautified” versions of the the recognised walls, windows, doors, and measurements on the screen.
- When the user is browsing through tiling options, the design application displays the designs specified by the dialogue manager as thumbnails, and also tiles the selected bathroom with the currently-active tile design, using the requested levels of borders and decoration.
- Also in the tile-browsing phase, the design application moves an on-screen pointer in response to commands from the fission application in order to give deictic gestures at on-screen objects.
- When the user chooses a three-dimensional tour of the room, the design application creates and plays a three-dimensional animated camera move through the room in its current configuration.

[2] http://www.visoft.de/

3.1.2.1 The ViSoft emulator

In addition to the main design application, COMIC also includes an emulator for the ViSoft application in tile-browsing mode. This module can be run on its own or in conjunction with the full design application, depending on the circumstances.

When the emulator is run on its own, it provides a full-screen, output-only emulation of the full design application in the tile-design browsing phase. This provides a lightweight version of the design application that can run on any computer. This makes it possible to run COMIC on a smaller number of less-powerful machines, for evaluations or demonstrations; the experiments in Chapter 4 use this interface, for example. The emulator also supports a richer set of pointer gestures than the full design application.

The emulator can also run in conjunction with the full design application. In phases other than tile-browsing, it remains invisible, but in the tile-browsing phase it provides several additional optional capabilities. First, it can place a menu bar across the top of the screen to provide an alternative input channel in addition to speech. As well, it can place a set of buttons over the thumbnails at the bottom of the screen to make them directly clickable instead of the user having to circle them to select them; if the buttons are placed on the the screen, the emulator is also able to use the richer pointer-gesture set. Finally, the emulator provides an additional method of changing the amount of borders and decoration in the selected tile design through direct manipulation, rather than speech; Appendix A.6 provides an example of this interface.

3.1.3 Input processing

In COMIC, as in most multimodal dialogue systems, input processing is a multi-stage process. First, the input modules interpret the signals from the user, which may include any combination of speech, pen/mouse gestures, and GUI selections. Next, the natural-language processing module parses and analyses any linguistic input from the speech recogniser and the pen-interpretation module. Finally, the fusion module resolves any cross-modal references and sends the dialogue manager a ranked list of hypotheses corresponding to the user input.

Turn-taking in COMIC is strictly half-duplex, with no barge-in; that is, while the system is producing output, the input-processing system is not active (i.e., the microphone is switched off and the GUI does not respond to any input). The end of a user turn is determined by the fusion module, as described in Section 3.1.3.4, using semantic and time-out information from the individual recognition modules.

After every system turn, the dialogue manager (Section 3.1.4.1) provides semantic expectations to the input modules to indicate the anticipated range of user responses to the most recent turn. For example, if the system has just asked the user to specify a wall length, the system will expect a dimension; after a *yes-no* question, on the other hand, it will expect one of those answers. The user is still free to provide answers other than what is expected, but the semantic expectations are helpful in processing and disambiguating the user responses.

3.1.3.1 Speech recognition

The COMIC automated speech-recognition (ASR) module (ten Bosch, 2005) is based around the HTK toolkit (Young *et al.*, 2000), a portable toolkit used for building and manipulating hidden Markov models that has been widely used for speech-recognition research. Since HTK itself does not provide real-time capabilities, the module is implemented as a wrapper around HTK which communicates with it using UNIX pipes; the wrapper re-formats the output of HTK to meet the requirements of COMIC's input processing system and responds to messages sent through the MULTIPLATFORM testbed as needed.

The language model used by the ASR module incorporates a BNF model as well as a bigram model trained on logged data from previous COMIC evaluations and was created to cover a range of possible user inputs in all phases of the COMIC system. The ASR module also incorporates a separate garbage model to allow the module to discard unrecognisable parts of its input and to create hypotheses from the rest. The module detects the end of a speech input by waiting for a configurable number of silence frames. Once the end of a segment is detected, it outputs the hypotheses for that segment and then carries on processing new input. The ASR module stops listening when the fusion module signals that the end of a user turn has occurred.

The output of ASR is a scored *n*-best list of hypotheses, which is sent to the natural-language processing module (Section 3.1.3.3) for further processing.

3.1.3.2 Pen-gesture and handwriting recognition

Pen gestures and handwriting are both recognised by a single module (Rossignol and Vuurpijl, 2005), which processes the pen streams and screen states from the design application and sends its output hypotheses to the fusion module. The pen module uses a KNN classifier (Willems *et al.*, 2005) trained on data gathered early in the COMIC project to determine the input mode. It is able to distinguish among handwriting, drawing, and deictic gestures, and among four classes of deictic gestures: tapping, encircling, erasing, and moving.

Encircling and tapping are both methods of selecting objects on the screen and are distinguished from one another using an empirically-determined threshold on the bounding box of the ink stream. A moving gesture consists of an arrow from the location of an on-screen object to a new location. An erasing gesture is produced by using "negative pressure" on the pen (i.e., tapping with the reverse end); the object to be erased is determined using the same process as for encircling and tapping. The pen module is also able to recognise compound objects; for example, in phase 1, the user is able to draw a wall and write its length on the screen as part of the same turn, or to write two lengths in a single turn.

Like the ASR module, the pen module also outputs scored hypotheses; linguistic hypotheses are processed further by the natural-language processing module, while object hypotheses are sent directly to the fusion module.

3.1.3.3 Natural-language processing

The input to the natural-language processing (NLP) module is the output of the handwriting- and speech-recognition modules, as well as the input expectations from the dialogue manager. The NLP module uses a template-based semantic parser (Engel, 2002) to create high-level representations of the input, which are then sent to the fusion module.

The semantic parser does not need a syntactic analysis; rather, the output structure is built directly from the word level. This is made feasible because the inputs are spoken utterances made to a dialogue system, which are usually shorter and syntactically uncomplicated. The approach also makes the system more robust against speech-recognition errors and syntactically incorrect user input. The NLP module incorporates a powerful rule language based on production systems, as well as a built-in ontology formalism which decreases the amount of work that must be done to port it to new dialogue systems.

The output of the NLP module is also a list of scored hypotheses representing the possible semantic interpretations of the input it is given, in the context of the current expectations. This output is sent to the fusion module.

3.1.3.4 Fusion

The fusion module in COMIC (Pfleger, 2004) combines the hypotheses from the various recognition modules, resolving cross-modal coreferences as necessary, and creates a ranked list of hypotheses to be sent to the dialogue manager. It is based on a production rule system and incorporates a high-level method of controlling the application of the production rules. This method has two parts: a goal stack that represents the focus of attention, and an activation process that determines the accessibility of objects in the working memory and of productions in the procedural memory.

Key to the approach is that every individual input event is interpreted and enriched with respect to its local turn context: all previously recognised input events, along with the dialogue state belonging to the current dialogue turn. This means that all of the knowledge on how to integrate events can be expressed in rules that operate on the context representation. Since each input module attaches confidence estimates to its output, the fusion module incorporates these confidence values into its processing to produce the final ranked hypotheses.

In addition to combining the individual input messages, the fusion module also has rules that determine when the user turn has ended, using a combination of semantic and time-out information from the input-processing modules. Once the fusion module has determined that the user turn is finished, it sends its hypotheses to the dialogue manager and disables the input modules; these modules remain disabled until the next system turn has been completed.

3.1.4 Dialogue management and knowledge sources

Once the fusion module has determined that the user turn is over, it sends its output to the dialogue manager. The dialogue manager processes the input and decides what the system should do and say in response, making use of a number of system resources. This section describes both the dialogue manager and the various knowledge-management modules.

3.1.4.1 Dialogue manager

In COMIC, the task of the Dialogue and Action Manager (DAM) (Catizone *et al.*, 2003) is to decide what the system should show and or say in response to a user input. The input to the DAM consists of multiple scored hypotheses from the fusion module, representing the user input at a high level independent of the channel on which it was given; the output of the DAM is a similar high-level specification of the desired system action. The DAM itself does not consider input or output modalities in its processing.

The COMIC DAM is a general-purpose dialogue manager which can handle different dialogue management styles including system-driven, user-driven and mixed-initiative. The general-purpose part of the DAM is a simple stack with a controller; all of the application-dependent information is stored in a variant of Augmented Transition Networks called *Dialogue Action Forms* (DAFs). These DAFs represent general dialogue moves, as well as sub-tasks or topics, and are maintained using the stack as the dialogue proceeds.

When processing a user input, the controller decides whether the DAM can stay within the current topic (and thus the current DAF) or whether a topic shift has occurred. In the latter case, a new DAF is pushed onto the stack and executed. After that topic has been exhausted, the DAM returns to the previous topic automatically. The same principle holds for error handling.

In the blueprint-drawing phase of the COMIC system, the DAM has an agenda of the information that the user must provide and prompts the user until all of the necessary data has been given. The DAM is prepared for the user to over-answer or to give information in a different order than was requested. In the guided-browsing phase of the COMIC system, the user may browse tiling designs by colour, style or manufacturer, look at designs in detail, or change the amount of border and decoration tiles. The DAM uses the system ontology to retrieve designs according to the chosen feature, and consults the user model and dialogue history to narrow down the resulting designs to a small set to be shown and described to the user.

3.1.4.2 Ontology manager

Information about all of the tile designs in the system is stored in an ontology represented in DAML+OIL (W3C, 2001). The ontology contains catalogue information including the manufacturer and series name, style, colours, and decoration, as well as any canned descriptive text associated with each design. It also stores the hierarchy of possible system and user dialogue acts; these are used to represent the high-level messages that are exchanged among the fusion, DAM, and fission modules.

3.1.4.3 Dialogue-history manager

For each design in the ontology, the dialogue-history manager keeps track of whether it has been mentioned in the dialogue. It also records the properties of each design that have been described to the user; note that a property may be described directly (e.g., *this design is classic*) or indirectly (e.g., *here are some classic designs* while indicating several designs). The dialogue history also stores the identity of the last design that was described.

The information in the dialogue history is used by both the dialogue manager when choosing the designs to present to the user, and the fission module to help decide how to describe the selected designs. In the first experiment described in Chapter 4, we show how information from the dialogue history is incorporated into the output-planning process, and demonstrate that it makes a perceptible difference to users of COMIC.

3.1.4.4 User-model manager

The user-model manager stores information about the user's domain preferences: that is, the features of tile designs that the user is believed to like and dislike. The user-model manager supports two types of queries. It can produce an overall ranking of a set of designs to help the dialogue manager to choose options that are relevant to the user; it can also produce a detailed evaluation of a single design, with numerical scores on each individual attribute, to help the presentation planner create descriptions focusing on the options that are most important for that user. The full details of how user preferences are represented are given in Section 4.1.

Like the dialogue history, the user model is also used by both the dialogue manager and the fission module when responding to user requests for tile designs. The second experiment in Chapter 4 describes how the impact of the user-model information on the generated text was evaluated; the generation processes in the talking-head experiments in Chapters 7–9 also make use of information from the user model.

3.1.5 Fission and output processing

Once the dialogue manager has decided on the content of the next system turn, it sends a specification of that content to the fission module. The fission module then selects and structures content to meet that specification, using a combination of the available modalities, and coordinates the presentation of that content. In this section, we give an overview of each of the modules involved in output planning and generation. In Section 3.2, we then give a detailed walkthrough of how a dynamic tile-design description is prepared and produced.

3.1.5.1 Fission

The input to the fission module is a high-level, modality-independent specification of the content of the next system turn; for example, for the output in Figure 3.2, the dialogue manager would simply indicate that the system should describe the design selected by the user. The fission module selects and structures content to meet such a specification, determines the content that should be sent to each individual output module, and coordinates the overall presentation. Not all of the output in COMIC is generated dynamically—in fact, in all of the phases of the interaction except for tile-browsing, the full multimodal content for all turns is scripted in advance.

The first step in creating an output turn is to create the textual content. For the dynamic tileset descriptions in the tile-browsing phase, the fission module makes use of the OpenCCG realiser as described in the following section. The process of creating dynamic output is described in detail in Section 3.2. For all of the scripted output, the fission module does not make use of the surface realiser at all—instead, it sends the canned text directly to the speech synthesiser.

As in many other multimodal generation systems, speech is the primary modality, and output in other modalities is planned to coincide with particular words in the speech. In the canned output segments, the multimodal commands are also specified in the scripts; for the dynamic output, multimodal content is selected based on the syntactic, semantic, and pragmatic information in the generated text. Cross-modal coordination is achieved by sending the text of a sentence to the speech synthesiser in advance, with marks on all of the words that have multimodal coarticulations. The timing information that is returned by the synthesiser is then used to set the final schedule for the other modalities.

The fission module plans and generates its output incrementally, using the time taken to produce the earlier parts of a turn to prepare the later parts; this allows output to begin as soon as possible, and also reduces the delay between segments of the output. The technical details of the output preparation, execution, and coordination processes are described in Appendix B.

3.1.5.2 Surface realisation

For the dynamic parts of the COMIC output, surface realisation is performed by the OpenCCG[3] realiser, a practical, open-source realiser based on Combinatory Categorial Grammar (CCG) (Steedman, 2000b). It employs a novel ensemble of methods for improving the efficiency of CCG realisation and, in particular, makes integrated use of n-gram scoring of possible realisations in its chart realisation algorithm (White, 2005, 2006b). The n-gram scoring allows the realiser to work in "anytime" mode—able at any time to return the highest-scoring complete realisation—and ensures that a good realisation can be found reasonably quickly even when the number of possibilities is exponential. This makes it particularly suited for use in an interactive dialogue system such as COMIC.

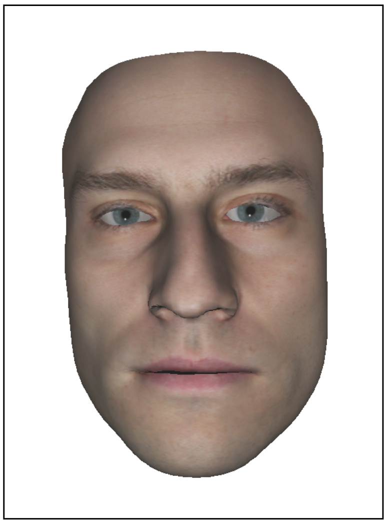
In COMIC, the OpenCCG realiser uses factored language models (Bilmes and Kirchhoff, 2003) over the words to select the highest-scoring realisation licensed by the grammar that satisfies the specification given by the fission module. The language models are based on a testbed of 549 target sentences covering the range of sentence structures produced by the grammar. Steedman's (2000a) theory of information structure and intonation is used to constrain the choice of pitch accents and boundary tones for the speech synthesiser, based on the information structure provided by the fission module.

3.1.5.3 Speech synthesis

The COMIC speech-synthesis module is implemented as a client to the Festival speech synthesis system (Clark *et al.*, 2004), using a custom-built Festival 2 voice with support for APML prosodic annotation (de Carolis *et al.*, 2004). This voice respects the pitch accents and boundary tones selected by the OpenCCG realiser or specified in the canned text; previous exper-

[3] `http://openccg.sourceforge.net/`

Figure 3.4: The COMIC talking head



iments have shown that synthesised speech with contextually appropriate prosodic features can be perceptibly more natural (Baker *et al.*, 2004).

Since the fission module needs the timing information for the words and phonemes so that it can set the schedules for the other modalities, the speech-synthesis module works in two stages. When it receives a piece of text, it prepares and stores the waveform for that text by sending it to the Festival server, and returns the timing information for the words, phonemes, and any multimodal coarticulation points. The stored waveform is then later played in response to a message from the fission module when the prepared turn is actually output.

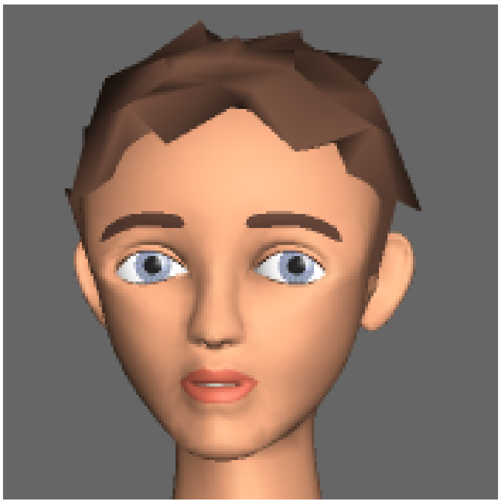
3.1.5.4 Talking head

The COMIC talking head (Cunningham, 2005) is a real-time, on-line, controllable computer head animation system; Figure 3.4 shows a screenshot of the head. The movements of the head are created using a combination of blend-shape and motion-capture animation (Breidt *et al.*, 2003): three-dimensional head scans provide the high-resolution spatial structure of the face at the peak of the animation, while motion capture determines the blend-shape weights from the high-resolution motion trajectories of a limited number of markers.

The talking head is able to produce proper lip motion for speech, simple visual emphasis to accompany prosodic emphasis in the speech, a range of conversational facial expressions, and a variety of "breath of life" animations when it is idle. Like the speech synthesiser module, it works in two stages: it first prepares the animation for a turn based on the schedule from the fission module, and then plays the prepared animation when it is time to output the turn.

Figure 3.5: Alternative talking heads

(a) Greta



(b) RUTH

Alternative talking heads In addition to the main talking head, we have also created alternative modules based on two existing heads: Greta (de Rosis *et al.*, 2003) and RUTH (DeCarlo *et al.*, 2004) (Figure 3.5). Like the ViSoft emulator (Section 3.1.2.1), these heads also require less computing power and support an extended set of output specifications. They also allow demonstrations and evaluations to be run on fewer or less powerful computers than are required for the full system. Chapter 6 gives a detailed description of the extended facial-display capabilities of the RUTH head, which was used in the experiments described in Chapters 7–9.

3.2 Generating dynamic output

In this section, we give a more detailed description of the process for generating dynamic output within COMIC. The experiments presented in the following chapters all describe enhancements to add variation to the process outlined here: Chapters 4–5 describe enhancements to the textual content, Chapters 7–8 describe more sophisticated methods of choosing motions for the talking head, while Chapter 9 combines variation on both of these output channels.

As mentioned in Section 3.1.5.1, a large percentage of the output generated by COMIC is canned: acknowledgements, error-handling prompts, and requests for room dimensions are all stored as complete multimodal scripts and played where appropriate. However, there are also parts of the output that are generated dynamically. In particular, each description of one or more tile designs (as in Figure 3.6) is individually generated, potentially taking into account contextual information from the dialogue history and the user's preferences.

Figure 3.6: Sample COMIC output (repeated from Figure 3.2)

THIS DESIGN is CLASSIC.
[circle tiles]
It uses tiles from the ALT METTLACH collection by VILLEROY AND BOCH.
[point at manufacturer name]
As you can see, the colours are DARK RED and OFF WHITE.
[point at tiles]

In the pipeline view of natural language generation (Section 2.1.2), many steps involve converting between increasingly specific tree structures. As Wilcock (2001) points out, this sort of tree transformation is a task to which XML—in particular, XSLT template processing (W3C, 1999)—is particularly suited. In COMIC, we plan text by treating the XSLT processor as a top-down, rule-expanding planner that translates dialogue manager specifications into logical forms for the OpenCCG text realiser; this process is outlined in Sections 3.2.1–3.2.3. Multimodal behaviours are chosen to accompany the generated text, as described in Section 3.2.4.

Using an external realiser at the end of the planning process provides two advantages. First, we can use the realiser to deal with those aspects of surface realisation that are difficult to implement in XSLT, but that the realiser is designed to handle (e.g., syntactic agreement via unification). Second, we take advantage of OpenCCG's use of statistical language models by sending multiple alternative logical forms to the realiser, and having it make the final choice of surface form. In the basic system, this is done by multiplying out the options and realising them successively; in Chapter 5, we describe an enhanced version using an updated version of OpenCCG that is able to realise directly from disjunctive logical forms. Allowing the text planner to produce multiple alternatives also eliminates the need for backtracking, which is not something that is otherwise easily incorporated into a system based on XSLT processing.

3.2.1 Content selection

As described in Section 3.1.4.1, the dialogue-manager specification for a tile-design description is high-level and modality-independent. For example, for the output shown in Figure 3.6, the DAM might have simply specified that the system should describe a particular design, and that the description must include a mention of the style. The first task of the fission module in planning the content of such a description is therefore to choose from the ontology the facts about the target design to include in the description.

The maximum length of a design description is normally three facts about the design under consideration. The fission module always includes any features specifically requested by the dialogue manager; in its default configuration, it then chooses the rest of the content (up to the maximum length) arbitrarily. In Section 4.2.1, we describe how information from the dialogue history and the user model is incorporated into this process.

Figure 3.7: Initial messages (unaggregated)

```
<messages>
  <msg id="t2-1-5" prop="has_colour" type="prop-has-val">
    <slot name="object" value="tileset9"/>
    <slot name="value" value="terracotta beige"/>
  </msg>
  <msg full-sentence="false" id="t2-1-6" prop="has_colour" type="canned-text">
    <slot name="object" value="tileset9"/>
    <slot name="value" value="give-tuscan-feeling"/>
  </msg>
</messages>
```

3.2.2 Content structuring

The work described in this section and the next both fall under the heading of microplanning in the three-stage NLG pipeline architecture outlined in Section 2.1.2; we describe them separately because the two tasks are done by distinct components within the fission module.

The result of content selection is an unordered set of tile-design features to describe; this set is converted into a text plan as follows. First, the selected messages are put into a linear order, using a number of heuristics. In the default configuration, the only heuristics are as follows:

- Any features explicitly requested by the dialogue manager are put at the start of the description.
- The designer, series, and style always come before the colour and decoration.

Since these heuristics impose only a partial order on the messages, a total order is created from the partial order by breaking ties at random.

The ordered list of selected features is then converted into a list of XML messages. Figure 3.7 shows two such messages; the first describes the colours of the design (terracotta and beige), while the second specifies a piece of canned text[4] that can be said about the colours (that they "give the room the feeling of a Tuscan country home"). Each element in this list contains the relevant information from the ontology. If the dialogue history and the user model are used, information from these sources is also included.

The next step is to aggregate the list of messages—that is, to create composite messages from the initial flat list so that the resulting generated text has a more interesting and varied structure. In many NLG systems, aggregation is done at the syntactic level; in COMIC, we instead work at the conceptual level. Thanks to the fact that we produce multiple alternative syntactic structures for the realiser (see Section 3.2.3), we can be confident that, whatever the final set of messages, there will be a syntactic structure available to realise them.

[4] Canned-text commentary is represented in the realiser lexicon as a multi-word verb.

Figure 3.8: Aggregation template (simplified)

```
<xsl:template match="messages">
  <xsl:variable name="void" select="set:clear()"/>
  <messages>
    <xsl:for-each select="msg">
      <xsl:variable name="next" select="following-sibling::msg[1]"/>
      <xsl:choose>
        <!-- Return nothing if we have already processed this message. -->
        <xsl:when test="set:contains(@id)"/>

        <!-- Add canned text to a sentence. -->
        <xsl:when test="@prop=$next/@prop and @type='prop-has-val'
          and $next/@type='canned-text' and not($next/@full-sentence='true')">
          <msg type="same-prop-canned-text" id="{concat(@id, '+', $next/@id)}">
            <slot name="prop"> <xsl:copy-of select="."/> </slot>
            <slot name="text"> <xsl:copy-of select="$next"/> </slot>
          </msg>
          <xsl:variable name="void" select="set:add(string(@id))"/>
          <xsl:variable name="void" select="set:add(string($next/@id))"/>
        </xsl:when>
        <!-- ... other tests ... -->

        <!-- Nothing matched: just copy the message across. -->
        <xsl:otherwise> <xsl:copy-of select="."/> </xsl:otherwise>
      </xsl:choose>
    </xsl:for-each>
  </messages>
</xsl:template>
```

Figure 3.9: Combined messages

```
<messages>
  <msg id="t2-1-5+t2-1-6" type="same-prop-canned-text">
    <slot name="prop">
      <msg id="t2-1-5" prop="has_colour" type="prop-has-val">
        <slot name="object" value="tileset9"/>
        <slot name="value" value="terracotta beige"/>
      </msg>
    </slot>
    <slot name="text">
      <msg full-sentence="false" id="t2-1-6" prop="has_colour"
              type="canned-text">
        <slot name="object" value="tileset9"/>
        <slot name="text" value="give-tuscan-feeling"/>
      </msg>
    </slot>
  </msg>
</messages>
```

Figure 3.10: Sentence-planning templates (simplified)

```
<xsl:template match="msg[@type='prop-has-val' and @prop='has_colour']" mode="s">
  <node pred="be" tense="pres">
    <rel name="Arg">
      <node pred="tile" det="the" num="pl"/>
    </rel>
    <rel name="Prop">
      <xsl:apply-templates select="slot[@name='value']" mode="np"/>
    </rel>
  </node>
</xsl:template>

<xsl:template match="msg[@type='same-prop-canned-text']" mode="s">
  <node pred="elab-rel">
    <rel name="Core">
      <xsl:apply-templates select="slot[@name='prop']/msg" mode="s"/>
    </rel>
    <rel name="Trib">
      <xsl:apply-templates select="slot[@name='text']/msg" mode="vp"/>
    </rel>
  </node>
</xsl:template>
```

Aggregation is done by a set of XSLT templates that combine adjacent messages based on various criteria. For example, the template shown in Figure 3.8 combines a feature-value message with the associated canned-text commentary.[5] Figure 3.9 shows the combined message that results when the messages in Figure 3.7 are processed by this template.

3.2.3 Sentence planning

After the content of a description has been selected and structured, the logical forms to send to the realiser are created by applying further XSLT templates to the result. Every such template matches a message with particular properties and produces a logical form for the realiser, possibly combining the results of other templates to produce its own final result. XSLT modes are used to select different templates in different target syntactic contexts.

Two sample templates are shown in Figure 3.10. The first template produces the logical form for a sentence (`mode="s"`) describing the colours of a tileset (e.g., *The tiles are terracotta and beige*). The second template creates a logical form representing a commentary message as a verb phrase (`mode="vp"`) and then appends it as an elaboration to a sentence about the same property. When the messages shown in Figure 3.9 are transformed by these templates, the result is the logical form shown in Figure 3.11, which corresponds to the sentence *The tiles are terracotta and beige, giving the room the feeling of a Tuscan country home*.

[5]This template is simplified; aggregation is actually performed in several passes to allow multi-level aggregation. The set namespace refers to a Java *Set* instance that stores message IDs to avoid processing a message twice.

Figure 3.11: Generated logical form

```
<!--
  The tiles are terracotta and beige,
  giving the room the feeling of a Tuscan country home.
-->
<lf id="t2-1-5+t2-1-6">
  <node mood="dcl" info="rh" pred="elab-rel" id="n7">
    <rel name="Core">
      <node tense="pres" id="n2" pred="be">
        <rel name="Arg">
          <node det="the" pred="tile" id="n1" num="pl"/>
        </rel>
        <rel name="Prop">
          <node id="n3" pred="and">
            <rel name="List">
              <node id="n4" kon="+" pred="terracotta">
                <rel name="Of">
                  <node idref="n1"/>
                </rel>
              </node>
              <node kon="+" id="n6" pred="beige">
                <rel name="Of">
                  <node idref="n1"/>
                </rel>
              </node>
            </rel>
          </node>
        </rel>
      </node>
    </rel>
    <rel name="Trib">
      <node id="n8" pred="give-tuscan-feeling">
        <rel name="Arg">
          <node idref="n1"/>
        </rel>
      </node>
    </rel>
  </node>
</lf>
```

Figure 3.12: Logical form containing alternatives

```
<lf id="t2-1-2">
  <!-- This design is ... -->
  <node tense="pres" mood="dcl" info="rh" pred="be" id="n13">
    <rel name="Arg">
      <node id="n1" num="sg" pred="design" kon="+">
        <rel name="Det"> <node kon="+" pred="this" id="n18"/> </rel>
      </node>
    </rel>
    <rel name="Prop">
      <one-of>
        <!-- ... in the country style. -->
        <node pred="in" id="n14">
          <rel name="Fig"> <node idref="n1"/> </rel>
          <rel name="Ground">
            <node num="sg" det="the" pred="style" id="n15">
              <rel name="HasProp">
                <node id="n16" kon="+" pred="country"/>
              </rel>
            </node>
          </rel>
        </node>

        <!-- ... country. -->
        <node id="n20" kon="+" pred="country">
          <rel name="Of">
            <node idref="n1"/>
          </rel>
        </node>
      </one-of>
    </rel>
  </node>
</lf>
```

Many messages can be realised by several different logical forms. For example, to tell the user that a particular design is in the country style, the options include *This design is in the country style* and *This design is country*. Often, the text planner has no reason to prefer one alternative over another. Rather than picking an arbitrary option within the text planner (as did, e.g., van Deemter *et al.* (2005)), we instead defer the choice and send all of the valid alternatives to the realiser in a packed representation. This makes the implementation of the text planner more straightforward and adds a level of stochastic variation to the output. Figure 3.12 shows such a logical form, incorporating both of the above options under a `<one-of>` element.

To process a logical form with embedded alternatives, the realiser makes use of the same *n*-gram language models that it uses to guide its search for the realisation of a single logical form. The version of OpenCCG used in COMIC was not able to directly handle the realisation of logical forms with embedded alternatives, so in the COMIC system the packed alternatives are first multiplied out into a list of top-level alternatives, whose order is randomly shuffled. The realiser then computes the best realisation for each top-level alternative in turn, keeping track of the overall best scoring complete realisation, until either the anytime time limit is reached or the list is exhausted. To allow for some stochastic variation, a new realisation's score must exceed the current best one by a certain threshold before it is considered significantly better. In Chapter 5, we describe an implementation of stochastic variation for COMIC in which an updated version of OpenCCG is used to realise directly from disjunctive logical forms, which is more computationally efficient than this option-multiplication process (White, 2006a).

3.2.4 Specifying multimodal content

In addition to creating the text of a description, the fission module also specifies the content of that turn in the other modalities: deictic gestures at objects on the screen with the pointer, pitch accents and boundary tones for the speech synthesiser, and facial displays for the talking head. Appendix B describes how the selected content is coordinated temporally across the modalities when a turn is generated.

The fission module specifies a deictic gesture at every object that has an on-screen referent, using the on-screen pointer. The type of the gesture (circling or moving the pointer) is completely determined by the type of the object. For example, the system always circles a design thumbnail, while it points to manufacturer names—these are the only two types of gestures supported by the full ViSoft design application.

Prosodic specifications for the speech synthesiser are generated directly by the realiser module along with the text, using Steedman's (2000a) theory of information theory and intonation. The logical forms that are sent to the realiser such as Figure 3.11 specify both the desired information structure (e.g., an `info="rh"` attribute on a node indicates that the subtree rooted at that node forms part of the rheme of the utterance) and the words that should be given contrastive pitch accents (indicated by `kon="+"`); the realisation module then selects the appropriate pitch accents and boundary tones to agree with that specification.

Facial emphasis commands are planned to coincide with pitch accents in the speech and are selected based on the text produced by the realiser. In the basic configuration, the fission module associates a facial emphasis with every *L+H** accent[6] in the speech; if there is no *L+H**, it chooses the last *H** in each sentence for an emphasis command. The emphatic motion that is used by the head is the same in all circumstances. In the talking-head experiments in Chapters 7–8, we describe several implementations of data-driven facial-display selection in which we vary both the distribution and the type of the facial motions; these implementations use the RUTH talking head for output (Figure 3.5(b)), which supports a wider range of commands than the default COMIC head (Figure 3.4).

[6] *L+H** and *H** are ToBI pitch-accent specifications; see Beckman *et al.* (2005) for details.

3.3 Summary

In this chapter, we have presented the COMIC multimodal dialogue system, first giving an overview of the architecture and components and then concentrating in more detail on the components that create the multimodal output. In its distributed modular architecture, with individual components responsible for specific input, output, or reasoning tasks, COMIC fits in with the emerging standard for multimodal dialogue systems (Allen *et al.*, 2001; van Kuppevelt *et al.*, 2005): similar architectures have been and are being used in a wide range of systems including WITAS (Lemon *et al.*, 2002), AdApt (Gustafson *et al.*, 2002), MATCH (Johnston *et al.*, 2002), Max (Kopp *et al.*, 2003), SmartKom (Wahlster, 2006), IMIX (op den Akker *et al.*, 2005), TALK (Lemon *et al.*, 2006), and JAST (Foster *et al.*, 2006). The components of COMIC themselves—the MULTIPLATFORM testbed, the OpenCCG surface realiser, the Festival speech synthesiser, and the RUTH talking head—are all state-of-the-art and have been included in several other systems.

The dynamic output generated by COMIC is of a type that has been widely studied: as described in Section 2.3, describing and comparing the features of objects drawn from a database is a task that dates back to some of the earliest natural-language generation systems and is still current today, while using an animated talking head to accompany spoken output is an output channel whose popularity has grown significantly in recent years (Section 2.4.2). The main sources of contextual variation in COMIC are the dialogue history and a model of user preferences, both of which have again been used widely as described in Sections 2.5.1–2.5.2; basing stochastic choices on the contents of a corpus is also an implementation technique that has been employed to add variation to other generation systems such as those mentioned in Section 2.5.4.

In summary, COMIC uses a standard architecture and state-of-the-art components and incorporates variation into its output using a range of standard techniques and information sources. This means that the results of the studies in the body of this thesis are relevant to a substantial set of past and current generation systems. In Section 10.2, we return to this topic and discuss the implications for other systems in more detail.

Chapter 4

Context-tailored textual descriptions*

> I should venture to assert that the most pervasive fallacy of philosophic thinking goes back to neglect of context.
>
> John Dewey, *Context and Thought*

In this first set of studies, we concentrate on the textual descriptions generated by the COMIC output system and measure the impact of rule-based contextual enhancements to all levels of the text planner. We make use of two types of contextual information: the dialogue history and a model of the user's likes and dislikes. As described in Sections 2.5.1–2.5.2, both of these information sources have been used to drive decision-making in a number of generation systems, and evaluations have generally shown that they have an impact. In this chapter, we perform experiments designed to confirm that this type of contextual variation also has a perceptible effect on the texts generated by COMIC.

This chapter addresses the first research question of the thesis: investigating the impact of different types of tailoring on the quality of generated output. In this case, we implement rule-based tailoring based on two different factors drawn from the dialogue context, and use an evaluation technique with human judges acting as "overhearers" in the dialogue and selecting the most appropriate system output in context. The results of this study give an indication of the circumstances under which this type of tailoring is perceptible; we also indirectly measure whether users appreciate this type of tailoring through the participants' selection of the version that they believe the hypothetical user would prefer.

*This chapter is based on Foster and White (2005).

The chapter is arranged as follows. In Section 4.1, we first describe in detail the model of user preferences that was used in COMIC, which is similar to the user-preference models used in several recent systems. Section 4.2 then outlines how the text-generation process described in Section 3.2 was modified to take into account information from the user model and the dialogue history at all stages. After that, we describe two human-evaluation studies, each designed to assess the individual impact of one of these sources of rule-based variation: the dialogue history in Section 4.3, and the user model in Section 4.4. Finally, in Section 4.5, we discuss the results of these two evaluation studies in the context of the thesis as a whole.

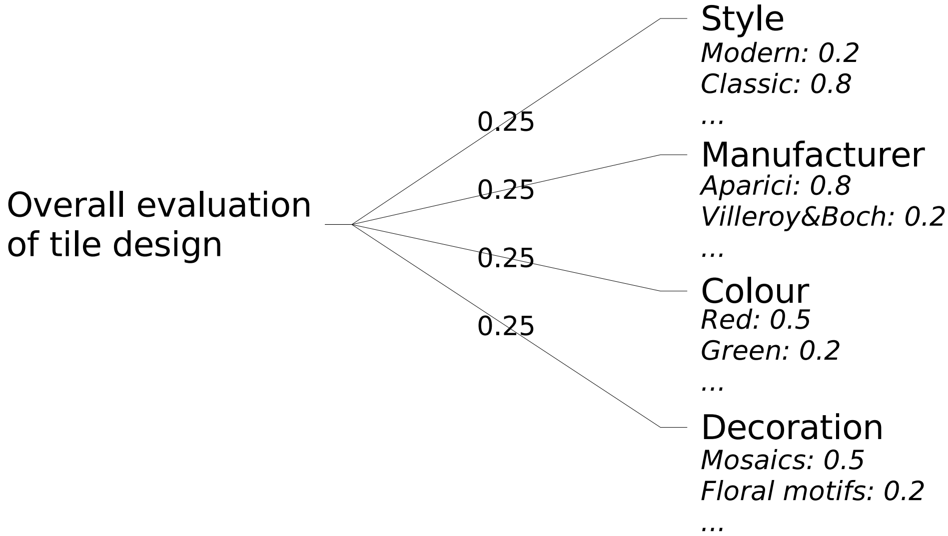
4.1 Modelling user preferences in COMIC

As in several of the previous systems described in Section 2.5.2, user preferences are represented in COMIC using an Additive Multiattribute Value Function (AMVF), which is a concept drawn from Multiattribute Utility Theory (Clemen, 1996). This type of model is 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. An AMVF is represented as a tree that models a user's values and preferences with respect to the entities in a class. The nodes in the tree correspond to a hierarchy of features of the entities; in the COMIC domain, the entities are tile designs, while the features are the colour, style, manufacturer, series, and decoration. Associated with each leaf node in the tree is a function that maps from feature values to a score between 0 (worst) and 1 (best). Each arc of the tree has an associated weight that indicates how important that part of the hierarchy is to the user. The overall evaluation of an entity is then the weighted sum of its evaluation on each individual feature.

The AMVF defines the user-independent structure of the user model. To create a model for a specific user, two types of information are needed: the values for the individual functions, and the weights on the arcs. The SMARTER procedure (Edwards and Barron, 1994) has been shown to be a reliable and efficient way of eliciting multi-attribute decision models. Users are asked questions designed to elicit an ordering of the possible features in the domain; these rankings are then translated into weights on the tree arcs. For the features which may have preferred and dispreferred values (e.g., the style of a house, the cuisine of a restaurant, or the airline of a flight), users must also be asked directly for their likes and dislikes, which are again translated into scores. Carenini and Moore (2006) describe this process in detail.

The tile-design user model in COMIC is made up of four features: style, colour, designer, and decoration. As in MATCH (Walker *et al.*, 2004), these features are represented in a one-level tree, as shown in Figure 4.1. The evaluation function for each of these features assigns a score between 0 and 1 to every possible value of that attribute. A score of 0.5 represents a neutral

Figure 4.1: Structure of the COMIC user model



evaluation; scores above or below that value indicate that the user respectively likes or dislikes that value, with the distance from 0.5 corresponding to the strength of that preference. For the *colour* and *decoration* attributes, which may have multiple values on a single design, the score is computed by combining the individual attribute values as follows: if the user does not like or dislike any of the values that a given design has for an attribute, or if there are values that have each of the evaluations, then the tile design receives a neutral score on that attribute; if the user likes any of the values, that attribute gets a positive score; and if the user dislikes any value, then it gets a negative score. In a general multi-attribute decision model of the type described above, a specific user's preferences are captured by setting the weight of each attribute, in addition to the evaluation function; however, in COMIC, we do not use the SMARTER procedure to elicit weights, so all attribute weights are set to 0.25 and the overall evaluation is simply the mean of the individual scores.

There are two ways that a user model can be defined in COMIC. One possibility is that a complete model is created and stored offline, before the dialogue begins, and is then selected and loaded at the start of the interaction. The other possibility is that the dialogue begins with a neutral user model, which then gets updated during the course of the interaction to take into account emerging preferences. In the current system, this is implemented by increasing the score for attribute values that the user specifically requests. For example, if the user asks for designs with blue tiles, the score for the colour blue is increased in the model for the remainder of the dialogue. There is no support in COMIC for dynamically decreasing scores during the interaction, or for explicitly discussing the user model as part of the dialogue; in the terminology of Kay (2006), the models are not *scrutable* in any way.

4.2 Implementation: Adaptive generation

Information from the user model and the dialogue history has an influence at all stages of the text-planning process outlined in Sections 3.2.1–3.2.3. This section outlines the main ways in which the dynamic-generation process was modified to use the adaptive information. (4.1) shows an example of the textual content of an adaptively-generated description generated by the enhanced process described here, while (4.2) shows the corresponding non-adaptive text; the parts of the output that are a result of the adaptive modifications are underlined.

(4.1) <u>As I said before</u>, this design is classic. It uses tiles from the Alt Mettlach collection by Villeroy and Boch, <u>though</u>. As you can see, the colours are dark red and off white.

(4.2) This design is classic. It uses tiles from the Alt Mettlach collection by Villeroy and Boch. As you can see, the colours are dark red and off white.

4.2.1 Content selection

In the basic COMIC content-selection procedure (Section 3.2.1), the system makes sure to include any attributes specifically requested by the dialogue manager, and then selects the rest of the facts at random, up to a maximum length of three facts. In the adaptive version, the planner still prefers the attributes that were explicitly requested; however, it uses more sophisticated techniques to choose the rest of the content, as follows.

First, the planner includes all attributes that have not been previously mentioned and that have a non-neutral evaluation in the user model, in decreasing priority of user-model score, up to the maximum length. If there is still space in the description, it then includes any other features with neutral evaluations that have not previously been mentioned, in an arbitrary order. Finally, if all of the selected facts have negative user-model evaluations, the planner includes a feature with a non-negative evaluation if possible, even if it has previously been described and even if the maximum length has been reached.[1] This process is a simplified version of the more sophisticated content-selection process based on *compellingness* used in systems such as GEA (Carenini and Moore, 2006) and FLIGHTS (Moore *et al.*, 2004).

4.2.2 Content structuring

The basic content-ordering heuristics are specified in Section 3.2.2. When the additional contextual information is available, the COMIC presentation planner also employs the following additional heuristics:

[1] This was done in order to avoid having a description composed entirely of features that the user dislikes, because the content-structuring heuristics described in the next section rely on finding a non-negative feature to use at the start of the description.

- Features requested by the dialogue manager are still put at the start, as in the basic version.
- Next come all features of this design that have not been previously described to the user: those features that have a non-neutral user-model evaluation come first, in decreasing order of user-model score, and then those with a neutral evaluation come at the end.
- After that are any features that have already been said, if any have been selected: again, those with a non-neutral evaluation come before those with a neutral evaluation, in decreasing user-model order.

Within each equivalence class, the preference for designer, series, and style over decoration and colour still holds.

The rules for structuring content reflect the reasons for having selected that content in the first place: features that we expect the user to be interested in get placed earlier in the description, as do specifically requested features, while features with positive evaluations are placed before those with negative evaluations. This is again similar to the procedure used by systems like GEA and FLIGHTS, which also base their structuring decisions on the reasons for having selected particular content.

After the messages have been sorted in this way, if the first fact in the list has a negative evaluation, a non-negative feature is moved to the front of the list. This is done because the first sentence of a description always introduces the design being described by mentioning one of its features (e.g., *This design is in the classic style*), and we wanted to avoid using a negative feature in this context.

The aggregation procedure, which combines adjacent messages into complex sentences, also takes into account the dialogue-history and user-model features of the messages that are selected. For example, a fact with a high user-model evaluation can be combined with a fact with a low evaluation to create a contrastive sentence, resulting in sentence structures like *although it is classic, it does have blue tiles*. Similarly, a feature that is common between the current design and the previous one can be merged with one that differs between the two to produce other contrastive combinations such as *once again it is family, but here there is artwork on the decorative tiles*.

4.2.3 Sentence planning

In addition to affecting the content that is presented in a description, the user model and dialogue history also influence its surface form. For example, if a description is to include a fact that we have already told the user, we signal this repetition with phrases such as *as I said* *before* or *as I mentioned earlier*;[2] if we are describing a property that we believe the user does not like, we add words such as *though* or *although* to the sentence. Such words and phrases are added to the logical forms by the sentence-planning templates described in Section 3.2.3, based on the contextual features included in the messages created by the extended content-structuring process described above.

4.3 Human evaluation: Perception of dialogue-history tailoring

In this first experiment, we measured the impact of the dialogue-history tailoring by having human judges observe recorded interactions between the system and a user, and asking them to choose which of two possible system outputs—one tailored to the dialogue history, and one not—was more appropriate. Participants made an initial selection based on the multi-modal version of the presentation (including speech and graphics, but no talking head), and then chose again based on a transcript of the text. Like Walker *et al.* (2004), we used an "overhearer" paradigm in which participants observed and judged simulated interactions; as described in Section 2.6.1, this is a paradigm that allows judgements to be gathered during the course of a dialogue rather than at the end, and that allows multiple alternative outputs to be compared in the same dialogue context.

This experiment is similar to that used by Walker *et al.* (2004) to evaluate the tailored output generated by the MATCH system; however, there are two differences. First, the MATCH participants judged all of the outputs first using text, and then judged them all again in speech; this presentation order was chosen to prime the proper nouns in the domain so that they could be recognised in speech. The presentation modality had no impact on the judgements of appropriate tailoring. In this experiment, we presented the speech first because speech is the primary presentation modality for COMIC and we wanted to measure the ability of participants to detect tailoring in the context of the actual system; also, all of the relevant proper names (manufacturer and series names) were displayed on the screen at all times. The text presentation was shown afterwards to allow the modalities to be compared. The second difference is that participants in the MATCH study answered Likert-scale questions to assess the quality of each output individually, while we used a forced-choice study. We made this choice because, for a transient medium like speech, direct judgements are difficult to make consistently, especially when the differences are subtle; cf. (Baker *et al.*, 2004; Rocha, 2004).

The results of previous evaluations of the impact of dialogue-history tailoring indicate that it generally has an effect. For example, Karasimos and Isard (2004) found that participants subjectively preferred texts generated by the M-PIRO system when they included explicit com-

[2] The choice between these two realisations is left to random choice in the realiser.

parisons between the object being presented and those that came earlier in the discourse; the learning outcomes of participants using the texts with comparisons were also higher than those of subjects using the baseline texts. In this experiment, we measured only the subjective impressions of the subjects.

Based on the results of the study described above, we had the following two hypotheses for this experiment:

1. Participants will select the versions generated with the dialogue history as appropriate significantly more often than those generated without tailoring.

2. There will be no difference in the pattern of responses between the two presentation modalities.

4.3.1 Participants

Participants in this experiment were recruited via an email to the Edinburgh University Informatics departmental student mailing list and were compensated for their participation in the experiment. The details of the participants were as follows:

Total number	25		
Gender	Female: 5	Male: 20	
Age	Under 20: 6	20–29: 15	30 and over: 4
Computer experience	Beginner: 0	Middle: 3	Expert: 22
Native language	English: 19	Other: 6	

4.3.2 Methodology

A modified, output-only version of the full COMIC dialogue system was used for this experiment. This system included the ViSoft emulator (Section 3.1.2.1) and all of the output-generation components except for the talking head; none of the input-processing, dialogue management, or knowledge components described in Sections 3.1.3–3.1.4 were enabled. A test harness was written that used the COMIC output system to play back scripted output and to gather and log participants' responses. The interface for this experiment was installed on the Edinburgh University Informatics internal network, so participants were able to take part in in the experiment at any time by logging into a departmental computer and running the program. A detailed description of the test harness, with screenshots and the full text of the instructions, is given in Appendix C.1.2.

This system was used to present two versions of each of six short (two-exchange) dialogues between a user and the system, where both the user input and the system output were played using speech and pointer gestures. The dialogues were fully synthesised in advance so that every participant saw and heard exactly the same version of each system turn; the user turns were simulated by playing recorded speech and simulating clicks on the user interface. For each pair of dialogues, one included system turns generated with the dialogue-history tailoring enabled, while the other had that tailoring disabled. The presentation order of the six dialogues was randomised individually for each participant, and the order of presentation within a dialogue pair was balanced so that the tailored version was first in three and the non-tailored version first in the other three.

Participants were instructed to pay attention to how the system responded to the user's requests and how it kept track of what had already been said in the conversation. After a participant had seen and heard both versions of a dialogue, they were asked the following question: *Which conversation had a more natural sequence of turns?* The choice was forced; the experiment could not continue until the participant had made a selection. Once the participant had chosen based on the multimodal presentation, they were then shown the transcripts of both versions of the interaction and asked again to respond to the same question.

4.3.3 Materials

For this experiment, we created a set of six short dialogues between COMIC and the user, using a neutral user model (i.e., one where all of the attributes had a score of 0.5). Each dialogue had two system turns. The system turns were created by manually creating the fusion-module output for each user turn and sending it to the dialogue manager, and then storing the full specification of the resulting generated output. The tile designs used in each dialogue were selected arbitrarily, with the restriction that some properties had to be in common across the two designs to make particular dialogue-history adaptations possible. We generated two versions of each dialogue: one in which the second system turn made use of the dialogue-history context provided by the first turn, and one in which it did not.

The set consisted of two dialogues in each of the following three classes, representing the three different ways in which the dialogue history can affect the generated output:

Also COMIC makes links between features in the current description and those in the preceding description (e.g., *this design is also classic*).

Repeat Whenever possible, COMIC avoids repeating information that it has already said about a design.

Said-before When COMIC does repeat previously-mentioned information—for example, because there is nothing new to say about a design—it signals the repetition by saying *as I said before* or *as I mentioned earlier*.

Figure 4.2: Also-class dialogue with dialogue-history tailoring enabled (4.3) and disabled (4.4)

(4.3)
a. **User:** Tell me about this design *[click on Lollipop]*
b. **COMIC:** Here is a family design. Its tiles are from the Lollipop collection by Agrob Buchtal. The colours are blue and green.
c. **User:** What about this one *[click on Asterix]*
d. **COMIC:** <u>Here we have another design in the family style.</u> Its tiles are from the Asterix collection by Villeroy and Boch. It has comic motifs on the decorative tiles.

(4.4)
a. **User:** Tell me about this design *[click on Lollipop]*
b. **COMIC:** This design is in the family style. The tiles are from the Lollipop collection by Agrob Buchtal. The tiles have a blue and green colour scheme.
c. **User:** What about this one *[click on Asterix]*
d. **COMIC:** <u>This design is in the family style.</u> Its tiles are from the Asterix collection by Villeroy and Boch. As you can see, there are comic motifs on the decorative tiles.

Figure 4.2 shows both versions of one of the dialogues that was generated in class **Also**: (4.3) makes use of the dialogue history, while (4.4) does not. The primary dialogue-history-based difference between the two versions is highlighted. All of the other surface differences occur because they were independently generated by the full COMIC presentation planner, which, as described in Section 3.2.3, incorporates some random choice.

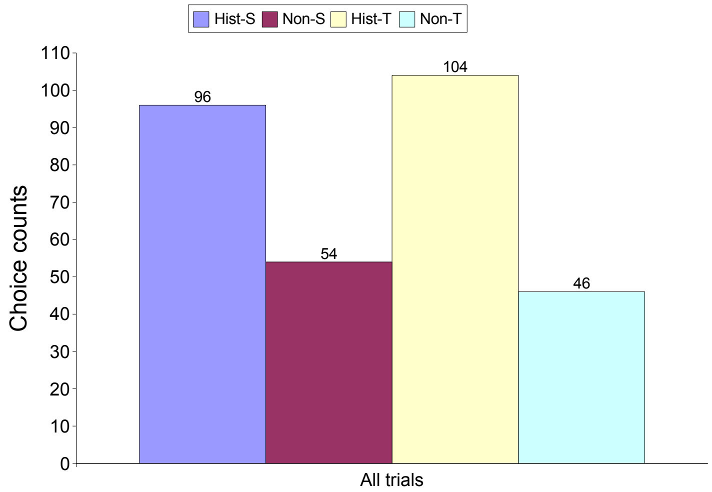
4.3.4 Results

For this experiment—like all of the other forced-choice studies in this thesis—we used a binomial test to measure whether participants' responses differ significantly from random choice. This test provides an exact measure of the statistical significance of deviations from a theoretically expected classification into two categories. In a binary forced-choice test such as this, the null hypothesis is that participants selected randomly between the two versions of the output, so the expected probability of each option is 0.5. To assess the significance level, we then consult the value of the binomial distribution for this probability and the observed counts.

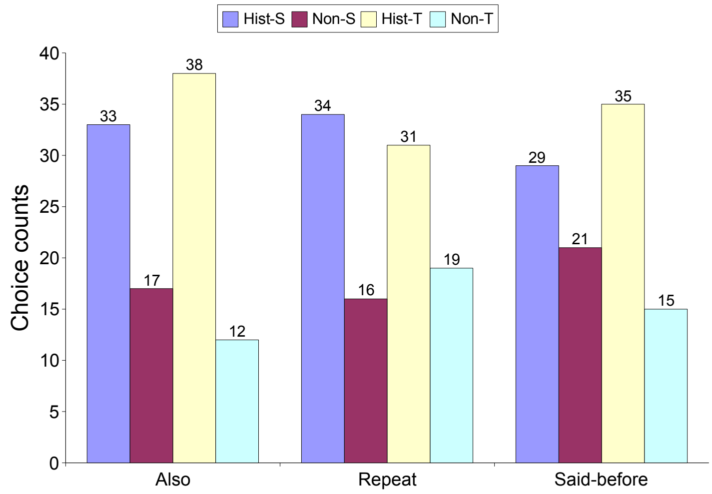
The overall results are shown in Figure 4.3(a). The first pair of bars represents the number of times participants selected the tailored and non-tailored versions, respectively, in speech, while the second pair shows the same choices in the text modality. In general, the participants chose the versions generated with the dialogue history enabled more often than those with it disabled in both text and speech presentation modalities (69% and 74% of the time, respectively); both of these differences are significant at the $p < 0.001$ level on a binomial test. There was little overall difference between the choices made in the two modalities ($\chi^2 = 0.735$; $df = 1$; $p = 0.39$).

We can also consider the responses on each individual dialogue class separately; these results are presented in Figure 4.3(b). The trends are the same for all of the dialogue classes: in every

Figure 4.3: Graphs of dialogue-history results



(a) Overall choice counts



(b) Choice counts by dialogue class

case, there is a tendency to choose the tailored version as having a more natural sequence of turns, although (due to the smaller sample sizes) not all of these preferences are significant. There is no significant difference between the response patterns on the different classes ($\chi^2 = 3.54$; $df = 6$; $p = 0.74$). None of the demographic features had an effect on the responses.

4.3.5 Discussion

The results of this experiment show that dialogue-history tailoring does have a perceptible effect in all cases, with both presentation modalities, confirming both of our hypotheses. This both confirms and refines the results of the M-PIRO evaluation described by Karasimos and Isard (2004), in which participants used output presented as text only, and in which both aggregation (combining facts to create more complex sentences) and comparison to previously-described objects were varied together. Participants in that study preferred the texts generated with the added generation techniques, and also showed higher learning outcomes. Our experiment confirms that this type of dialogue-history adaptation is also perceptible in the COMIC domain, and also shows that this effect holds when the generated text is presented as speech in a multimodal dialogue system.

Although the presentation-modality hypothesis was generally confirmed, there was a trend in most cases for the preference to be more pronounced when the output was presented as text. If we examine the responses on the individual items, it appears that this effect is likely due to intelligibility issues with the synthesised speech on particular items. For example, on the particular **Also**-class dialogue shown in Figure 4.2, preferences were essentially at chance when participants chose on the basis of the speech (12–13; $p = 1$). However, when participants were able to read the text, there was a trend[3] in favour of the dialogue-history version (18–7; $p \simeq 0.04$). As highlighted in the figure, the primary difference between the two versions is the single use of the word *another*. Adaptations with more obvious surface impacts—e.g., avoiding repeating information, using *as I said before* to signal an unavoidable repetition, or describing multiple common properties of two designs—were perceived at a similar rate in text and speech.

The overall result of this first experiment is that participants generally perceived the dialogue-history tailoring and selected output that included such tailoring as having a more natural sequence of turns. The only exception was on specific items where the tailoring had a subtle effect on the surface: for such items, the tailoring was preferred when the output was presented as written text, but not when it was presented with synthesised speech.

[3]When performing multiple tests on the same data in this way, it is advisable to use a Bonferroni correction on the significance values to avoid finding a significant effect by chance. The necessary Bonferroni adjusted value for $p < 0.05$ overall significance is $p < 0.0083$ on each of the six individual instances.

4.4 Human evaluation: Perception of user-preference tailoring

The results of the evaluation described in the preceding section demonstrate that the dialogue-history tailoring was appreciated. In this section, we describe a second experiment designed to measure whether the user-model tailoring is similarly perceptible and preferred. In this case, we asked the human judges to select between output tailored to a target user model and output tailored to the preferences of some other user, again based first on a multimodal presentation and then on the textual transcripts.

Several previous studies have investigated the impact of user-preference tailoring on generated output. Carenini and Moore (2006), for example, performed a task-based evaluation of the output generated by the GEA system, which produced user-tailored descriptions of real estate and found that user-tailored descriptions were significantly more effective at persuading users to consider options than were non-tailored descriptions. As mentioned earlier, the evaluation of user-tailored generation in the MATCH system (Walker *et al.*, 2004) mentioned earlier used a similar "overhearer" setting to the current study. The results of that evaluation indicated that users preferred the output tailored to their own model to output tailored to the preferences of some other user.

Based on those studies and on the results of the previous study, the hypotheses for this experiment were similar to those in the preceding section:

1. Participants will select the version tailored to the target model more often than they will select the version tailored to the other model.

2. There will be no difference in the pattern of responses between the two presentation modalities.

4.4.1 Participants

This experiment was run consecutively with the experiment described in the preceding section, and the same set of participants took part. Due to technical difficulties, two of the participants from the previous study were unable to complete this experiment, so the details of the participants for this experiment are as follows:

Total number	23		
Gender	Female: 5	Male: 18	
Age	Under 20: 6	20–29: 13	30 and over: 4
Computer experience	Beginner: 0	Middle: 3	Expert: 20
Native language	English: 17	Other: 6	

4.4.2 Methodology

This experiment was run using the same test harness as the experiment described in the preceding section, and the interface was very similar. After finishing the first experiment, participants were presented with a window describing the details of the second experiment. Once a participant had read the new set of instructions, they pressed a button to begin the experiment. The experiment interface is described in Appendix C.1.3.

For this experiment, an additional small window was added to the interface showing the known likes and dislikes of the user being observed, in a format similar to Figure 4.4. Participants were told to read through the preferences before beginning the experiment, and to keep the preferences in mind when making their responses; the window was shown on screen at all times. The participants were then shown a short dialogue consisting of eight user requests and system responses. There were two possible versions of each system response, one of which was tailored to the given user preferences and one to the preferences of some other hypothetical user. The participants were instructed to watch and listen to both responses and then to answer the following question: *Which COMIC output was more appropriate for this user?* As in the previous experiment, the choice was forced; also as in the previous experiment, participants chose first on the basis of the spoken multimodal presentation, and then chose again immediately based on reading a transcript of the speech.

There were four different possible user models, and participants were assigned to the models in rotation. Across the eight system turns, one of the possibilities was always tailored to the target model; the other was tailored either to one of the other models, or to a neutral model. Each model other than the target was used as the non-target model for two of the turns, with the assignment of models to turns made randomly for each participant. The order of presentation within a turn was also balanced so that output tailored to each non-target model was the first option in four of the turns and the second option in the other four.

4.4.3 Materials

To generate the materials for this experiment, four random user models were first created. Each model was generated by selecting three feature values that the user liked—which were given an evaluation of 0.8—and five values that the user disliked—which were given an evaluation of 0.2.[4] All other values were given the default evaluation of 0.5, and the feature weights were equal in all four user models. One of the generated models is shown in Figure 4.4.

For each model, an individual dialogue with the system was then created, manually providing user requests to the full COMIC decision-making and output system. Each dialogue started

[4]We selected a greater number of dislikes because there is a wider range of output options for negative features.

Figure 4.4: Sample user model

Feature	**Likes**	**Dislikes**
Colour	blue, beige	pink
Style		modern, classic
Decoration	floral motifs	geometric shapes
Designer		Porcelaingres

Figure 4.5: User request and system output generated for three different user models

(4.5) **User:** Tell me about this design *[click on Alt Mettlach]*

(4.6) **COMIC:**

a. *(target)* Here is a family design. As you can see, the tiles have a blue and green colour scheme. It has floral motifs and artwork on the decorative tiles.

b. *(non-target)* Here is a family design. Its tiles are from the Lollipop collection by Agrob Buchtal. Although the tiles have a blue colour scheme, it does also feature green.

c. *(neutral)* This design is in the family style. It uses tiles from Agrob Buchtal's Lollipop series. There are artwork and floral motifs on the decorative tiles.

with the system's default selection of designs and consisted of eight user requests and system responses. The user requests were selected to be plausible for the target user model, and the dialogue manager also made its choices based on that target model.

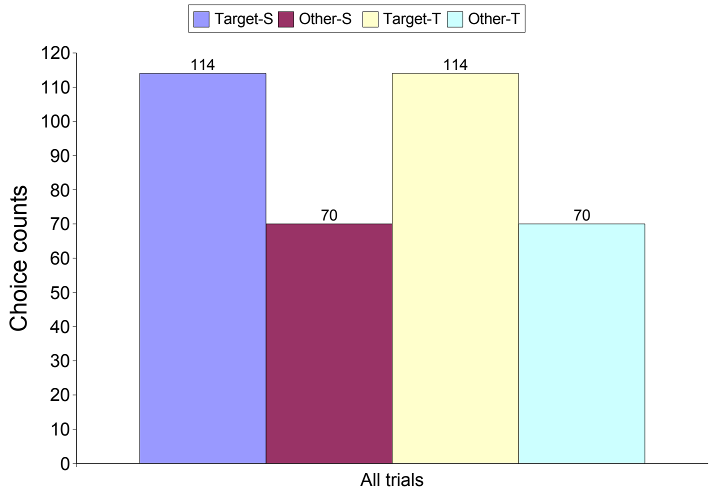
A total of four additional versions were then generated of each system output in each dialogue: one version based on the preferences of each of the other three random models, as well as a version based on a neutral user model (all evaluations 0.5, as in the dialogue-history study). Figure 4.5 shows three versions of a user request and system response: (4.6a) is generated based on the user model in Figure 4.4, (4.6b) reflects the preferences of one of the other user models, while (4.6c) is based on a neutral model. The highlighting in the figure indicates the primary user-preference-driven difference between the outputs; we discuss these differences in more detail in Section 4.4.4.

4.4.4 Results

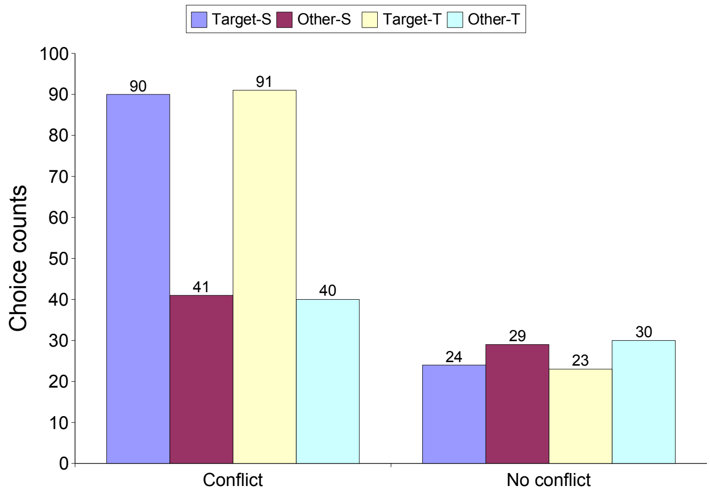
The overall results of this experiment are shown in Figure 4.6(a); again, the first two bars indicate the results for the speech modality, and the second two the results for text. These results are similar to those of the dialogue-history evaluation: participants generally chose the presentation generated for the target model over the one generated for the other model, using both presentation modalities. Using either modality, participants selected the correctly tailored output 62% of the time, which is significant at $p < 0.005$ on a binomial test.

In the MATCH evaluation (Walker *et al.*, 2004), the trials were subdivided based on the distance between the target model and the other model. However, the distance metric used there

Figure 4.6: Graphs of user-model results



(a) Overall choice counts



(b) Choice counts by trial type

was based on the difference between the feature weights and is therefore not applicable to the current study in which the weights were the same in all the models. Instead, we divided the trials based on conflicting use of explicit concessions to negative preferences in the two descriptions. For example, in Figure 4.5, the highlighted sentence in (4.6b) has the concession *although the tiles have a blue colour scheme*, as if the user expressed dislike for the colour blue; in contrast, the corresponding highlighted sentence in (4.6a) has no such concession. This resulted in the following two classes of trials:

Conflict There was at least one conflicting concession across the two versions—for example, a trial comparing (4.6a) and (4.6b) would fall into this class.

No conflict There were no concessions at all, or the concessions were the same in both versions. A trial comparing (4.6a) and (4.6c) would be in this class.

Due to the way the user models were generated, with more negative preferences than positive, the majority of trials fell into the **Conflict** class.

Figure 4.6(b) shows the response counts for each of these classes, again with the results for speech first and those for text second. There is a significant difference between the responses in the two classes: $\chi^2 = 19.66$, $df = 3$, $p < 0.005$. If we examine the preferences in each class, there is a very significant preference for the target description in the **Conflict** class (69% and 70% for text and speech, respectively; both $p < 0.0001$), but there is no significant preference either way in the **No conflict** class (44% and 45%; $p > 0.4$). We also investigated whether there was a difference between cases where the “other” model was the neutral model and when it was one of the other generated models, but there was no difference between those cases except for that which was covered by the conflicting concessions.

4.4.5 Discussion

As with the previous experiment, both of our hypotheses are confirmed, and the results of this study both confirm and extend the results of similar evaluations. The overall results show that, when the user preferences are taken into account, the participants were good at distinguishing output tailored to a target model from output tailored to some other user’s preferences, with similar performance in both presentation modalities; this is in line with the results of other studies including those of Carenini and Moore (2006) and Walker *et al.* (2004).

The participants in this study could not tell the difference between a description intended to be positive and one intended to be neutral; the effect of the user-tailored generation was only noticeable when there were conflicting concessions to negative preferences in the two versions being compared. In the output planner, the positive preferences do have an effect on both the content that is selected for a description and the order in which it is presented: features that the user likes will always be included in the description, and are generally placed nearer the beginning (e.g., compare the content selection in descriptions (4.6a) and (4.6c) in Figure 4.5). However, these factors made no difference to the participants in this study.

There are several possible explanations for this result: one possibility is that this is a result of the participants basing their judgements on a random user model rather than on their own preferences. It is also possible that, due to the "sales" setting, the default mode for output in COMIC is positive, and that only marked departures from that default are perceptible. We will return to this issue in later chapters of this thesis.

As described in Section 3.1.4.1, the dialogue manager also consults the user preferences when selecting designs to show in response to a request. However, when we generated the materials for this study, the dialogue manager made its selections based on the target model, and the additional versions of each turn were generated using the same output specification but different user models. This choice was made so that content tailored to different models could be presented in context without disrupting the flow from turn to turn. Although the impact of the dialogue-manager selections was not studied here, the results of the MATCH evaluation (Walker *et al.*, 2004) suggest that user tailoring at this level would also be perceptible.

4.5 Summary

In this chapter, we described how user preferences are represented in COMIC and showed how information from that source and from the dialogue history is incorporated into the text-planning process to produce contextually tailored descriptions of tile designs. We then presented the results of two overhearer-style human evaluations in which participants made a forced choice between two possible versions of the system output in context: in each case, one version was correctly tailored to the context, while the other was not. Each study looked at the individual impact of one of the two contextual factors: the first addressed dialogue-history tailoring, while the second addressed user-model tailoring. In each study, the participants judged the output based first on the full speech-based multimodal presentation and then on the written transcript of the speech.

The overall results of the two studies are similar: in both cases, participants selected the correctly tailored version significantly more often than the other version, and the results were similar using both presentation modalities. These results confirm and extend the results of previous studies such as those of Karasimos and Isard (2004), Walker *et al.* (2004), and Carenini and Moore (2006).

While the high-level results of the two studies are similar, if we look at the results more closely, some differences emerge. In the dialogue-history study, there was a tendency for the tailored version to be selected more often with the textual presentation than it was with the speech presentation, especially when the difference between the two versions presented in a trial was subtle. This suggests that, while the dialogue-history tailoring always had an effect, in some cases users' ability to perceive this effect was diminished by the synthesised speech.

There was no such modality effect in the user-model study. However, participants in that study were only able to detect user-model tailoring on trials where the two descriptions contained conflicting concessions to negative preferences. These participants were unable to tell the difference between descriptions intended to be positive and those intended to be neutral. We propose two possible explanations for this result. One possibility is that it is a result of the participants making judgements based on a random user model rather than on their own preferences. It could also be the case that the default mode of the COMIC system is to be positive, and that only variations that are markedly different than this default are perceptible. We will return to this question in later chapters.

In the context of the thesis, these results directly address the first research question: determining when variation is perceptible to—and appreciated by—users of a system. In this study, the judges were generally able to perceive the tailoring in both studies, and also generally felt that the context-tailored version was more appropriate to the hypothetical system user. In other words, incorporating this type of rule-driven, context-based tailoring into a system is likely to increase the quality of the output. Note that there were some factors that lessened participants' ability to detect the tailoring: in the first study, the synthesised speech was an issue when the tailoring was subtle, while in the second study subjects only noticed the tailoring when either the target or the other description expressed a negative user-model evaluation.

These results also contribute to the goals of the thesis in another, more indirect way. In Section 3.3, we described how COMIC is similar to a number of other dialogue and generation systems, and claimed that this means that results from studies of COMIC can also generalise to these other systems. The results of both of the human evaluations described in this chapter are generally in line with the results of similar studies on other systems (Karasimos and Isard, 2004; Walker *et al.*, 2004; Carenini and Moore, 2006), which provides additional evidence that findings based on on COMIC also apply to other generation systems.

Chapter 5

Periphrastic variation in text*

> Those who cannot remember the past are condemned to repeat it.
>
> George Santayana, *Life of Reason*

In this chapter, as in the previous one, we concentrate on the textual descriptions generated by COMIC. In the previous chapter, we investigated rule-based variation introduced in the text- and sentence-planning stages; in this chapter, we focus on the opportunities for stochastic variation at the surface-realisation stage.

A classic rule of writing is to avoid repetition in order to keep text interesting and lively. When designing systems to automatically generate text, it is often assumed that this goal should be taken on board as well; for example, all of the systems described in Section 2.5.4 used this technique. However, the specific impact of avoiding repetition on judgements of text quality has not often been directly assessed: some evaluations have compared output generated with and without stochastic variation, but generally they have employed individual texts presented in isolation rather than discourses.

To address this issue, we implement two techniques for increasing the variability of the COMIC textual descriptions: choosing options with n-gram scores below the top, and penalising words from the preceding discourse. We evaluate the impact of these techniques in two ways. First, we measure human judgements of the understandability, repetitiveness, and overall quality of a set of tile-design description sequences generated with and without these techniques. We then perform a systematic exploration of the parameter space for both implementation techniques to assess their individual impact on various aspects of the generated description sequences.

*This chapter is based on Foster and White (2007).

Like the experiments in the preceding chapter, the results of this human-judgement experiment contribute to the first research question of the thesis: the human judgements of minimal pairs of texts generated with and without periphrastic variation adds to our knowledge of human judges' reactions to such variation in generated output. Comparing users' reactions to the texts generated by the modified system to those generated by the default system—which always selects the highest-probability option based on the corpus—also allows us to address the second question (comparing rule-based generation systems to those that vary stochastically). In addition to performing the human-judgement study, we also compute corpus-based and other automated measures on the texts. Comparing the results of these automated measures to the findings from the user study also provides the first indications of a response to the third research question—determining how human judgements of output quality relate to automated measures.

We first give details of the implementation of both of these anti-repetition techniques within the OpenCCG realisation process (Section 5.1). In Section 5.2, we then describe the methodology and results of a human evaluation in which judges compared the quality of minimal pairs of description sequences generated with and without the anti-repetition measures. Next, in Section 5.3 we describe the results of an exploration of the impact of the parameters in both anti-repetition measures, using three automated measures on the generated texts: the variability across descriptions, the average corpus-based n-gram scores, and the rate of dispreferred paraphrases. Finally, in Section 5.4, we discuss the results of both of these studies in the context of the topics of this thesis.

5.1 Implementation: Anti-repetition measures

As described in Section 3.1.5.2, the OpenCCG surface realiser uses a hybrid symbolic-statistical chart realisation algorithm to transform logical forms into text, incorporating language models for making choices among the options left open by the grammar. We implemented two techniques for avoiding repetition in the generated text through periphrastic variation, both of which modify the basic OpenCCG realisation process. The first technique, and the simplest, is *ϵ-best sampling*: we perform the normal realisation process, and then select randomly among those alternatives whose n-gram score is within a threshold ϵ of the top-scoring alternative. The second implementation technique is *anti-repetition scoring*: storing the words from previously-generated sentences and penalising any repeated words when scoring a proposed realisation for a new sentence.

The implementations take advantage of the recently-added integrated support for disjunctive logical forms in the OpenCCG realiser (White, 2006a). The starting point was the XSLT-based text planner described in Section 3.2. The text-planning process was enhanced to generate disjunctive logical forms and to cover the full range of paraphrases supported by the current COMIC grammar (which has been enhanced since the COMIC project finished).

In this section, we first give an overview of disjunctive logical forms and of how the text planner was extended to support them (Section 5.1.1). We then describe the two anti-repetition measures in detail and show how each was implemented using the enhanced text planner.

5.1.1 Disjunctive logical forms

Disjunctive logical forms are logical forms that indicate possible paraphrases by including explicit disjunctions and optional information. Disjunctive logical forms have the advantages over underspecified forms that they permit domain-specific, sentence-level paraphrases and that they enhance the reusability for grammars across applications. As an example, the logical form excerpt in Figure 5.1 corresponds to the introduction of a tile design in the country style. The `one-of` tags indicate disjunction, while the set of nodes at the bottom are common to more than one option within the disjunctions; it is also possible to include optional information in a logical form via an `opt` tag. The paraphrases produced by the sample logical form include *This design is in the country style*, *This one is country*, and many other possibilities.

White (2006a) describes how the OpenCCG realisation algorithm was extended to efficiently generate paraphrases from such disjunctive forms; the following is a high-level summary. First, the input logical form is flattened to a set of elementary predications (EPs), each corresponding to a single lexical predication, semantic feature, or dependency relation. Constraints are added to ensure that only combinations of EPs that are licensed by the original logical form can be produced in the final output. These constraints are used to implement a revised version of *edge coverage*: testing whether a proposed realisation covers exactly the specification in the input logical form. White also demonstrated that this integrated treatment of paraphrasing is more efficient than multiplying out the alternatives and successively realising each, as is done in the basic COMIC generation process outlined in Section 3.2.3.

To create the materials for these experiments, the sentence planner from Section 3.2.3 was modified in two ways. First, the range of possible paraphrases was extended to cover the new syntactic structures added to the COMIC grammar since the end of the project. Second, the text-planning XSLT templates were modified to create the new disjunctive syntax required by the updated version of OpenCCG.

Figure 5.1: Disjunctive logical form

```
<lf>
  <one-of>
    <!-- Sentences with main verb "is" -->
    <node id="b1:state" pred="be" info="rh" mood="dcl" tense="pres">
      <!-- Subject -->
      <rel name="Arg">
        <node id="d1:mental-obj" num="sg">
          <one-of>
            <!-- This design -->
            <atts pred="design" kon="+">
              <rel name="Det">
                <node idref="t1:proposition" shared="true"/>
              </rel>
            </atts>
            <!-- This one -->
            <atts pred="pro_one" kon="-">
              <rel name="Det">
                <node idref="t1:proposition" shared="true"/>
              </rel>
            </atts>
            <!-- This -->
            <atts pred="this" kon="+"/>
          </one-of>
          <!-- Common node "this" (determiner) -->
          <node id="t1:proposition" pred="this" kon="+"/>
        </node>
      </rel>
      <!-- Property -->
      <rel name="Prop">
        <one-of>
          <!-- in the country style -->
          <node id="i1:proposition" pred="in">
            <rel name="Fig">
              <node idref="d1:mental-obj"/>
            </rel>
            <rel name="Ground">
              <node idref="s1:abstraction" shared="true"/>
            </rel>
          </node>
          <!-- country -->
          <node id="c2:style" pred="country" kon="+">
            <rel name="Of">
              <node idref="d1:mental-obj"/>
            </rel>
          </node>
        </one-of>
      </rel>
    </node>
    <!-- Also "here we have ..." -->
  </one-of>
  <!-- Common nodes, referenced above -->
  <!-- "country" -->
  <node id="c1:style" pred="country" kon="+"/>
  <!-- "the country style" -->
  <node id="s1:abstraction" pred="style" det="the" num="sg">
    <rel name="HasProp">
      <node idref="c1:style" shared="true"/>
    </rel>
  </node>
</lf>
```

5.1.2 ϵ-best sampling

The simpler of the two anti-repetition measures is *ϵ-best sampling*: running the full realisation process and then choosing from among the high-scoring realisations rather than selecting the single highest-scoring possibility. This implementation is similar in spirit to Belz and Reiter's (2006) *greedy roulette p*CRU implementation of surface realisation (Section 2.2.1) and to Stone *et al.*'s (2004) technique of sampling from low-cost utterances when selecting behaviours for an embodied agent (Section 2.5.4). Both of these implementations also perturbed the selection strategy of a stochastic generation system to cause the generator to explore more of the possibilities than a Viterbi-style single-best search could reach.

As described in Section 3.1.5.2, the OpenCCG realiser has a chart-based realisation algorithm that integrates n-gram scoring of possible realisations. The n-gram models in COMIC are based on a testbed of 549 target sentences covering the range of sentence structures produced by the grammar included in the final integrated system. In its default mode, this algorithm always returns the realisation with the highest corpus-based n-gram score.

ϵ-best sampling was implemented by having the realiser create the full set of possible surface realisations for a given disjunctive logical form and then randomly selecting one from that set whose n-gram score is within a given threshold of the top score. As the scores for sentences can vary by many orders of magnitude, the threshold is specified as a distance in log-10 space.

Depending on the threshold value chosen, this process can add more or less variation to the output. However, there is no guarantee that the system will not continue to repeat itself (although the probability is lower), and if the threshold is too large and the grammar overgenerates, the selection may include paraphrases that are dispreferred, or even ungrammatical.

5.1.3 Anti-repetition scoring

ϵ-best sampling adds variation to the output by incorporating a degree of random choice into the output process. In this section, we describe a more deterministic technique for ensuring variation: storing the preceding discourse and modifying OpenCCG's n-gram scoring process to penalise possible realisations that repeat structures that were used earlier. A similar mechanism was implemented in the M-PIRO personalisation server (Androtsopoulous *et al.*, 2007) to avoid repeating expressions over the course of an interaction.

OpenCCG provides a built-in mechanism for implementing such caching models and integrating them into its normal n-gram-driven search for good realisations. Isard *et al.* (2006) used a similar mechanism to implement alignment (Pickering and Garrod, 2004) in dialogues between two simulated agents; in that case, they stored the preceding discourse and used that information to boost the scores of possible outputs that were similar to what came before. In our implementation, we used the same mechanism, but penalise the preceding context rather than using it to boost the scores.

The anti-repetition scorer was implemented as follows. First, for a proposed realisation, the count c of open-class words that occurred in the preceding discourse was computed. This count was weighted by the distance in the history: a word that appeared in the immediately preceding context received a full count of 1, one that appeared only in the context before that received a count of 0.5, one from further back was weighted at 0.25, and so on. The repetition score r for a proposed realisation was then 10^{-p*c}, where p was the specific penalty value. Note that this formula returns 1 if there are no repeated items, and a score that is linear in log space with the number of repeated items otherwise. The overall score for the proposed realisation was computed by multiplying r by the normal corpus-based n-gram score. In this way, preferences regarding word order and function words are still determined by the n-gram model, since the anti-repetition scorer only pays attention to single open-class words.

5.2 Human evaluation: Preferences for periphrastic variation

As described in Section 2.5.4, several previous systems (e.g., Stone *et al.*, 2004; van Deemter *et al.*, 2005; Androtsopoulous *et al.*, 2007) have incorporated random choice into the generation process; however, there was never any evaluation of the impact of that design choice on the quality of the generated output. Stent *et al.* (2005) and Belz and Reiter (2006) have done evaluations in which participants preferred texts generated by a system that included stochastic choice; however, those evaluations presented the texts one at a time, in isolation, and so did not directly test the impact of repetitiveness.

In this study, we directly addressed the issue of avoiding repetitiveness in the output by gathering human judgements on short description sequences, presented as minimal pairs: one realised by a baseline system, and one incorporating anti-repetition measures. Participants made forced choices between the two versions on three factors: understandability, repetitiveness, and the quality of the writing. Presenting a series of descriptions allows us to measure the impact of anti-repetition measures in discourse, rather than in isolation, giving finer-grained judgements on the impact of this type of variation.

None of the previous studies addressed precisely the question that we evaluate in this study. However, based on the results of similar studies—and our own intuitions—we proposed the following hypothesis for this study:

1. Participants will significantly prefer the description sequences generated with the anti-repetition measures enabled.

5.2.1 Participants

This experiment was run over the world-wide web. Participants were recruited via an email to the Edinburgh University Informatics departmental student mailing list, and by posting the experiment on the Language Experiments Portal,[1] a website devoted to online psycholinguistic experiments. All participants in the study were entered into a draw for a gift certificate. The details of the participants were as follows:

Total number	37		
Gender	Female: 20	Male: 17	
Age	Under 20: 2	20–29: 19	30 and over: 16
Computer experience	Beginner: 0	Middle: 15	Expert: 22
Native language	English: 37	Other: 0	

5.2.2 Methodology

Participants were presented with a series of eight pairs of four-item description sequences generated by the COMIC text planner—one set realised with OpenCCG running in its default mode (choosing the highest-probability option in all cases, with no penalty for repeated items) and one with both of the anti-repetition measures enabled. For each sequence pair, a participant was asked to make three forced choices: which of the two versions was (1) more understandable, (2) more repetitive, and (3) better written. Once a participant had made selections for one pair of sequences, the next pair were displayed. The interface is presented in detail in Appendix C.2.

All participants saw the same eight description sequences, each in an individually-chosen random order. The order of presentation was balanced so that each participant saw the anti-repetition version as sequence 1 in four of the trials and as sequence 2 in the other four; the allocation of orders to the items was also randomly chosen for each participant.

5.2.3 Materials

To create the materials for this study, we used the enhanced text planner described in Section 5.1.1 to create disjunctive logical forms for a set of eight description sequences. Each sequence consisted of four consecutive descriptions of tile patterns, with a different set of randomly-chosen designs from the COMIC ontology used for each. Every logical form was created using a fixed set of rules for content selection and structuring. There was no aggregation

[1] http://www.language-experiments.org/

Figure 5.2: Sample description sequence realised in both modes

Default		**Anti-repetition**
This design is country. It is based on the Sandstein collection by Porcelaingres. The colours are brown, grey and black. There are geometric shapes on the decorative tiles.		Here is a design in the country style. It uses tiles from the Sandstein collection by Porcelaingres. It has brown, grey and black in the colour scheme. The decorative tiles have geometric shapes.
This design is also country. It is based on the Cardiff collection by Aparici. The colours are cream and dark red. It also has geometric shapes on the decorative tiles.		This one is also country. It draws from Cardiff, by Aparici. The colour scheme features cream and dark red. The decorative tiles also have geometric shapes.

or user-preference tailoring, and the features were always presented with style first, manufacturer and series second, colour third, and decoration last. Later descriptions in a sequence took into account the information from the preceding descriptions to include words such as *also* and *another*, but there was no other tailoring enabled.

We then used the OpenCCG realiser to generate text from these disjunctive logical forms in two ways: in the default mode—with no anti-repetition measures enabled—and with both ϵ-best sampling and anti-repetition scoring enabled. We used a value of 20 for both the threshold in ϵ-best sampling and the penalty in the anti-repetition scorer. Figure 5.2 shows the first two sentences of one of the description sequences, realised with the realiser with the realiser running in both modes. For the anti-repetition scorer, each description as a whole provided the context for the sentences in the next: that is, the realiser was run on the logical forms for an entire set of sentences, and then all of the sentences were added to the context before the next description was processed.

5.2.4 Results

The overall results are presented in Figure 5.3; as one participant gave responses for only half of the questions, there were a total of 292 responses to each question from the 37 participants in this experiment.

For the understandability question, participants chose the anti-repetition version in 157 cases (54%); this preference was not significant on a binomial test ($p \approx 0.2$). However, the responses to the other two questions did show significant trends. Participants chose the default version as more repetitive 210 times (72%) and the anti-repetition version as better written 180 times (62%); both of these preferences are significant at $p < 0.0001$ on a binomial test.

Figure 5.3: Overall results of the human evaluation

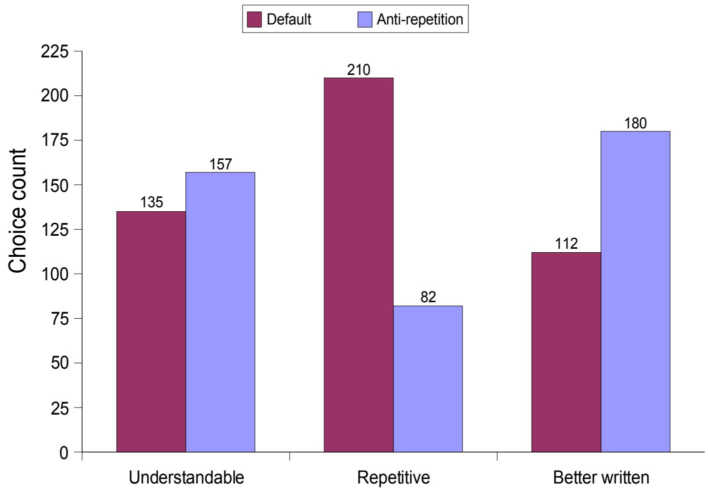
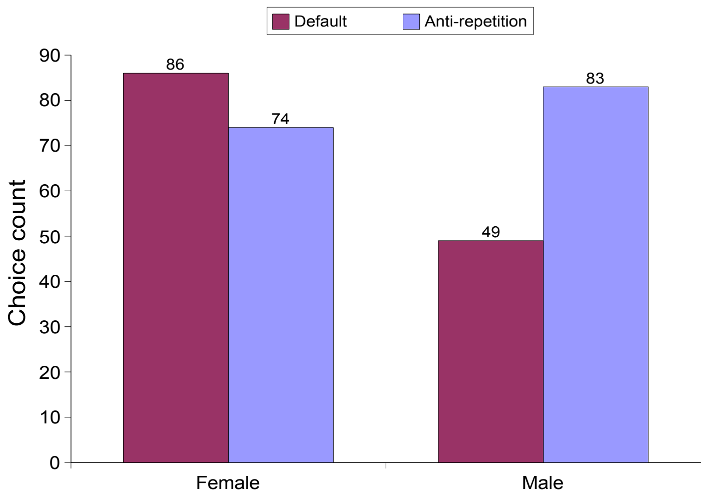


Figure 5.4: Responses on the understandability question, divided by gender



If we consider the demographic factors, only one had any significant effect on the responses: male and female participants' responses to the understandability question were significantly different ($\chi^2 = 7.39$; df = 1; $p < 0.01$). Indeed, if we perform a binomial test on just the male participants' responses to this question, they selected the anti-repetition version significantly more often than the default (63%; $p < 0.005$), while the female subjects actually mildly preferred the versions generated in default mode. There were no other significant effects of demographic factors on any of the responses.

There was also a significant difference in participants' responses on one of the eight description sequences. Figure 5.6 shows participants' responses on item #1 compared to the overall responses. For this item (shown in Figure 5.5), 26 of 37 participants judged the description generated in default mode as being better written, and 18 out of 37 chose the version generated with the anti-repetition measures as being more repetitive. If we use a χ^2 test to examine the differences between this item and the rest on these two questions, they are both significant:

- Better written: $\chi^2 = 12.5$, $df = 1$, $p < 0.0001$
- Repetitive: $\chi^2 = 5.63$, $df = 1$, $p < 0.05$

There was no significant difference on the pattern of responses for the understandability question on this item. We will return to the features of this description sequence in Section 5.4.

5.2.5 Discussion

The results of this study indicate that the participants were sensitive to the impact of the anti-repetition measures. Participants judged sequences realised with the measures enabled to be better written and less repetitive, and to be equally understandable; the male participants actually found the anti-repetition texts to be more understandable. This confirms and extends the results of previous studies looking at this type of variation. As in the studies of Stent *et al.* (2005) and Belz and Reiter (2006), our participants also preferred the output of a system that takes steps to avoid repetitiveness; however, since our evaluation materials were short discourses, rather than isolated texts, we can hypothesise that repetitiveness, rather than inherent text quality, was a factor in participants' judgements.

Figure 5.5: Description sequence #1

Default

- This design is modern. It is based on the Carioca collection by Aparici. The colours are orange, red and ochre. There are geometric shapes on the decorative tiles.
- Here we have a design in the family style. It is based on the Girotondo Tile collection by Alessi Tiles. It features white in the colour scheme. There are comic motifs on the decorative tiles.
- This design is classic. It is based on the Ateliers collection by NovaBell. The colours are grey, green and rose. There are floral motifs and geometric shapes on the decorative tiles.
- Here we have a design in the family style. It is based on the Creative System Amazonas collection by Villeroy and Boch. The colours are green, yellow and red. There are animal motifs on the decorative tiles.

Anti-repetition

- Here we have a modern design. Its tiles are from the Carioca collection by Aparici. The design features orange, red and ochre. the decorative tiles have geometric shapes.
- This one is family. It draws from Girotondo Tile, by Alessi Tiles. It features white in the colour scheme. The decorative tiles have comic motifs.
- This one is classic. It's based on Ateliers, by NovaBell. It features grey, green and rose. There are floral motifs and geometric shapes on the decorative tiles.
- Here is a family design. It draws from Villeroy and Boch's Creative System Amazonas collection. The design features green, yellow and red. The decorative tiles feature animal motifs.

Figure 5.6: Tables for χ^2 comparisons on description sequence #1

	Understandable		Repetitive		Better written	
	Default	**Anti-rep**	**Default**	**Anti-rep**	**Default**	**Anti-rep**
Overall	135	157	210	82	112	180
Item #1	14	23	19	18	26	11

5.3 Automated evaluation: Variability and *n*-grams

The results of the preceding experiment show that human judges find text generated with anti-repetition measures equally understandable, less repetitive and better written than text generated without such measures. Both of the measures that were implemented depend on the setting of a parameter: the threshold value for ϵ-best sampling, and the repetition penalty for anti-repetition scoring. Each technique can be implemented on its own, or they can be used in combination.

In the human evaluation described above, both measures were enabled, and the parameter for each was set to the relatively large value of 20. In this section, we explore the relative impact of these two techniques on the generated text by running the realiser on a large set of disjunctive logical forms under a range of settings, and then computing several automated metrics of variability and text quality on the resulting text. We can also look at the scores of the materials that were used in the human evaluation to see which of the automated measures agree with the users' preferences.

5.3.1 Methodology

For this experiment, we generated the disjunctive logical forms for a set of 20 tile-design description sequences, each based on a different randomly-chosen set of four designs from the COMIC ontology. The logical forms were created with the same fixed rules for content selection and structuring as in the preceding experiment (Section 5.2.3).

For each sequence, we then ran the realiser on all of its logical forms in turn, using all combinations of the following values for both of the parameters: 0, 1, 5, 10, and 20. A threshold of 0 meant that the realiser chose the single highest-scoring result, while a repetition penalty of 0 amounts to no repetition penalty at all. When both parameters are set to 0, this corresponds to the default sentences from the human evaluation; when both parameters are 20, this corresponds to the anti-repetition sentences from that study. As in the previous study, we realised all of the sentences for a given description and then added the results to the context for the anti-repetition scorer for the next descriptions. To compensate for any variability introduced into the process by the random choice in ϵ-best sampling, we realised the whole set of sequences six times.

For each sequence realised under each combination of parameter settings, we computed the following metrics on the generated texts: the variability across the sentences in the description, the average *n*-gram score, and the incidence of dispreferred substrings. These three metrics give different perspectives on the text quality. The first metric assesses the degree of variation in the texts, regardless of corpus score; the second compares the texts to the corpus examples, penalising all deviations from the corpus equally; while the third counts the occurrence of specific deviations from the corpus that were deliberately not included in the language models. The following sections give more detail about each of these metrics.

5.3.1.1 Variability

To assess the variability in a generated description sequence, we computed the string edit distance between all pairs of descriptions in the sequence; that is, the number of insertions, deletions, and replacements required to transform one description into another. Guégan and Hernandez (2006) used a similar edit-distance-based metric to detect parallelism in texts. The score for a sequence was the mean edit distance between all pairs of descriptions in the sequence, where a higher score indicates greater variability. To give a concrete example, the edit distance between the two default-mode sentences in Figure 5.2 is 10, while the distance between the anti-repetition sequences is 24. The mean edit distance for the sequences used in the human evaluation described in the preceding section was 13.4 for the default versions and 23.1 for the anti-repetition versions.

5.3.1.2 *n*-gram scores

The variability metric evaluates the generated texts in isolation; it does not make any use of the corpus data. In contrast, this metric compares the generated descriptions against the corpus by computing *n*-gram scores based on the language models that were used in the generation process. The score for a description is the mean score of the sentences that make up that description. Since *n*-gram scores can vary by many orders of magnitude, we take the negative base-10 log of the score. The score of a description is then the arithmetic mean of the scores of each of the sentences in it. A lower score on this measure indicates that the sentences in the generated description are closer to the examples in the corpus.

Note that the COMIC grammar generates prosodic specifications (pitch accents and boundary tones) in addition to text, and this prosodic information is also included in the language model. The pitch accents are included as integral parts of words and are chosen using deterministic rules based on the information structure of the sentence. However, the boundary tones are treated as additional words. This means that, for a given word sequence, different boundary-tone sequences can be generated, only some of which agree with the examples in the corpus. This has a strong impact on the *n*-gram scores. For example, the only difference between the following two examples is the *LL%* boundary between *motifs* and *on*:

(5.1) There are floral_H* motifs_H* LL% on the decorative_H* tiles LL% .

(5.2) There are floral_H* motifs_H* on the decorative_H* tiles LL% .

The corpus sentences with this structure all have the boundary tone in the middle, so any sentence with that tone has a much higher n-gram score than one without it. In the example, the score of (5.1) is 1.6×10^{-3}, while (5.2) gets a score of 3.7×10^{-10}. Since the values can vary so widely, we compare the n-gram scores in negative log space.

Since the generated text was presented without prosody to the experimental participants in the study described in the preceding section, we compute n-gram scores for the generated descriptions in two ways: using the actual prosody that was selected, and using the highest-probability prosody for the word sequence. The former score corresponds more closely to the selection strategy that was used to create the sentences, while the latter score gives a better indication of the corpus similarity of the materials that were presented to the participants.

The generated prosody is not available for the texts used in the human evaluation, so we can only give numbers for the highest-probability prosody. Using this method, the mean negative log score was 3.00 (i.e., n-gram scores around 0.001) for the default descriptions and 7.37 for the descriptions generated with the anti-repetition techniques.

5.3.1.3 Dispreferred paraphrases

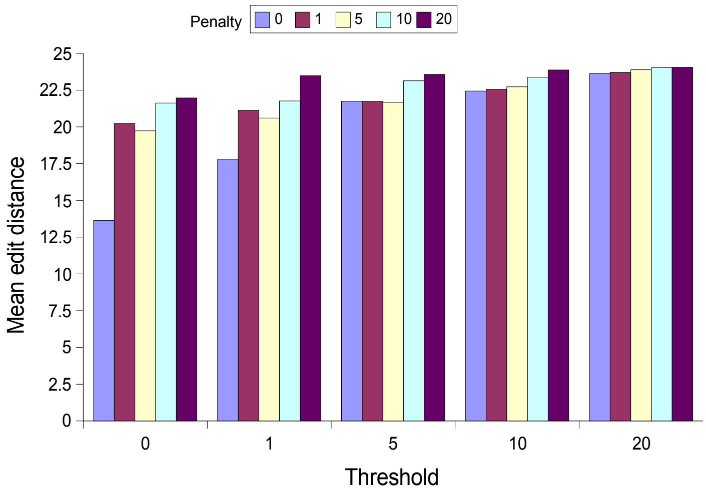
The n-gram-based measure described above computes the difference between the generated texts and the corpus examples, penalising all departures from the corpus equally. However, not all differences from the corpus are equally bad in practice. Some occur simply because a particular word sequence happens not to be in the corpus, which was created midway through the COMIC project and does not cover all of the paraphrases in the full grammar. However, other paraphrases are not in the language model deliberately: word sequences that are permitted by the grammar but that we rely on the n-grams to filter out during the realisation process. One of the features of OpenCCG's use of language models is that it allows developers to write a mildly overgenerating grammar and to constrain its outputs through the models. When the realisation process is modified, these sequences may no longer be filtered out.

For this measure, we search for a set of specific word sequences that are permitted by the COMIC grammar, but that were deliberately not included in the n-gram models used by OpenCCG to avoid their being generated. The occurrences of the following substrings were counted in the generated texts:

- Sentence-initial and sentence-final *also*
- *we here have* ... and *we have* ... *here* (instead of *here we have*)
- *is family* (instead of *is in the family style*)[2]

[2]Unlike *classic style*, *family style* is actually a noun-noun compound, but is not modelled that way for uniformity. The grammar therefore also permits *is family*, which is odd, so n-grams were used to avoid this wording.

Figure 5.7: Mean edit distances



In the anti-repetition descriptions used in the human evaluation in the preceding section, there was one instance each of *is family* and sentence-initial *also*.

5.3.2 Results

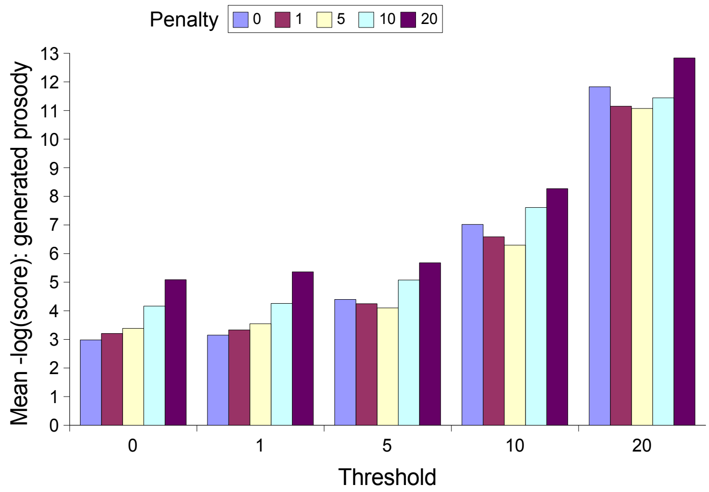
This section describes the results of each of the measures outlined above on the generated descriptions. Except where indicated, the results describe the data from all 20 of the description sequences, averaged over the six runs: a total of 12,000 generated descriptions arranged in 3000 sequences, 120 sequences with each of the 25 possible parameter combinations.

5.3.2.1 Variability

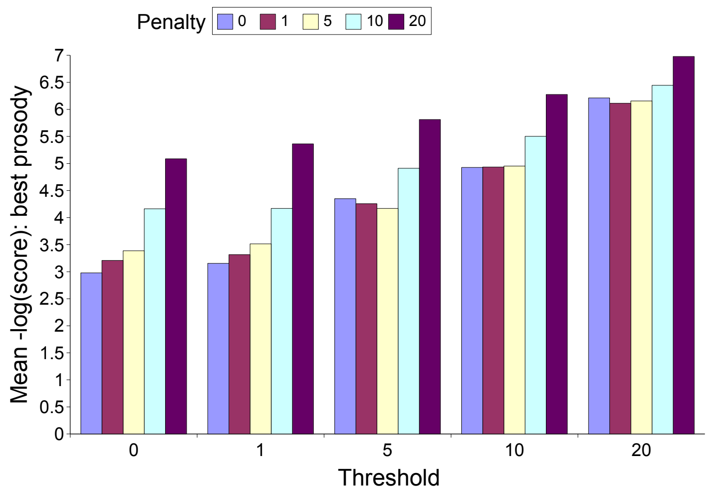
Figure 5.7 shows the mean sequence-level edit distance for all settings of the parameters across all of the runs: this value reflects the mean edit distance between all pairs of descriptions in a sequence. Across the x axis, each set of five bars corresponds to a different setting of the ϵ-best threshold parameter; within a set of bars, each shows the result for that threshold setting with a different value for the anti-repetition scorer penalty.

To assess the significance of these results, we performed a linear regression, treating the values of each parameter as levels of an ordered factor. The resulting model explained approximately 70% of the total variance ($R^2 = 0.71$). Using Helmert contrasts (i.e., comparing the value for each level of a factor with the mean of the values from the preceding levels), the regression coefficients for each of the individual factors (threshold and penalty) were both significantly

Figure 5.8: Mean *n*-gram scores (negative logarithm)



(a) Actual prosody



(b) Highest-scoring prosody

greater than 0 ($p < 0.0001$), showing that an increase in either tended to result in a corresponding increase in the edit distance. However, the regression coefficient for the interaction of the factors is negative (also $p < 0.0001$), indicating that the effect of the two methods is not simply additive.

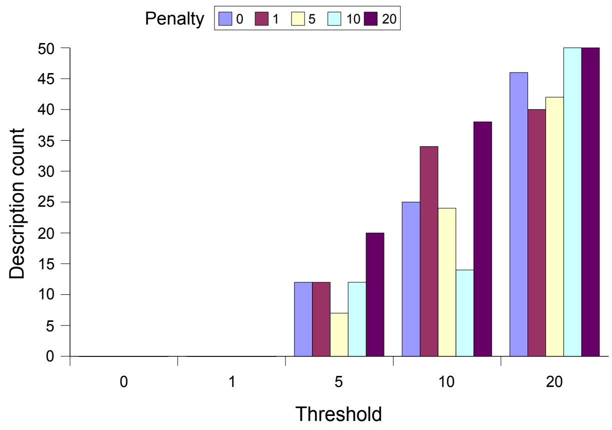
5.3.2.2 *n*-gram scores

Figure 5.8 shows the mean negative log n-gram score for the generated descriptions: Figure 5.8(a) shows the scores based on the actual prosody that was selected by the realiser,[3] while Figure 5.8(b) shows the n-gram score of the highest-scoring prosody for the words in each generated description. The groups of bars on these graphs are the same as those on Figure 5.7: each group of bars across the x axis corresponds to a different setting of the ϵ-best threshold, while each bar within a group corresponds to a different setting of the anti-repetition penalty.

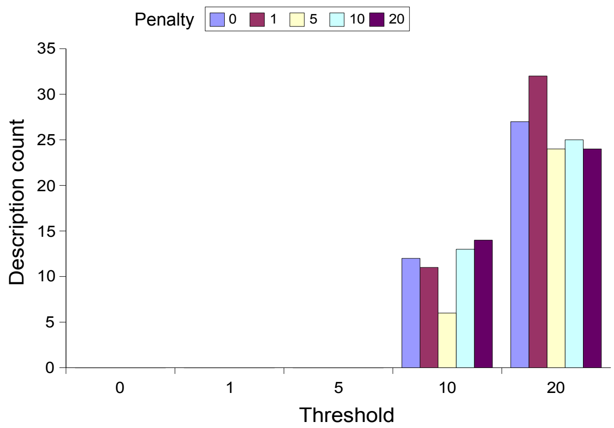
Just as in the previous section, we also used linear regression to determine the effect of each of the parameter settings on the two values, again treating each of the parameters as an ordered factor and using Helmert contrasts to determine the regression coefficients. For the actual generated n-gram scores (Figure 5.8(a)), R^2 for the regression model was 0.94, indicating that the model explains nearly all of the variance in the data. The regression coefficients in the model indicate that penalty values of 1 and 5 and a threshold value of 1 do not have any significant effect on the n-gram score ($p > 0.1$); this lack of effect can be seen on the graph. However, all of the parameter values above these levels did have a significant positive effect on the negative log n-gram score ($p < 0.0001$). The regression coefficients for threshold levels are much larger than those for the penalty levels (up to 1.4 for the threshold, compared to up to 0.3 for the penalty); this difference is also evident on the graph. Unlike the edit distance, for this measure there was little interaction between the influence of the two factors ($p < 0.001$ on ANOVA): both factors independently contribute to increasing the n-gram distance from the corpus examples, although the threshold has a greater contribution.

If we consider the n-gram scores for the highest-scoring prosody (Figure 5.8(b)), the pattern is similar, but the range of scores is less (the mean for (20,20) is 6.98, compared to 12.84 for the real prosody). R^2 for the regression model on this data is only 0.39, so the regression explains much less of the variance. Again, there was no impact on the score of the lower values of the repetition penalty (1 and 5); a threshold of 1 had a somewhat significant effect ($p < 0.005$), while all of the higher values had more significant effects ($p < 0.0001$). In this case, there was also a significant interaction of the two parameters (ANOVA, $p < 0.0001$): when both parameters are above 5, the regression coefficient is negative, indicating that the combined impact is not additive on this metric.

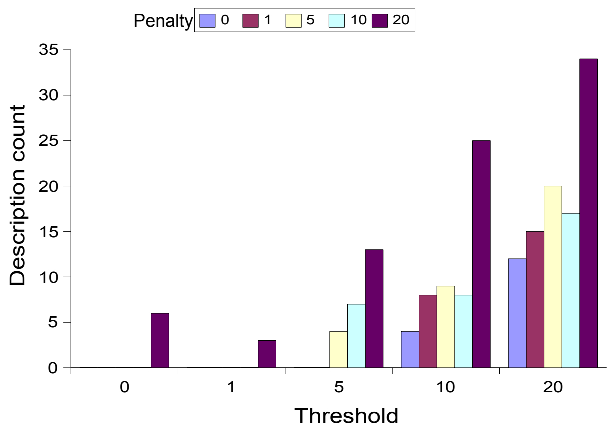
Figure 5.9: Counts for dispreferred paraphrases



(a) Sentence-initial and sentence-final *also*



(b) *we have . . . here* and *we here have*



(c) *is family*

5.3.2.3 Dispreferred paraphrases

Figure 5.9 shows the counts of dispreferred paraphrases under all of the parameter settings. The count for each setting indicates the number of descriptions that contained the specific substring. A total of 480 descriptions were generated under each combination of parameter settings: 20 sequences, each consisting of 4 descriptions, each generated 6 times. The count for each setting indicates the number of those descriptions that contained the specific substring. For example, 35 (7%) of the descriptions generated with both parameters set to 20 contained *is family* (Figure 5.9(c)). By inspection, it is clear that the dispreferred paraphrases tended to occur very infrequently at low parameter settings and to increase as the threshold increases; the penalty value appears to have had an effect only on the frequency of *is family*.

To assess the significant factors for each of the dispreferred paraphrases, we analysed the influence of both of the the parameters on the rate of that paraphrase by fitting a log-linear model to the contingency table of frequency counts for each of the paraphrase types; this type of model is suitable for use on count data and allows us to assess the influence of each of the factors on the counts in the table. The results indicate that the values that appear to be significant from the graph are indeed significant: increasing the threshold has a significant influence on the rate of all three of the paraphrases ($p < 0.0001$, ANOVA), while increasing the repetition penalty affects only the occurrence of *is family* (also $p < 0.0001$).

5.3.3 Discussion

We have computed the value of several automated metrics on description sequences generated under a range of settings of the parameter for both of the anti-repetition techniques. The results indicate that both the ϵ-best threshold and the anti-repetition penalty have an effect in most cases: increasing the value of either of the parameters both increases the variability of the generated text (as measured by edit distance) and decreases the corpus-based n-gram score, both the score based on the actual prosody that was selected and the score based on the highest-scoring prosody for the word string. The influence of both parameters on the edit distance is similar and largely redundant, while the ϵ-best threshold has a greater effect on the n-gram score than does increasing the repetition penalty.

Increasing the threshold also tends to increase the incidence of dispreferred paraphrases, particularly those that involve re-ordering the words in a sentence such as the placement of adverbs like *also* and *here*; dispreferred paraphrases like *is family* that also involve word choice are more likely to occur as well when the repetition penalty is increased. Since the anti-repetition scorer works by penalising the open-class words from the preceding discourse, it

[3] The data for this analysis was drawn from one run of the generation process: 2000 descriptions in total.

makes no distinction between paraphrases that involve the same words in different orders, so it is not surprising that it has no effect on the adverb-placement paraphrases.

The testbed on which the n-gram models was based was created fairly early in the COMIC project, so it does not cover all of the paraphrases that are now supported by the most recent COMIC grammar. For example, there are no examples in the testbed of sentences having the form *The design is country* or *The decorative tiles have floral motifs*. This means that descriptions including such sentences are penalised on corpus-based measures the same way as descriptions containing other, deliberately excluded paraphrases. For this reason, the dispreferred-paraphrase count is a better assessment of the text quality than the corpus similarity in this case; like Reiter and Sripada (2002), we also have a corpus that should not necessarily be treated as a "gold standard" for generation purposes.

Each of the parameters individually increased the variability of the generated descriptions to a similar degree. However, the effects were not additive: increasing both parameters did not have a greater effect than increasing one or the other in isolation. This indicates that there is a ceiling on the possible amount of variability when measured through edit distance on descriptions generated as these were, with fixed rules for content selection and structuring. A greater degree of variability could be introduced by also varying these features of the output.

5.4 Summary

We have described two implementation techniques for adding periphrastic variation to the output of the surface realiser in the COMIC system: ϵ-best sampling and anti-repetition scoring. Both of these techniques modified OpenCCG's realisation process to guide it to select options other than the single highest-scoring one when translating a disjunctive logical form into surface text. We performed two evaluations on the texts generated with and without these techniques. First, we gathered user preferences between description sequences generated with both techniques enabled and corresponding sequences generated with the techniques disabled. We then computed a set of automated metrics on the texts generated with and without the anti-repetition techniques, evaluating the individual impact of each technique.

The results of the user evaluation show that human judges found the description sequences generated with the anti-repetition techniques to be significantly better written and less repetitive than those generated without such measures; there was also a trend to find the texts with anti-repetition measures more understandable, particularly among the male participants. This result agrees with the results of other studies measuring human evaluations of texts that incorporated variation such as those of Stent *et al.* (2005) and Belz and Reiter (2006). By considering the effect of repetition in the discourse context, these results go beyond those found by Belz and Reiter (2006) in their evaluation of a range of knowledge-based stochastic surface realisers. Their *greedy roulette* implementation, which selected generation rules based on corpus probabilities, had a similar effect on the generated texts as our variation measures: their implementation "will tend to use different words and phrases in different texts, whereas the other statistical generators will stick to those with the highest frequency" (Belz and Reiter, 2006). This generator was penalised by automated evaluation measures because it tended to diverge from the corpus more than the others; however, the expert human evaluators ranked the output of this generator better than that of the bigram-based version. The effect of periphrastic variation on the other systems mentioned at the start of this chapter was never evaluated at all.

The results of the automated evaluation metrics were not surprising: texts generated with the anti-repetition techniques had lower n-gram scores and more variability across a description, and also had a higher level of dispreferred paraphrases. For most of the metrics, increasing the parameter value of either of the techniques had a significant impact, although these effects were not always additive: for example, increasing both the threshold and the penalty did not increase the variability beyond what can be obtained by increasing just one of the parameters. Anti-repetition scoring had a smaller effect on the n-gram scores and the dispreferred paraphrases than did ϵ-best scoring.

If we compare the user preferences with the results of the automated studies, we see that the users tended to prefer the outputs that had higher variability and lower n-gram scores. The materials generated for the user study happened not to have very many dispreferred paraphrases: there was one instance each of *is family* and sentence-initial *also* across the whole set. The responses for the *also* description are similar to those on the entire set; however, the description *is family* is the item mentioned in Section 5.2.4 for which the participants' responses were significantly different. On this item, 70% of the participants chose the description generated without the anti-repetition measures (and therefore without the dispreferred paraphrase) as being better written. This indicates that, while the variability introduced by the anti-repetition measures does appear to be appreciated by the participants, there is a danger that departing too far from the corpus examples can lead to outputs that they do not like.

The results of the experiments in this chapter contribute to all of the research questions set out at the start of the thesis: examining the impact of variation on the quality of generated output, testing the utility of making stochastic choices in the output, and comparing automated and human evaluations of output quality. For the first question, the experiments provide evidence that for this type of output, human judges generally prefer output generated with periphrastic variation to output generated without such variation; however, if the process of adding such variation selects paraphrases that were specifically excluded from the corpus, the positive effect can be negated.

The fact that the judges preferred the description sequences where the generation process selected options other than the single highest-scoring one based on the corpus also contributes to the second research question. The results of this study provide an initial indication that a corpus-driven generation system can produce output that has better overall quality if it selects from a wider space of available options rather than always choosing the alternative with the highest n-gram score.

That the overall n-gram-based scores of output quality did not agree with the human judges' preferences in turn contributes to the third research question. This could be partly because the corpus did not contain all of the paraphrases and the measure therefore penalised valid but unseen options; but even if the other options had been in the corpus, using the basic OpenCCG process would still have selected from a small set of high-scoring options, while the human evaluation shows that people prefer output that covers more of the space. There is also a suggestion that more targeted metrics that look for specific bad options—rather than penalising all departures from the corpus equally—correspond more closely to users' subjective preferences. Finally, we have also demonstrated another automated measure on the generated texts—the variability, as measured by the edit distance between descriptions—which did generally agree with the preferences of the human participants.

Chapter 6

An annotated corpus of conversational facial displays*

> The computer can't tell you the emotional story. It can give you the exact mathematical design, but what's missing is the eyebrows.
>
> Frank Zappa, quoted in Krahmer and Swerts (2004)

THE IMPLEMENTATIONS described in the two preceding chapters investigated adding different types of variation to the linguistic output of the COMIC output planner. We now turn our attention to the other major output channel in COMIC: the facial displays generated by the animated talking head. In Section 3.2.4, we described the basic implementation of output planning for the head. This process specifies a nod on a selection of the accented words in the output, and also chooses a range of conversational expressions that are generated before or after the speech. Nodding was the only facial display that the basic COMIC head was able to produce while speaking, so there is no possibility of creating more complex animation schedules for that head. The RUTH talking head (DeCarlo *et al.*, 2004) is able to produce a much wider range of facial displays, so for these experiments we use that talking head.

As discussed in Section 2.4.1, it is well-established that the body language that accompanies a spoken utterance is tightly related to the content of the speech. For faces in particular, Ekman (1979), for example, says that eyebrow raises "appear to coincide with primary voice stress, or more simply with a word that is spoken more loudly." Similarly, Graf *et al.* (2002) found that in their corpus of facial recordings, "rises of eyebrows are often placed at prosodic events, sometimes with head nods, at other times without." Translating these *appear to*, *often*, and

*This chapter is based on Foster (2007) and Foster and Oberlander (2007).

sometimes into concrete schedules for an embodied agent is a difficult task. Since there are so many contextual factors to consider and so many aspects of the embodied agent that can be controlled, this is a task for which data-driven techniques can be beneficial; the systems described in Section 2.4.2 all base their generation decisions in some way on the recorded behaviour of humans.

The study of multimodal corpora has become increasingly popular in recent years, as discussed in Section 2.2.2, and a number of large annotated corpora of human behaviour are now available (e.g., Carletta, 2006). However, the requirements for a corpus to be used in generation differ from those if the corpus is to be used for other common tasks such as analysis, summarisation, and information retrieval; this means that it is difficult to use an existing corpus directly for generation. For this reason, we designed, recorded, and annotated our own corpus of facial displays specifically for use in generating facial displays in COMIC.

This chapter describes the process of creating this corpus and is arranged as follows. In Section 6.1, we first discuss the particular needs that a corpus to be used in generation must satisfy, and describe why we therefore chose to create our own corpus rather than to adapt an existing one. Section 6.2 then describes the design of the corpus, how the scripts for the corpus were created and how they were recorded. Section 6.3 then describes the annotation scheme and tool that were used, and also discusses the measures that were taken to ensure that the annotation was reliable. In Section 6.4, we summarise the main patterns that were found in the corpus data. Finally, in Section 6.5, we discuss how this corpus fulfils the requirements for a generation corpus.

6.1 Requirements for corpus-based multimodal generation

To select the facial displays of the talking head, we need to study the displays produced by human speakers in similar contexts. As described in Section 2.2.2, the recording and annotation of multimodal corpora is an active research field at the moment; it would be helpful if some existing corpus could be used for this implementation.

However, if a multimodal corpus is to be used as a source for decision-making in generation, that adds some extra requirements to the corpus that do not necessarily arise in other common applications such as analysis, summarisation, and information retrieval. This means that existing corpora are often not suitable to be used "as is" as a source for generation. The pertinent considerations when designing a corpus-based multimodal generation system include the following:

- The contextual information necessary for making generation decisions must be represented in the corpus;

- The granularity of the cross-modal links must be appropriate to the generation task; and
- The generation system must be able to reproduce the corpus data in an appropriate way.

In the remainder of this section, we discuss each of these requirements in more detail; in Section 6.5, we describe how the face-display corpus meets each of these requirements.

6.1.1 Contextual information

In many cases, a multimodal corpus is based on naturally-occurring human behaviour: the participants being recorded are free to speak and act as they wish, and annotators then analyse the behaviour based only on the recordings. The corpus resulting from such a recording cannot contain any more information than what is available from observing the behaviour, and—possibly—from annotators applying their own judgement to add extra information, such as the dialogue-act and topic-structure annotations on the AMI corpus (Carletta, 2006).

For some embodied agents, this sort of surface-level annotation of context is sufficient; for example, for an embodied agent whose motion is selected entirely based on the features of the speech signal, such as that of Shimodaira *et al.* (2005), no deeper representation of the context is needed. However, in many cases, a generation system has available a much richer notion of context as it is planning its output. For example, the input to the Greta talking head (de Carolis *et al.*, 2002) represents the target information structure and affective content of its utterances; similarly, the output of the agent of Stone *et al.* (2004) depends on factors including the move that the user attempted in the video game and the outcome of the attempt.

In the terminology of van Deemter *et al.* (2006), this means that a corpus to be used for generation must be *semantically transparent*: the pragmatic conditions under which the data in the corpus was produced must be known so that the generation algorithm has access to them. The required contextual information can be included in the corpus in two ways. Either it can be manually added after the fact by annotators, or the corpus can be created in such a way that the required information is present by design. The latter can be achieved by using corpora based on scripted output in the domain of the eventual target system. If the human being recorded is following a known script, then all of the relevant contextual information can easily be added to the corpus at construction time. This approach was taken by Stone *et al.* (2004), for example. It has the advantage that no additional manual effort is required; however, it also has the disadvantage that the corpus must be created specifically for the target application, which rules out using existing annotated corpora.

6.1.2 Cross-modal links

In many multimodal corpora, each separate modality is represented on its own timeline, with the only links between modalities those that are implicit in the timestamps. For example, in the AMI corpus, there are many levels of links corresponding to different aspects of the spoken signal; however, the gesturing behaviour is represented on its own timeline with its own start and stop times. This type of representation is adequate if the goal is to extract events or to analyse human behaviour; however, if the goal is to generate novel output based on the corpus, more explicit links between the modalities are useful, as the temporal structure may not coincide with the underlying generation process. Kipp *et al.* (2006), for example, "found that the claim that gesture stroke and lexical affiliate always co-occur is often wrong" while annotating the gesturing behaviour of skilled public speakers. Similarly, in the context of paper-based documents, both Bateman *et al.* (2001) and Power *et al.* (2003a) have found that the rhetorical structure of a document is different than its layout structure.

One important design decision for a corpus is the level at which cross-modal links are represented—that is, the size of the segment on each channel that can be associated with segments on other channels. For example, when associating multimodal behaviours with speech, motions may be associated with phonemes or syllables, with single words, with syntactic constituents, or with arbitrary sequences of words. Which of these is chosen depends on the level at which the generation system will later be selecting these motions; if the assumptions are later changed, it may prove necessary to re-annotate the entire corpus with the new schema.

As well as the level of representation, the criteria for making a link must be established: is the choice based strictly on temporal or spatial coincidence, or is semantic information also used? The former is easier to annotate, and may even be automatically inferred from an existing annotated corpus, but may not generalise as readily to new outputs; the latter requires a more involved annotation process that makes more demands on the annotators for careful judgement calls. For example, Kipp *et al.* (2006) chose to record temporal co-occurrence and lexical affiliation as separate attributes when annotating hand gestures for generation; temporal co-occurrence was derived largely automatically from the video, but annotating the lexical links relied on "gesture literature and sometimes intuition."

6.1.3 Reproducing corpus data

For multimodal generation, it is generally not the case that corpus data can be directly combined to produce output in the way that diphones can be concatenated for speech synthesis, or words for text generation. In most cases, a multimodal generation system creates entirely synthetic output by specifying commands for each of the relevant output channels, rather than combining existing pieces of output directly. Even in cases where motion-capture data is used directly (e.g., Stone *et al.*, 2004; Cunningham *et al.*, 2005), the recorded motions must still be mapped to animation commands and synchronised with the speech.

When the generated output is specified at a higher level, then more complex mappings must be made. For example, Kipp *et al.* (2006) use an annotation scheme for hand gestures that makes the conscious decision not to represent every single feature of the motion, but rather to capture the essentials, in some cases using gesture "lexemes" to abstract over the data. To recreate a gesture schedule annotated using this scheme, the motion specifications must be translated into specific commands for the animation engine.

6.2 Recording

All of the criteria listed in the preceding section mean that it is difficult to use an existing multimodal corpus directly for generation. For this reason, we decided to record and annotate a corpus specifically for use in generating facial displays in the COMIC system. The corpus was based on scripted output in the domain of the COMIC system, with full contextual information in the recording scripts, and was annotated at a level that can be directly translated into commands for the generation system.

Similar corpora based on the recorded actions of a individuals have been used in several other embodied-agent implementations, including those of Stone *et al.* (2004) and Kipp (2004). The advantage of this approach is that the motions are both naturally varying and coherent. A corpus based on the behaviour of a number of different people could contain conflicting displays; it is also possible, depending on the generation strategy, that basing decisions on a multi-person corpus could end up selecting "average" behaviours and losing the natural variation. In state-of-the-art unit-selection speech synthesis (Hunt and Black, 1996), synthetic voices are also generally based on large single-speaker databases for exactly these reasons. The danger in using a single speaker is that the recorded motions may be too idiosyncratic, and may not have the desired effect when used for generation; in Section 6.4 and Chapter 7, we verify in several ways that the motions of our corpus speaker vary systematically, and that these patterns are identifiable.

In this section, we describe how the script for the recording was created and how the recording proceeded; in the next section, we describe the annotation scheme and tool.

Figure 6.1: Sample sequence from the recording script

- *Start of a new description*
  Here is ANOTHER design.
- *More about the current design; they dislike the first feature, but like the second one*
  There are GEOMETRIC SHAPES on the decorative tiles, but the tiles ARE from the ARMONIE series.
- *More about the current design; comparing to last design*
  ONCE AGAIN the tiles are by STEULER, but HERE it is in the CLASSIC style.

6.2.1 Speaker

The speaker who was recorded was a male postgraduate student at the University of Edinburgh who was also an amateur actor. He was not told before the experiment what the goal of the recording was, only that he would be asked to read a number of sentences into a camera. He was paid for his participation.

6.2.2 Methodology

A script for the recording was created consisting of a set of typical sentences generated by COMIC, chosen to provide coverage of all of the syntactic patterns that were covered by the grammar included in the final integrated system. By request of the speaker, the sentences were combined into coherent turns for the recording, where each turn mentioned a feature of a tile design at most once. Figure 6.1 shows one of the description sequences that was used; the information in italics indicates the contextual information associated with each sentence.

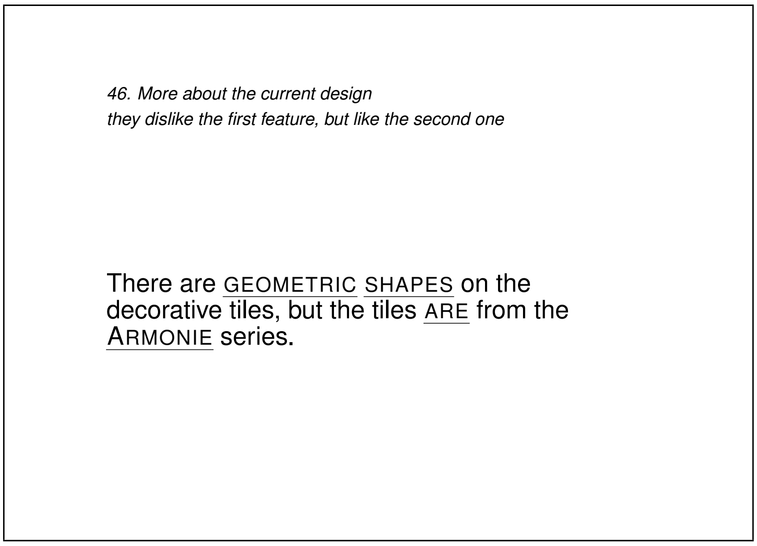
The sentences were presented one at a time to the speaker, who was instructed to read each sentence out loud as expressively as possible into a video camera directed at his face. Each sentence was presented in a large font on a laptop computer directly in front of the speaker, with the camera positioned directly above the laptop; this ensured that he was looking towards the camera at all times.

A sample slide from the recording session is shown in Figure 6.2. The words in the sentences for which the COMIC presentation planner specified pitch accents were highlighted with capital letters and underlining. As well, any applicable user-model and dialogue-history information was included at the top of the slide, to ensure that the sentence was said in the intended context. Recording the full set of sentences took approximately two hours in total.

6.2.3 Materials

To create the scripts for the corpus, we selected a set of sentences generated by the COMIC text planner (Section 3.2), including the user-model and dialogue history modifications described

Figure 6.2: Sample slide from recording session



in Section 4.2. The sentences were chosen to provide coverage of all of the syntactic patterns that can be generated by COMIC; in total, 384 sentences were selected. For each generated sentence, we also stored the derivation tree generated by the OpenCCG surface realiser. Every node in the derivation tree for each sentence in the script was initially annotated with all of the available syntactic and pragmatic information from the output planner, including the following features:

- The user-model evaluation of the object being described (positive or negative);
- Whether the fact being presented was previously mentioned in the discourse (*as I said before, . . .*) or is new information;
- Whether the fact is explicitly compared or contrasted with a feature of the previous tile design (*once again . . . but here . . .*);
- Whether the node is in the first clause of a two-clause sentence, in the second clause, or is an only clause;[1]
- The surface string associated with the node;
- The surface string, with words replaced by semantic classes or stems drawn from the grammar (e.g., *these designs are classic* becomes *this [mental-obj] be [style]*); and
- Any pitch accents specified by the text planner.

[1]No sentence in the script had more than two clauses.

Figure 6.3: Annotated OpenCCG derivation tree

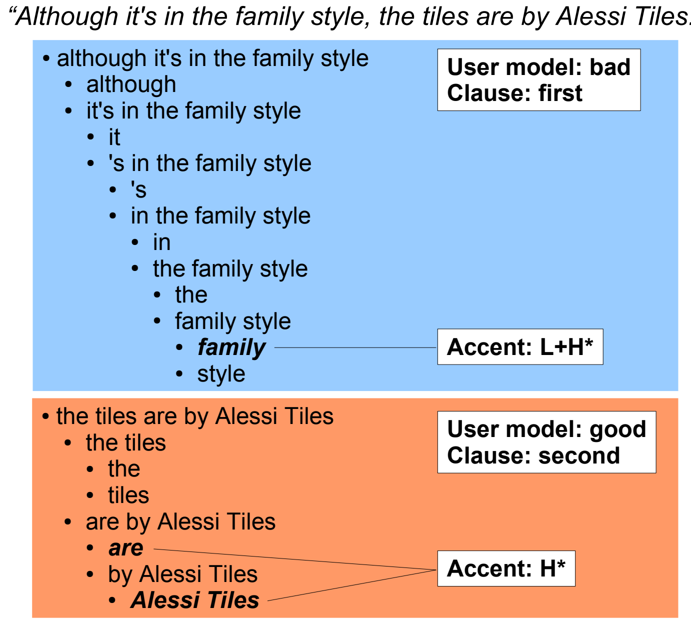


Figure 6.3 illustrates the annotated OpenCCG derivation tree for a sample sentence drawn from the recording script. The annotations indicate that every node in the first half of this sentence is associated with a negative user-model evaluation and is in the first clause of a two-clause sentence, while every node in the second half is linked to a positive evaluation and is in the second clause of the sentence. The figure also shows the pitch accents selected by the output planner.

To create the coherent turns requested by the speaker for the recording session, each introductory sentence (e.g., *This design is classic*) was repeated four times in the script, while filler sentences (such as *Here is another design*) were added as necessary to complete the set of turns. In total, the recording script consisted of 440 content sentences—including the repeated introductory sentences—and 151 filler sentences.

6.3 Annotation

Once the corpus was recorded as described in the preceding section, the next step was to annotate the facial displays that occurred. We first used Anvil (Kipp, 2004) to split the recorded video into individual clips corresponding to each non-filler sentence. Since several of the sentences were repeated during the session, this produced a total of 444 video clips. This section describes how the facial displays in each of the clips were annotated. In Section 6.3.1, we first present the annotation scheme that was used; Section 6.3.2 then describes the tool that was used to perform the annotation. At the end of this section, in Section 6.3.3, we describe the measures that were taken to ensure that the annotation was reliable.

6.3.1 Annotation scheme

The speaker's facial displays were linked to the span of nodes in the OpenCCG derivation tree with which they were temporally related. This scheme was chosen for two reasons. First, due to the nature of the recording scripts, we had available the full, contextually-annotated tree for each sentence in the corpus. Second, we intended to use this full set of information and these trees in the generation process. In an initial annotation of the same set of videos (Foster and Oberlander, 2006), we enforced the restriction that motions could only be associated with single leaf nodes; however, as described in Section 6.5, that assumption proved to be unrealistic in practice, so for this corpus we allowed a motion to be associated with an arbitrary span of nodes in the tree.

A display was associated with the full span of words that it coincided with, as follows. If a single node in the derivation tree covered exactly all of the relevant words, then the annotation was placed on that node; if the words spanned by a display did not coincide with a single node, it was attached to the set of nodes that did span the necessary words. For example, in the derivation shown in Figure 6.3, the sequence *the family style* is associated with a single node, so a motion temporally associated with that sequence would be attached to that node. On the other hand, if there were a motion associated with *the tiles are*, it would be attached to both the *the tiles* node and the *are* node.

The following were the features that were considered; for each feature, we note the corresponding Action Unit (AU) from the well-known Facial Action Coding System (Ekman and Friesen, 1978; Ekman *et al.*, 2002).

- Eyebrows: up (AU 1+2) or down (AU 4)
- Eye squinting (AU 43)
- Head nodding: up (AU 53) or down (AU 54)
- Head leaning: left (AU 55) or right (AU 56)
- Head turning: left (AU 57) or right (AU 58)

This set of displays was chosen based on a combination of three factors: (a) the various emphatic facial displays that have been documented in the literature (Section 2.4.1), (b) the capabilities of the RUTH talking head, and (c) the actual movements of the speaker during the recording session. Figure 6.4 shows some typical display combinations from the videos.

Figure 6.4: Characteristic facial displays from the recordings

(a) Neutral

(b) Right turn + brow raise

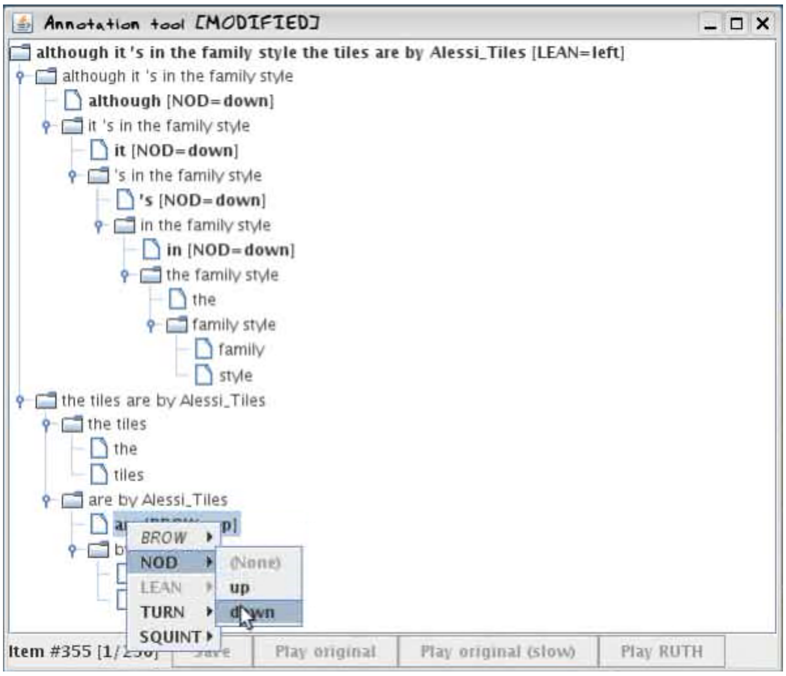
(c) Left lean + brow lower

(d) Squint + brow raise

(e) Upward nod

(f) Downward nod

Figure 6.5: Annotation tool



6.3.2 Annotation tool

The tool for the annotation was a custom-written program that allowed the coder to play back a recorded sentence at full speed or slowed down, and to associate any combination of displays with any node or set of nodes in the OpenCCG derivation tree of the sentence. The tool also allowed the coder to play back a proposed annotation sequence on a synthetic talking head to verify that it was as close as possible to the actual motions. Figure 6.5 shows a screenshot of the annotation tool in use on the sentence from Figure 6.3. In the screenshot, a left turn is attached to the entire sentence (i.e., the root node), while a series of nods is associated with single leaf nodes in the first half of the sentence. The annotator has already attached a brow raise to the word *are* in the second half and is in the process of adding a nod to the same word.

The output of the annotation tool is an XML document including the original contextually-annotated OpenCCG derivation tree of each sentence, with each node additionally labelled with a (possibly empty) set of facial displays. Figure 6.6 shows the fully-annotated version of the sentence from Figure 6.3. This document includes the contextual features from the original tree, indicated by italics: every node in the first subtree has `um="b"` and `first="y"`, while every node in the second subtree has `um="g"` and `first="n"`; the accented items also have an `accent` feature. Every node also specifies the string generated by the subtree that it spans, both in its surface form (`surf`) and with semantic-class and stem replacement (`sc`). This tree also includes

Figure 6.6: Annotated sentence from the corpus

```
<node surf="although it 's in the family style the tiles are by Alessi_Tiles" LEAN="left"
    sc="although [pro3n] be in the [style] [abstraction] the [phys-obj] be by [manufacturer]">
  <node surf="although it 's in the family style" um="b" first="y"
      sc="although [pro3n] be in the [style] [abstraction]">
    <node surf="although" um="b" first="y" NOD="down" />
    <node surf="it 's in the family style" um="b" first="y"
        sc="[pro3n] be in the [style] [abstraction]">
      <node surf="it" stem="pro3n" um="b" first="y" NOD="down" />
      <node surf="'s in the family style" um="b" first="y" sc="be in the [style] [abstraction]">
        <node surf="'s" stem="be" um="b" first="y" NOD="down" />
        <node surf="in the family style" um="b" first="y" sc="in the [style] [abstraction]">
          <node surf="in" um="b" first="y" NOD="down" />
          <node surf="the family style" um="b" first="y" sc="the [style] [abstraction]">
            <node surf="the" um="b" first="y" />
            <node surf="family style" um="b" first="y" sc="[style] [abstraction]">
              <node surf="family" sc="[style]" accent="L+H*" um="b" first="y" NOD="down" />
              <node surf="style" sc="[abstraction]" um="b" first="y" />
            </node>
          </node>
        </node>
      </node>
    </node>
  </node>
  <node surf="the tiles are by Alessi_Tiles" um="g" first="n"
      sc="the [phys-obj] be by [manufacturer]">
    <node surf="the tiles" um="g" first="n" sc="the [phys-obj]">
      <node surf="the" um="g" first="n" />
      <node surf="tiles" sc="[phys-obj]" stem="tile" um="g" first="n" />
    </node>
    <node surf="are by Alessi_Tiles" um="g" first="n" sc="be by [manufacturer]">
      <node surf="are" stem="be" accent="H*" um="g" first="n" BROW="up" NOD="down" />
      <node surf="by Alessi_Tiles" um="g" first="n" sc="by [manufacturer]">
        <node surf="by" um="g" first="n" />
        <node surf="Alessi_Tiles" sc="[manufacturer]" accent="H*" um="g" first="n" />
      </node>
    </node>
  </node>
</node>
```

the facial displays added by the coder in Figure 6.5, indicated by underlining: (`LEAN="left"`) attached to the root node), a number of downward nods (`NOD="down"`) on individual words in the first half of the sentence, and a nod with a brow raise (`BROW="up"`) on *are* near the end.

6.3.3 Reliability of annotation

Several measures were taken to ensure that the annotation process was reliable. As the first step, two independent coders each separately processed the same set of 20 sentences, using a draft of the annotation scheme. The coders discussed the differences in the outputs and agreed on a final scheme, which one of those coders then used to process the entire set of 444 sentences. This set of sentences is the set that was used for the analysis described at the end of this chapter and all of the experiments in the following chapters.

As a further check on the reliability of the annotation scheme, a third annotator was then instructed on the use of the annotation tool and scheme, and used the tool to process 286 sentences (approximately 65% of the total sentences); the instructions given to this annotator are shown in Appendix D.

To assess the agreement between these two annotators, we used a version of the β agreement coefficient proposed by Artstein and Poesio (2005). β is designed as an agreement coefficient that is weighted, that applies to multiple coders, and that uses a separate probability distribution for each coder. Like other weighted agreement measures, it is based on the ratio between the observed and expected disagreement in the corpus. To use this coefficient, it is necessary to define a measure that computes the distance between two proposed annotations. It is common to define a measure that permits degrees of agreement, so that partial agreement is penalised less than total disagreement.

To compute the observed disagreement $D_o(S)$ on a sentence S, we used a measure similar to that proposed by Passonneau (2004) for measuring agreement on set-valued annotations. Intuitively, for each display proposed by each annotator on the sentence, we search for a corresponding display proposed by the other annotator—one with the same value, and a similar span of nodes. If both propose exactly the same display, that indicates no disagreement (0); if one display covers a strict subset of the nodes covered by the other, that indicates minor disagreement ($\frac{1}{3}$); if the nodes covered by the two proposals overlap, that is a more major disagreement ($\frac{2}{3}$); and if no corresponding display at all can be found from the second annotator, then that indicates the maximum level of disagreement (1). Full details of the computation of D_o are in given Appendix E.1.

The expected disagreement $D_e(S)$ for a sentence S depends on the length of that sentence. Intuitively, we used the probability of each annotator assigning a facial display to spans of varying lengths to estimate the likelihood of their assigning identical, super/subset, overlapping, or distinct annotations to the sentence, and take the weighted sum of these probabilities to compute the expected disagreement. The formal details are given in Appendix E.2.

The overall observed disagreement D_o is the arithmetic mean of the disagreement on each sentence processed by both annotators; similarly, the overall expected disagreement D_e is the mean of the expected disagreement across all of the sentences. To compute the value of β, we subtract the ratio between these two values from 1:

$$\beta = 1 - \frac{D_o}{D_e}$$

As Artstein and Poesio (2005) point out, there is no significance test for weighted measures such as β, and the actual value is strongly affected by the distance metric that is selected; however, β values can be compared with one another to assess degrees of agreement.

The overall β value between the two annotators on the full set of 286 sentences processed by the final annotator after training was 0.561, with β values on individual facial displays ranging from a high of 0.661 on nodding to a low of 0.285 on squinting (a very rare motion). To put these values into context, we can also compute β on the set of 20 sentences processed by the final annotator as part of his training process (which are not included in the set of 286), the overall β value for these sentences is 0.231, with negative values for some of the individual displays. This demonstrates that the training process had a positive effect on agreement between these two annotators.

6.4 Patterns in the corpus

As an initial analysis of the corpus, we investigated the features of a node to see which had the most significant effect on the facial displays occurring on that node. To determine the most significant factors, we performed multinomial logit regression and selected the factors and factor interactions that had the most significant effects on the distribution of each display. In this section, we list the most significant factors and give a qualitative description of the impact that each had on the facial displays; for a more detailed analysis, please see Appendix F.

The occurrence of all facial displays on a node was significantly affected by the user-model evaluation attached to that node, as well as by some combination of the following features: the predicted pitch accent, the clause of the sentence, and the number of leaf nodes spanned by the node. Not all of these factors affected every display, but none of the other factors had as significant an effect. The main contextual influences on face-display behaviour are similar in the sub-corpus processed by the final annotator; again, see Appendix F for details.

Nodding and brow raising were both more frequent on nodes with any sort of predicted pitch accent. In negative user-model contexts, eyebrow raising, squinting, and left leaning were all relatively more frequent; in positive contexts, on the other hand, the relative frequency of right turns and brow raises was higher. In the first half of two-clause sentences, brow lowering was also more frequent, as was upward nodding, while downward nodding and right turns showed up more often in the second clause of two-clause sentences.

Several of these patterns agree with previous findings on conversational body language. The increased frequency of nodding and brow raising on accented words agrees with the findings of Ekman (1979); Cavé *et al.* (1996); Graf *et al.* (2002); Keating *et al.* (2003); Krahmer and Swerts (2004); Flecha-García (2006), all of whom noted similar displays on prosodically accented parts of the sentence.

The speaker's tendency to move right on positive descriptions and left on negative descriptions is also consistent with other findings. According to the influential work of Davidson and colleagues (Davidson *et al.*, 1990; Davidson and Irwin, 1999), emotion and affect processing are asymmetrically organised in the human brain. The right hemisphere is associated with negative affect (and withdrawal behaviours), and the left with positive affect (and approach behaviours). Because both perceptual and motor systems are contra-laterally organised, this means that higher levels of right hemisphere activity are associated with attention being oriented towards the left, while higher levels of left hemisphere activity are associated with attention being oriented to the right; this fits with our speaker's pattern of movements.

6.5 Satisfying the requirements for a generation corpus

This corpus addresses all of the requirements for a generation corpus outlined in Section 6.1. As in many previous corpora, we ensured that the corpus included full contextual information by basing it on output created in known pragmatic contexts. Also like many others, we designed the annotation scheme to consider only those behaviours (head and eyebrow motions) that could easily be controlled on the target talking head. Note in particular that we chose not to annotate the amplitude of mouth movements, despite the fact that it has been documented to be correlated with prosodic emphasis, because this is not a dimension that can easily be controlled on the target head.

In the final corpus, cross-modal links were made between facial displays and sets of nodes in the OpenCCG derivation tree, which is useful in the generation process and also allowed for respectable inter-coder agreement. Selecting a linking level took some effort and experimentation, and two other versions were considered before settling on the one in the final annotation scheme. We can use the data in the corpus to test whether these modifications to the scheme were justified.

In a previous study using the same video recordings but a different, simpler scheme (Foster and Oberlander, 2006), facial displays could only be associated with single leaf nodes (i.e., words); that is, in the terminology of Ekman (1979), all motions were considered to be *batons* rather than *underliners*. Based on the data in the current corpus, that restriction is clearly unrealistic: the mean number of nodes spanned by a display in the full corpus is 1.95, with a maximum of 15 and a standard deviation of 2. The results are similar in the sub-corpus produced by the additional coder, with a mean node span of 2.25.

Another extension to the original annotation scheme was to allow displays to be attached to more than one node in the tree in cases where the span of words was not a syntactic constituent. The corpus data also supports this extension: approximately 6% of the annotations in the main corpus—165 of 2826—were attached to more than one node in the derivation tree, while the additional coder attached 4.5% of displays to multiple nodes.

6.6 Summary

In this chapter, we have outlined the requirements for a multimodal corpus to be used for generation and explained why that led us to record and annotate a custom corpus in the domain of the COMIC system. Since the corpus was based on scripted output from the COMIC output planner, all of the contextual information that is needed to make a decision on facial displays is included with no extra effort; the processing effort could instead be concentrated on detecting the facial displays and linking them to node spans in the OpenCCG derivation tree for the sentences.

Two annotators separately processed the sentences in the corpus. To measure agreement on this complex task, we used a version of Artstein and Poesio's (2005) β agreement measure with a distance measure based on the one proposed by Passonneau (2004); this measure allows for degrees of agreement in cases of overlapping annotations. We found an acceptable level of agreement; in particular, agreement increased greatly between the initial sentences processed by the final annotator and the final set of sentences, indicating that the training process was successful in teaching that annotator to follow the scheme. Also, the patterns in the corpus are the same for both annotators, as described in Appendix F.

Many contextual factors influenced the distribution of facial displays. The most influential was the user-preference evaluation, which affected the distribution of all of the facial displays that were annotated; other significant factors included the predicted pitch accent, the clause of the sentence, and the number of leaves spanned by a node. Many of these patterns are consistent with previous findings on facial displays and other aspects of body language. This provides evidence that the displays made by the particular speaker recorded for this corpus are similar to those known to be typical for the larger population.

In the next chapters, we investigate different ways of making use of the information from this corpus in selecting the behaviours for the talking head in the COMIC system.

Chapter 7

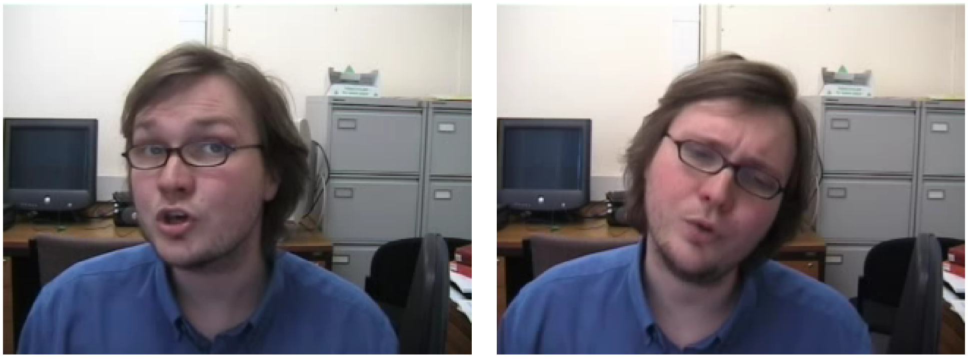
Rule-based generation of facial displays

> **Colonel O'Neill:** To be fair, General, I did it. Carter and Daniel protested. And Teal'c, well, he really didn't say anything, but I could tell he was opposed to my actions by the way he cocked his head and sort of raised his eyebrow.
>
> Jonathan Glassner, *Stargate SG-1: "Shades of Grey"*

THE SINGLE MOST influential factor in the face-display corpus described in the preceding chapter was the user-preference evaluation. In this chapter, we implement a simple rule based on the user-preference evaluation to select displays for the talking head and perform two human studies to test the output generated using this rule. In the first experiment, videos were created combining neutral speech with characteristic positive, negative, and neutral facial displays, as well as no displays at all. Participants were then asked to identify the intended user-model evaluation of the sentence based on the videos. In the second experiment, sentences with overt positive and negative content were combined with positive, negative, and neutral facial displays, and participants were asked for their preferences between videos with congruent and conflicting speech and facial displays. These experiments are similar to those carried out by, for example, Krahmer and Swerts (2005) to measure users' ability to perceive uncertainty through body language, and Rehm and André (2005) to test their ability to perceive deceptive non-verbal behaviour.

These studies serve two purposes. On the one hand, they provide verification that the characteristic motions of the speaker are identifiable to users when resynthesised; this is a necessary step to take before creating any further, more complex implementations based on the data in

Figure 7.1: Characteristic positive and negative displays from the speaker



(a) Positive **(b)** Negative

the corpus. On the other hand, the implementation itself provides an initial version of a more extensible method of face-display selection than is available in the basic COMIC system, as we now make use of a talking head that can make a greater range of displays while speaking.

Like the experiments in Chapter 4, these studies also address the first research question set out at the start of this thesis: again, we present output to participants and test whether they prefer the version that is correctly tailored to the dialogue context. However, in this case the textual content is held constant, while the variation is all in the facial displays.

This chapter is arranged as follows. In Section 7.1, we first present the corpus-based rules that were used to select the facial displays, while in Section 7.2 we describe how the RUTH head was used to synthesise the resulting display schedules. We then describe two experiments that measured human reactions to facial displays generated using this technique. The experiment in Section 7.3 assesses users' ability to recognise the intended user-model context based on the facial displays, while the experiment in Section 7.4 measured whether users could distinguish between consistent and inconsistent combinations of linguistic content and facial displays. At the end, we summarise the results of these two studies and discuss their implications in the context of the research questions of this thesis.

7.1 Implementation: Rule-based generation

At the end of the preceding chapter, we described the influence of a number of factors on the facial displays produced by the speaker recorded for the corpus. The biggest single factor determining the facial displays was the user-model evaluation, which had an influence on all of the displays that were considered. The following displays were relatively more frequent in a positive user-model context (Figure 7.1(a)):

- Turning to the right
- Raising the eyebrows

Conversely, the following displays were all relatively more frequent in a negative user-model context (Figure 7.1(b)):

- Leaning to the left
- Lowering the eyebrows
- Squinting the eyes

The overall most frequent facial display across all contexts was no motion at all, with the next most frequent option a downward nod with no other motions. Nodding was somewhat more frequent in positive user-model contexts; however, it was also quite frequent across all contexts.

We use these patterns as the basis for our selection of facial displays. The input to the system is the OpenCCG derivation tree of the intended utterance, contextually annotated as in Figure 6.3 on page 112. The generation algorithm selects the leaf nodes in which a fact about a tile design is mentioned—i.e., mentions of styles, colours, manufacturers, series, or decorative motifs. It specifies a facial display to accompany every node of this type, with the type of display depending on the user-model information attached to the node, as follows:

Positive context right turn + brow raise

Negative context left lean + brow lower + eye squint

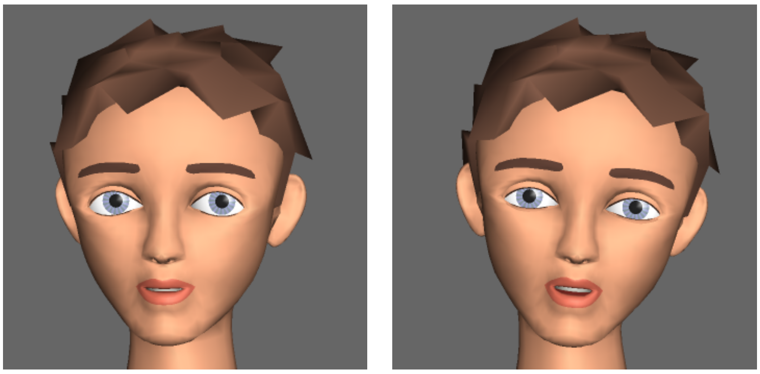
Neutral context downward nod

Figure 7.2 shows the characteristic positive and negative facial displays synthesised on the RUTH head.

7.2 Generating RUTH videos

For this experiment, and for the ones that follow, we created stand-alone videos in an offline process based on the COMIC output-generation process outlined in Section 3.1.5.1. To create a video, we first built the logical form for the sentence as normal and sent it to OpenCCG for realisation. The output of OpenCCG is the textual content of the sentence, as well as the full derivation tree. The tree was then annotated with all of the relevant contextual and syntactic information, and then the above process was used to assign facial displays to specific nodes based on the above rules.

Figure 7.2: Synthesised positive and negative displays



(a) Positive **(b)** Negative

Next, the generated text was sent to the speech synthesiser, which prepared the waveform and returned the timing information for all of the words in the synthesised sentence, along with their phonemes. The timing information was then used together with the specified displays to create a low-level schedule for RUTH, indicating both phonemes (for lip synchronisation) and the displays and their timing; a display was scheduled to begin and end at the start and end of the node or nodes that it spans.

The waveform and timing information were then sent together to RUTH, which created a sequence of still images specifying the frames of the animation corresponding to the schedule. Finally, we used the `mencoder`[1] video-encoding program to combine the frames and add the soundtrack to create a video of the generated display schedule for the sentence, and `ffmpeg`[2] to encode the video in the correct format.

7.3 Human evaluation: Recognisability of facial displays

In Section 7.1, we presented a simple rule for selecting facial displays based entirely on the user-model evaluation of the objects being described. The displays to use in each context are based on the most characteristic behaviours of the recorded speaker in the corresponding contexts. This first study aims to measure whether participants were able to identify the intended user-model context of a sentence based only on the accompanying facial displays.

To do this, we used syntactically neutral sentences and created videos in which the talking head used positive, negative, neutral, or no facial displays and ask participants what they believed the user-model orientation of the sentence to be. A similar paradigm was has been

[1] http://www.mencoder.hu/
[2] http://ffmpeg.org/

used by many, including (Krahmer and Swerts, 2005; Rehm and André, 2005; Marsi and van Rooden, 2007) to test the influence of facial displays on the interpretation of a sentence.

This experiment was designed to assess whether the positive and negative displays have the intended effect on the user. The results also give some indication of users' perceptions of the displays intended to be neutral (nodding alone) and of the videos with no motions other than lip-synchronisation.

The hypotheses for this experiment were as follows:

1. The positive displays will be perceived as positive.
2. The negative displays will be perceived as negative.
3. There will be no significant pattern of responses for either the neutral displays or the schedules with no motion.

7.3.1 Participants

Like the study described in Section 5.2, this experiment was also run through a web page, and participants were also recruited via an email to the Informatics departmental student mailing list and through the Web Experiments Portal; participants were also entered into a prize draw. The details of the participants who took part in this experiment are as follows:

Total number	26			
Gender	Female: 14	Male: 12		
Age	Under 20: 1	20–29: 20	30 and over: 4	
Computer experience	Beginner: 1	Middle: 14	Expert: 11	
Native language	English: 11	German: 4	Spanish: 4	Other: 6

7.3.2 Methodology

Participants were shown a series of 16 videos created by the video-generation technique described in the preceding section. The videos were presented as embedded Shockwave Flash videos in a popup window on a web page. The textual content of all of the videos was neutral with regard to the user model in all cases, and the facial displays of the speaker were generated in one of four different ways: using characteristic positive, negative, and neutral displays as described above, or using only lip-synchronisation. After viewing each video, the participants were asked to choose whether they thought the speaker believed that they liked or disliked the feature being described; they could also choose *don't know* if they were unable to tell the intended user-model evaluation. The interface and instructions for this experiment are presented in detail in Appendix C.3.

Figure 7.3: Sample schedules from the recognition study

	This	*design*	*features*	*orange*	*in*	*the*	*colour scheme.*
Positive				tn=r, bw=u			
Negative				ln=l, bw=d, sq			
Neutral				nd=d			
Nothing							

All participants saw videos of the same 16 sentences, in an individually randomly-chosen order. The display schedules were also randomly allocated to the videos so that participants saw four schedules of each type (positive, negative, neutral, no motion).

7.3.3 Materials

To create the materials for this study, we used the basic text planner (Section 3.2) to generate 16 sentences with no overt user-model information: for example, *This design features orange in the colour scheme.* For each sentence, we used the rules described in the above section to create three facial-display schedules, one with each of the positive, negative, and neutral display types. We then used the RUTH head to generate videos of the head speaking the sentence with each of the sets of facial displays; we also created a fourth video for each in which the head made no motions other than lip-synchronisation. Figure 7.3 shows the facial-display schedules generated for one of the sentences: *tn=r* means a right turn, *ln=l* a left lean, *bw=u* or *d* eyebrow movements up or down, *sq* a squint, and *nd=d* a downward nod.

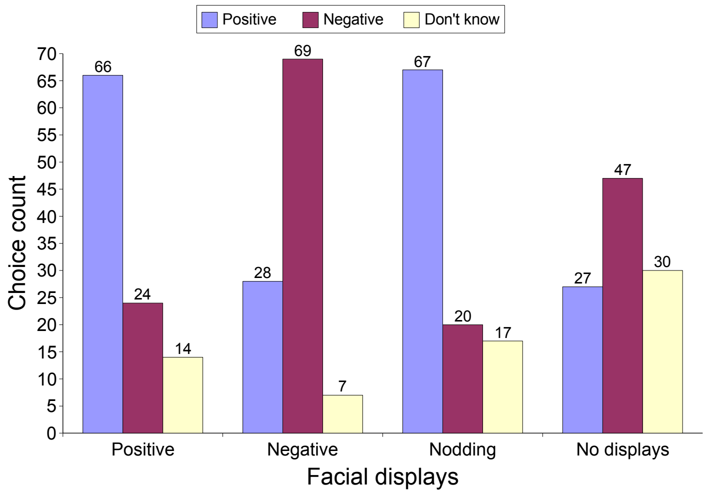
7.3.4 Results

All 26 of the participants in this study gave a response for all of the videos, yielding a total of 416 judgements, 104 on each video type. None of the demographic factors had any effect on the results of the experiment.

The overall results of this study are presented in Figure 7.4. The x axis in this graph shows the actual facial displays that were used, while the three bars in each group show the number of responses in which participants believed that the speaker was being positive or negative, or whether they could not tell, respectively. For example, the videos with negative facial displays were identified as positive on 28 trials and as negative on 69 trials, while on the remaining 7 trials the judges were unable to make a decision.

As in previous experiments, we can also use a binomial test to test whether the pattern of choices for each video type is significant, using an expected proportion of $\frac{1}{3}$ as this was a three-way choice rather than two-way as in the experiments in Chapters 4–5. These results indicate

Figure 7.4: Results for the recognition study



that participants were successful at identifying the intended polarity of both the positive and negative facial displays (66% and 63%, respectively; $p < 0.0001$), and also tended to identify the neutral displays (nodding only) as positive (64%, also $p < 0.0001$); for the schedules with only lip-synchronisation, they had a weaker but still significant tendency to identify them as negative (45%, $p < 0.05$). The differences between the responses for the different video types are statistically significant: $\chi^2 = 87.4$, $df = 6$, $p < 0.0001$.

7.3.5 Discussion

The results of this experiment clearly confirm our first two hypotheses: participants were able to identify the intended user-model evaluation when the head used both the characteristic positive and negative displays of the corpus speaker. However, the third hypothesis is not confirmed: with the neutral facial displays, participants made essentially identical judgements as on the positive displays, while when there were no facial displays, participants had a weaker but significant tendency to identify the intended evaluation as negative.

The tendency of participants to identify a sentence with a nod as positive could be due to several factors. First, in the corpus, nodding is actually relatively more frequent as well in positive user-model contexts, although it is also quite frequent across all contexts (Appendix F.3). So they could have been interpreting the nodding as a positive facial display even though it was not intended as such. Another possibility, as mentioned in Section 4.4, is that the default interpretation of output in COMIC is that the object being described is given a positive context unless it is explicitly marked as negative.

There was a weak tendency for videos with no motion at all to be identified as negative, which initially seems to contradict the above assumption. However, in the psychiatric literature, *flat affect*—the absence of emotional expression—is associated with disorders including depression (Gaebel and Wölwer, 2004); this factor may well have contributed to the selections made by participants on these trials.

7.4 Human evaluation: Consistency of facial displays

The results of the previous study shown that participants were able to correctly identify the characteristic negative facial displays produced by the recorded speaker, and that they tended to identify as positive both the displays intended as positive and the one intended as neutral. In this experiment, we measured whether this ability to identify the user-model polarity of facial displays translated into any preferences regarding the consistency between the evaluation expressed in the text and that expressed by the talking head. This experiment also addressed the issue of whether there is any difference for participants between the facial displays we defined as "positive" and "neutral" in Section 7.1, by eliciting user preferences between minimal pairs of videos with these two display types in different conversational contexts.

Experiments involving inconsistent cues in the language and the accompanying facial displays have been used to answer two main types of questions: which of these two channels has the greater effect on the perception of the intended effect, or whether participants prefer or perform better with congruent cues to conflicting ones. The user task in the former type of study is generally to indicate whether they perceive the message from the speech or the one from the body language. For example, Swerts and Krahmer (2004) manipulated video sequences so that the visual stress was on one syllable and the phonetic stress was on another, and asked participants to select which of the syllables was more prominent. The results indicated that the auditory cues from the visual signal were more prominent, although both channels had an effect. A study of the latter type is that of Berry *et al.* (2004), who compared several versions of an embodied agent and found that participants performed best on a recall task when the affective content of the language and the facial displays agreed. This study is of the second type: we played videos where the facial displays either agreed or disagreed with the user-model evaluation expressed in the text and asked for subjective preferences.

Based on the results of the previous study, we had the following hypotheses:

1. In a negative context, participants will prefer negative user-model displays over neutral or positive displays.

2. In a positive context, participants will prefer positive and neutral displays over negative displays.

3. There will be no significant preference between positive and neutral displays in either context.

7.4.1 Participants

Again, this experiment was run over the web and was advertised through email and the Language Experiments Portal, and again participants were entered in a prize draw for gift certificates. The details of the participants in this study were as follows:

Total number	18		
Gender	Female: 8	Male: 10	
Age	Under 20: 1	20–29: 14	30 and over: 3
Computer experience	Beginner: 2	Middle: 5	Expert: 11
Native language	English: 8	Spanish: 3	Other: 7

7.4.2 Methodology

Each subject was shown 12 pairs of videos. Both videos in a pair had identical speech content, but the face-display schedules differed: each trial included two of the three possible types of facial displays (positive, negative, or nodding only). For each trial, the subject was asked to indicate which video they preferred; subjects were encouraged to go with their first instincts and were not otherwise instructed on the selection criteria. All subjects saw videos of the same 12 sentences, in an individually-chosen random order. Six of the sentences suggested a positive evaluation, while the other six indicated a negative evaluation. The trials were balanced so that a subject made each pairwise comparison between schedules twice per sentence type (positive or negative), once in each order, while the allocation of comparisons to items was made randomly for each subject. The full interface and instructions for this experiment are given in Appendix C.4.

7.4.3 Materials

To create the materials for this experiment, we generated a further 12 sentences from the text planner, again based on neutral user preferences. We then created a positive version of six of the sentences and a negative version of the other six by prepending either *You will like this* or *You will not like this*. For each sentence, we then created three facial-display schedules, again using the rule-based strategy described in Section 7.1: positive, negative, and neutral facial displays; we also added a nod on the initial *this* in all cases. We then created videos of each version of each sentence, using the RUTH head. Figure 7.5 shows the three schedules

Figure 7.5: Sample schedules from the consistency study

	You	*will*	*like*	*this:*	*it*	*is*	*classic.*
Positive				nd=d			tn=r, bw=u
Negative				nd=d			ln=l, bw=d, sq
Neutral				nd=d			nd=d

generated for one of the positive sentences in this study; the labels for the displays are the same as in those in Figure 7.3.

7.4.4 Results

All 18 subjects in this experiment gave responses on all of the items, so there were 36 judgements on each pairwise comparison between video versions. No demographic factor had any influence on these results.

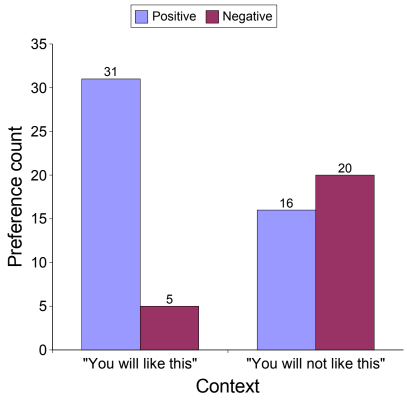
The results of this study are shown in Fig. 7.6. On each graph, the bars on the left indicate the choices made in a positive context, while the right-hand bars show the choices made in a negative context. For example, when comparing positive facial displays against negative displays (Fig. 7.6(a)), subjects preferred the positive displays in a positive context 31 out of 36 times, while in a negative context they preferred the negative displays 20 times out of 36. There is a clear pattern: in a negative context, subjects generally preferred the less positive facial displays (negative over positive; negative over nodding; nodding over positive), while in a positive context these preferences were all reversed.

The only preferences that were individually significant on a binomial test were those for positive displays over each of the others in a positive context (both $p < 0.05$). Using a χ^2 test, the preference between schedules using positive and negative displays (Fig. 7.6(a)) was found to be significantly different in the two contexts ($\chi^2 = 12.0$, $p < 0.001$); for the positive-nodding choice (Fig. 7.6(b)), the context had a marginally significant impact ($\chi^2 = 3.63$, $p \approx 0.06$); while for the nodding-negative choice (Fig. 7.6(c)), the effect was further from being significant ($\chi^2 = 2.72$, $p \approx 0.1$).

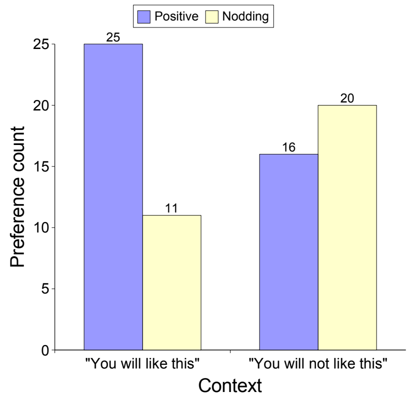
7.4.5 Discussion

The results of this experiment partially confirm our hypotheses. Regarding the first two hypotheses, there were preference for negative displays in a negative context and positive displays in a positive context, but not many of the preferences were significant. As for the third hypothesis, there was actually a significant preference for positive displays over neutral displays in a positive context, which was not predicted. However, although not many of the

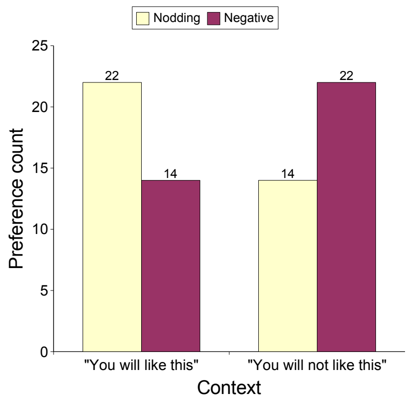
Figure 7.6: Choice counts for the consistency study



(a) Positive vs. negative



(b) Positive vs. nodding



(c) Nodding vs. negative

individual preferences were significant, the context did have a significant effect on the choices that were made: for all pairwise choices, the more negative motions were preferred in negative contexts, while the more positive motions were preferred in positive contexts.

Unlike in the previous study, in this case participants showed a near-significant difference in their responses to positive and neutral facial displays. This suggest that, although the neutral displays appear to be positive when presented in isolation, when they are contrasted with the characteristic positive displays, the difference becomes apparent. As well, there appears to be a general preference across all of the trials for the positive facial displays: the strongest preferences are those for the positive displays over the others, while adding a negative context only reduces the preference to just under 50%.

7.5 Summary

In this chapter, we have described a simple rule-based method for selecting facial displays based on the characteristic motions made by the corpus speaker in positive, negative, and neutral user-model contexts. We then presented two user evaluations designed to measure whether participants are able to correctly identify the intended user-model evaluation based on the facial displays selected by the rules. In the first experiment, participants were reliably able to identify positive and negative evaluations based on the facial displays; however, they tended to treat displays intended to be neutral as positive, and also had a weaker tendency to identify videos with no motion other than lip-synchronisation as negative. In the second study, participants generally preferred schedules with more positive displays to accompany sentences with positive evaluations, and those with more negative evaluations to accompany negative sentences. In this second study, participants also gave different responses for positive and neutral facial displays when they were presented as forced-choice minimal pairs in the two contexts.

The results of both of these studies indicate that the characteristic positive and negative facial displays of the recorded speaker were identifiable when resynthesised on the virtual agent; the results for the two display types expected to be neutral (nodding and no motion) were more mixed. In the context of the thesis, these results provide more evidence to address the first research question regarding the impact of variation on generated output. In this case, we held the spoken content constant and varied the facial displays that accompanied it; and, just as in the experiments in Chapter 4, the participants in these studies were also generally able to identify the intended effect. In the second study, they also preferred the videos with consistent content on the two output channels to those where the content was inconsistent.

The result of this study also largely agree with the findings from Section 4.4 on the conditions where the participants were not able to identify the tailoring. In the first experiment here, just as in the previous study, participants did not make a distinction between output intended to be positive and output intended to be neutral. In the second experiment, when the materials were presented side by side, there was a difference in the responses on positive and neutral facial displays, but the distinction between positive and negative displays was stronger.

In addition to addressing the first research question directly, the results in this chapter contribute to the goals of the thesis in one other important way: they confirm that the most characteristic behaviours of the corpus speaker—namely, the facial displays he used in positive and negative contexts—are identifiable to human judges. In combination with the findings on the overall corpus patterns from the previous chapter, this means that we can be more confident that, when we use the data from this corpus as the basis for selecting displays in the next chapter, the resulting facial displays are based on a reliable source.

Chapter 8

Corpus-based generation of facial displays*

Imitation is the sincerest flattery.

C. C. Colton, *The Lacon*

THE STUDIES in the preceding chapter used a simple rule-based implementation of facial-display selection based on the patterns in the corpus to determine whether participants could recognise the intended user-model evaluation based on characteristic facial displays. That implementation made use of the corpus data in much the same way as Reiter and Dale (2000) described for natural language generation: it used rules derived from patterns in the corpus, but did not itself make direct use of the data represented in the corpus. The results of those studies showed that the most characteristic motions of the speaker were identifiable when reproduced on the talking head; however, the synthetic motions produced by the system that was evaluated are very different from the actual display combinations used by the speaker, who employed a much wider range of displays.

In this chapter, we describe a second set of implementations that make direct use of the corpus data to select facial displays to accompany the generated text; this is more similar to the more recent corpus-based generation methods described in Section 2.2.1 and is also similar to the embodied-agent implementations of Kipp (2004) and Stone *et al.* (2004) described in Section 2.4.2. The implementations provide two different strategies for choosing the facial displays in a given context: choosing the majority corpus option at all times, or making a weighted random choice among all of the corpus possibilities.

*This chapter is partly based on Foster and Oberlander (2007).

To evaluate these implementations, we used several methods to compare the schedules generated by both of them against each other, against the original schedules from the corpus, and against the schedules created by the rule-based implementation from Chapter 7. First, we computed a range of automated evaluation metrics: several that assess corpus similarity, as well as a set of metrics that compute the range and distribution of facial displays included in the schedules. We then asked human judges to compare the quality of the schedules produced by each method: first comparing the two corpus-based strategy against each other, and then comparing the higher-rated corpus-based strategy against the rule-based strategy from the previous chapter.

This chapter addresses all three of the research questions of this thesis. First, the results of all of the human evaluations contribute once again to our knowledge of the impact of several different types of variation on both automated and human-judgement metrics. The evaluations also directly compare three different strategies for selecting body language for an embodied agent, all of which are based on the corpus data but each of which uses that data in a different way: simple corpus-derived rules, corpus-based majority choice, and corpus-based weighted choice; this study directly addresses the second question (comparing methods of making selections based on corpus data). Finally, as in Chapter 5, here again we evaluate the same generated outputs using both human judgements and a range of automated evaluation metrics; this provides additional evidence of the relationship between human and automated evaluations of generated output.

This chapter is arranged as follows. In Section 8.1, we begin by describing the two strategies we used for making face-display selections based directly on the corpus data. Section 8.2 then presents several automated metrics that were used to compare the output generated by the two corpus-based strategies to the data in the corpus and to the output generated by the rule-based strategy from the previous chapter, and describes the results of each of the metrics. We then describe two human evaluations studies directly examining the quality of the generated output: in Section 8.3, we compare the schedules generated by the two corpus-based strategies to each other, while in Section 8.4 we compare the schedules generated by the highest-scoring corpus-based strategies to those generated by the rule-based implementation.

8.1 Implementation: Corpus-based generation

In the preceding chapter, we described a rule-based method for selecting facial displays based on the derivation tree. In this chapter, we use the data in the full corpus to select displays for the talking head as follows. Based on the analysis of the corpus patterns presented in Appendix F, we selected the following node features to use in selecting facial displays: the user-preference evaluation, the clause of the sentence, the pitch accent, and the surface string associated with the node (with semantic-class replacement—e.g., *classic* becomes *style*).

Like the rule-based implementation described in Section 7.1, this process also starts with the annotated OpenCCG derivation tree created by the text planner; the nodes in this tree include all of the features listed above. The algorithm then proceeds depth-first down the tree, choosing a set of displays for each node as it is encountered. To choose a display set for a given node, we consider the set of display combinations on all nodes with the same contextual information and choose a display in one of two ways: selecting the majority class in all cases, or making a choice among all of the combinations, weighted by the relative frequency.

As a concrete example of the two generation strategies, consider a hypothetical context in which the speaker made no motion 80% of the time, a downward nod 10% of the time, and a downward nod with a brow raise the other 10% of the time. For such nodes, the majority generation strategy would choose the majority class of no motion 100% of the time, while the weighted strategy would choose nothing with probability 0.8, a downward nod with probability 0.1, and a nod with a brow raise with probability 0.1. In cases where there is a tie for the majority class, the majority-choice strategy chooses one of the possibilities arbitrarily.

The average number of nodes in the corpus with the same contextual information ranges from 1 (522 instances, mostly very specific syntactic structures) to 134 (*design* or *colour scheme*[1] in a neutral context), with a mean size of 7.31. Within a context, the majority class had an average probability of 0.916, with a maximum of 1.0 and a minimum of 0.23. Of the context classes that occurred more than once, the most frequent option was no display at all 88.1% of the time, a downward nod on its own 7.1% of the time, with the remaining 4.8% divided among a number of other display combinations. 90.6% of the contexts that occurred only once had no associated motion.

Several practical factors were taken into account when making these choices. First, in the cases where no nodes with a matching context were found—primarily the single-item classes, with long and specific syntactic content—both strategies always chose to make no motion, as that was the majority motion in nearly all such single-node contexts. Secondly, the displays in the corpus that were associated with non-standard constituents such as *the tiles are* (and were therefore attached to more than one node in the tree) are treated as a sequence of identical annotations: a display attached to multiple nodes is considered the same as an identical display on each of the individual nodes.

We also ensure while generating that, if a particular motion class is selected for a parent node, no motion of that class can be specified on any of the child nodes. For example, if a right turn is selected for an internal node covering a span such as *on the decorative tiles*, no turning motions

[1] These words have the same semantic class (*mental-obj*) in the grammar.

Figure 8.1: Face-display schedules for a sample sentence

	Although	*it's*	*in*	*the*	*family*	*style,*	*the*	*tiles*	*are*	*by*	*Alessi.*
C	nd=d	nd=d	nd=d		nd=d				nd=d,bw=u		
	....	....	....	....	....	.... ln=l	....	....	....	....	....
M					nd=d						nd=d
W	nd=d				nd=d		. . tn=r . .				
R					ln=l,bw=d						tn=r,bw=u

of any sort can be selected on any of its child nodes. If the generation process proposes such a conflicting motion combination, the conflicting parts are removed; in the example case, if the algorithm proposes a right turn combined with a brow raise on the node *decorative tiles*, it is transformed into a brow raise on its own.

Finally, we explicitly do not select any motions on words for which the speech-synthesiser output is very short such as *but* and *is*: even though the recorded speaker often made motions on words such as these, it was not possible for the synthetic voice to make those words long enough to make motions sensible, so videos generated from such schedules tend to be very unnatural-looking.

Figure 8.1 shows a sample sentence from the corpus, the original displays annotated for it in the corpus (C), the displays that were selected for it by each of the corpus-based strategies (M and W), and the displays that were specified by the rule-based strategy from Chapter 7 (R). Note that the original annotations include a left lean covering the entire sentence, in addition to the nods and other motions on specific words.

8.2 Automated evaluation: Corpus similarity and display range

As a first evaluation of the facial-display schedules generated by the different strategies, we computed several different automated metrics. First, we compared the generated schedules against the corpus data in three ways: by measuring precision, recall and F score; by computing the node accuracy (i.e., the percentage of nodes with completely correct displays, including correctly selecting no motion); and by using the β agreement metric described in Section 6.3.3. Precision, recall, F score, and node accuracy measure the proportion of the generated displays that exactly match the corpus data, while, as described in Section 6.3.3, β allows for partial agreement if displays overlap.

As described in Section 2.6.2, several previous studies have shown that evaluation metrics based on corpus similarity are not necessarily the best predictors of human preferences for generated output; this finding has already been confirmed by the results of the studies in Chapter 5. In addition to comparing the generated schedules against the corpus data, therefore, we also measured several other features of the schedules to see if they agree better with human judgements: the rate of facial displays used per sentence, the range of different displays generated per sentence and in the entire corpus, and the uniformity of the display-combination distribution. These metrics were also computed on the original corpus sentences. Later in this chapter, we describe the results of human evaluations comparing schedules generated by all of these strategies and compare those results to the results of these metrics.

8.2.1 Methodology

To generate the schedules to be used for the evaluations, we divided the corpus into 10 equal-sized segments at random. For each segment, the counts of face-display combinations in each context were gathered using the other 90% of the corpus; these probabilities were then used to create display schedules for each of the sentences in the held-out 10%, using both of the corpus-based strategies described in the preceding section. We also generated schedules for all of the sentences using the rule-based strategy presented in Section 7.1; as that strategy does not use the corpus data directly, these schedules were created by simply running the generation procedure on the entire set of sentences in the corpus. These generated face-display schedules were then used to compute a range of automated metrics, as follows.

8.2.1.1 Corpus similarity metrics

The first set of automated metrics compared the generated schedules sentence-by-sentence against the schedules in the corpus. For recall, we counted the proportion of the corpus displays for a sentence that were reproduced exactly in the generated output, while for precision we counted the proportion of generated displays that had exact matches in the corpus. The F score for a sentence was then computed by taking the harmonic mean of these two values. We also computed a score for *node accuracy*: the proportion of nodes in the derivation tree where the proposed displays were correct, including those nodes where the algorithm correctly proposed no motion. Note that a baseline system that never proposes any motion scores 0.79 on this measure. Overall scores for all of these measures were obtained by averaging the sentence-level scores across the corpus. We also computed a value for β as described in Section 6.3.3, comparing the full set of generated sentences for a strategy against the full set of corpus sentences.

8.2.1.2 Sentence-level metrics

The metrics described above compared the displays selected for a sentence against the actual displays found in the corpus for that sentence. We also computed two sentence-level metrics directly on the schedules without any reference to the corpus data; since these metrics do not require the corpus data for comparison, we also computed them on the original sentences.

We counted two features of each sentence: the total number of facial-display combinations (i.e., the number of display *tokens*), and the number of different combinations (i.e., the number of display *types*). If more than one display was associated with the same span of nodes (e.g., the nod and brow raise on *are* in the corpus schedule in Figure 8.1), that was counted as a single display; overlapping displays were counted as individual displays. These metrics provide an indication of how much motion there is per sentence and how many different displays were used. To get an overall score, we averaged the values across the sentences.

8.2.1.3 Global metrics

All of the metrics in the preceding two sections were based on the displays for individual sentences, and the overall score for a set of sentences was obtained by averaging the scores across all of the sentences. In a final set of metrics, we considered the distribution of facial displays across the entire set of sentences. First, we counted the number of different display combinations that were used on all of the sentences, again considering displays covering the same set of nodes as simultaneous. We then counted the number of times that each display combination occurred across the whole set of sentences, and computed the mean and standard deviation of these counts. These metrics provide an indication of the range of displays selected across the entire set of sentences and how equally they are distributed. Again, as these metrics do not require the corpus data for comparison, we computed them on the original sentences in addition to the schedules generated by all of the strategies.

8.2.2 Results

This section presents the output of all of the above evaluation metrics across the full set of 444 sentences in the corpus.

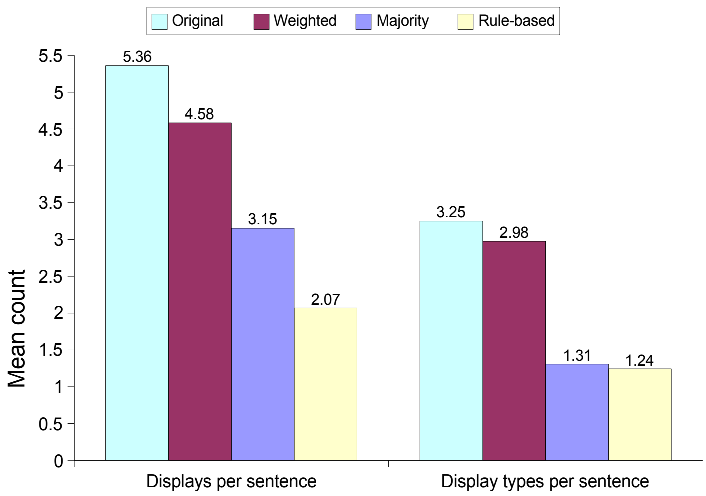
8.2.2.1 Corpus similarity metrics

Figure 8.2 shows the results for all of these corpus-similarity measures, averaged across the sentences in the corpus. The majority strategy scored uniformly higher than the weighted strategy, which in turn scored higher than the rule-based strategy on all measures except for node accuracy. The difference was particularly dramatic for precision, where the value for the majority strategy (0.52) was nearly twice that for the weighted strategy (0.29); that is, the

Figure 8.2: Results for the corpus-similarity measures, averaged across sentences

	Majority					Weighted					Rule-based				
	Prec	*Rec*	*F*	*NAcc*	*Beta*	*Prec*	*Rec*	*F*	*NAcc*	*Beta*	*Prec*	*Rec*	*F*	*NAcc*	*Beta*
Mean	0.52	0.31	0.18	0.82	0.34	0.29	0.24	0.12	0.75	0.23	0.22	0.10	0.06	0.77	0.14
Min	0.0	0.0	0.0	0.56	–	0.0	0.0	0.0	0.40	–	0.0	0.0	0.0	0.52	–
Max	1.0	1.0	0.5	1.0	–	1.0	1.0	0.4	0.95	–	1.0	1.0	0.5	1.0	–
Stdev	0.32	0.22	0.12	0.08	–	0.25	0.20	0.10	0.09	–	0.35	0.17	0.10	0.08	–

Figure 8.3: Sentence-level metrics



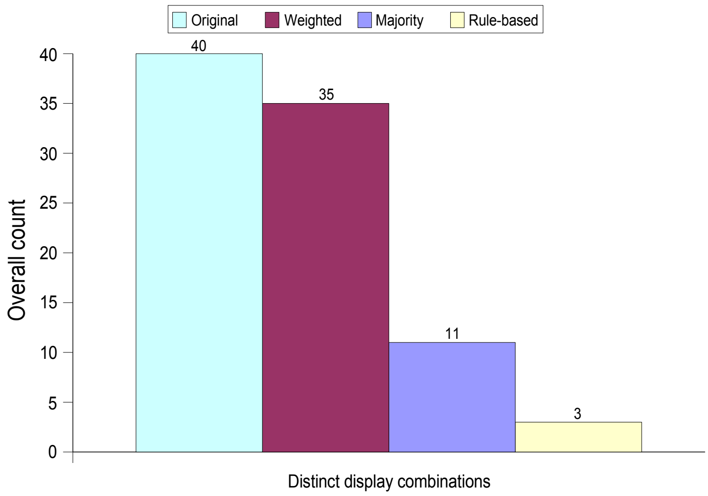
motions proposed by majority strategy were identical to the corpus nearly twice as often. Also, the recall for the rule-based strategy was particularly low.

Using a T test, all of the differences on mean precision, recall, and node accuracy are significant at $p < 0.001$; also, the node accuracy score for the majority strategy is significantly better than the no-motion baseline of 0.79, while those for the weighted and rule-based strategies are significantly worse ($p < 0.01$). Significance cannot be assessed for the differences in the F scores or β values, but the trend is the same.

8.2.2.2 Sentence-level metrics

The pattern is somewhat different for the sentence-level metrics than for the corpus similarity metrics, as shown in Figure 8.3. Recall that we computed these metrics on the original corpus sentences as well as on the output of all of the strategies, so the results for each of the generation strategies can be compared against those for the corpus. The original corpus had both the most displays per sentence and the most different display types per sentence; the values for weighted choice were a fairly close second, those for majority choice third, and the rule-based

Figure 8.4: Total distinct face-display combinations



strategy scored lowest on both of these metrics. Except for the difference between majority choice and rule-based choice on the facial-display types, which is not significant, all of the other differences between the strategies are significant at $p < 0.0001$ on a paired T test.

8.2.2.3 Global metrics

Figures 8.4–8.6 show the results for the metrics that address the range of display combinations used across the entire corpus. Figure 8.4 shows the total number of different display combinations included in the corpus and those generated by each of the generation strategies. The original sentences used 40 different display combinations, while—by design—the sentences generated by the rule-based strategy used only 3; the numbers for the two corpus-based generation strategies fell between these two extremes. Figure 8.5 shows the frequency of all of the display combinations used, in log-10 space: so, for example, the most frequent class for the original sentences and both of the corpus-based strategies was used about 10^3 times in each case; this class was downward nodding in all three cases. The distribution of class sizes is similar for the original sentences and the output of the weighted strategy, while the class size for the majority-choice strategy drops off much more steeply. The rule-based strategy only uses three different classes, so its curve is very short, but it appears to be closer to the original and weighted curves than to the majority one.

Figure 8.6 shows the mean and standard deviation of the class sizes—that is, how many times each display was used. The mean number of occurrences of each display combination (the leftmost bars in Figure 8.6) are essentially the reciprocal of the number of combinations used (Figure 8.4): in this case, the rule-based strategy has a much higher average rate of

Figure 8.5: Distribution of display-combination frequencies

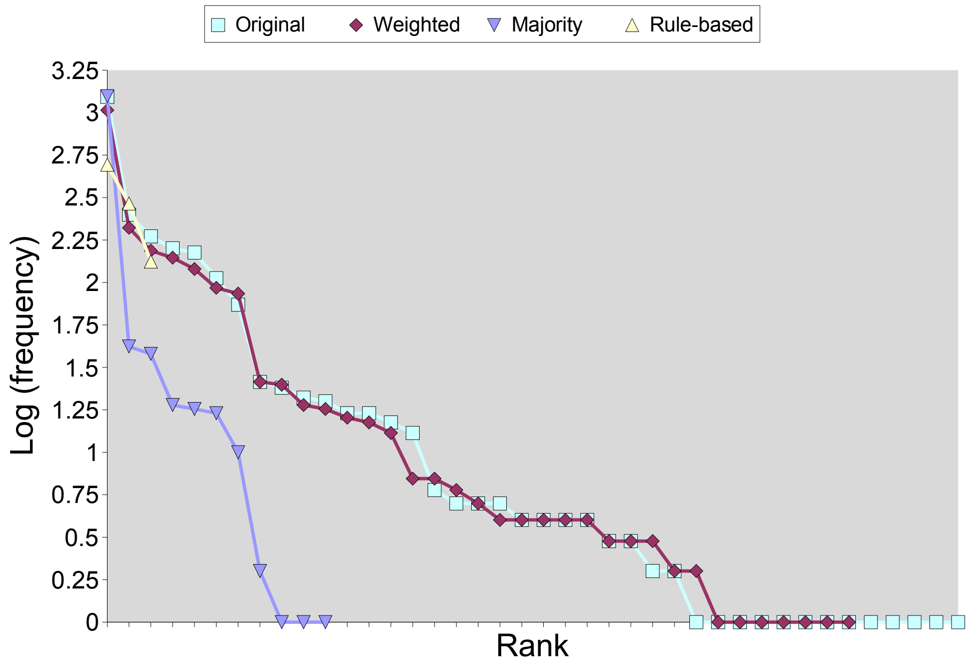
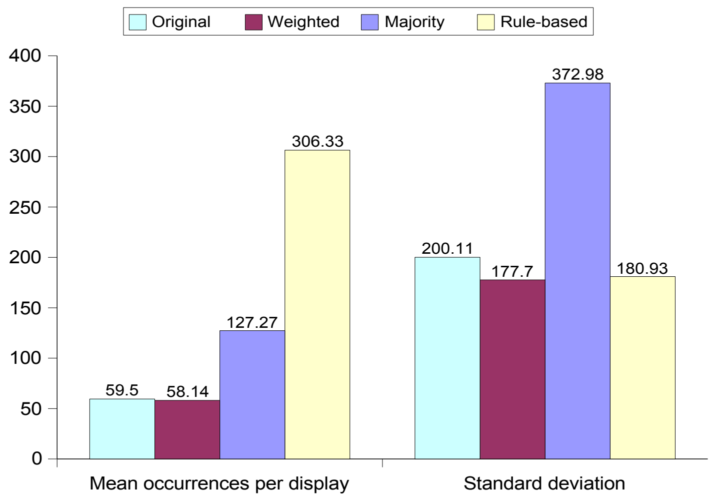


Figure 8.6: Mean and standard deviation of display-combination frequencies



occurrences per class, the majority-choice strategy somewhat fewer, while the values for the weighted-choice strategy and the original sentences are lower still. More interesting are the results for the standard deviation of the counts: the values for the rule-based, weighted, and corpus strategies are similar, but the value for the majority strategy is much larger.

8.2.3 Discussion

On the corpus-based similarity metrics, the schedules generated by the majority-choice strategy scored the highest in all cases, with the weighted-choice schedules second and the rule-based schedules third. These results are not surprising: the majority-choice strategy is designed to choose the highest-probability option in all cases, so its outputs are likely to be closer to the corpus sentences than those generated by the weighted-choice strategy, which deliberately chooses options other than the highest-probability one. Similarly, the rule used in the rule-based strategy is based on the most *distinctive* motions of the speaker, not the most *frequent* motions, so again it is not surprising that the outputs of this strategy tend to diverge the most from the corpus examples.

On the other metrics, the values for the original corpus sentences are the highest, but the results of the weighted strategy are fairly close: the weighted strategy used more and more varied displays per sentence, generated a wider range of display combinations, and distributed the displays much more equally across the sentences. This is a reflection of the fact that, in most contexts, the majority option in the corpus is either no motion or a downward nod, so taking the highest-probability option results in fewer and less varied motions. In fact, the most frequent display in the majority schedules—nodding—was used a total of 1251 times across the full set of sentences, while the total number of occurrences of the other 10 display combinations was just 149. The original corpus schedules still scored higher than the weighted schedules on both metrics, indicating that the original speaker used an even greater repertoire of motions than those generated by weighted choice.

The rule-based schedules scored the lowest on the metrics that assess the number and range of expressions generated, but the variability in the full set of rule-based schedules (as measured by the standard deviation of the class sizes) was similar to that of the weighted schedules and the original corpus sentences; on this metric, it is the value for the majority-choice strategy that is different from the others. As mentioned above, this is because the majority-choice strategy chooses a nod almost 90% of the time that it specifies a display, while the other strategies have a more even distribution.

8.3 Human evaluation: Comparing data-driven strategies

On the automated metrics that compared each sentence against the corpus, the face-display schedules generated by the majority strategy scored above those generated by the weighted strategy on all of the metrics. However, on the metrics that computed properties of the sentences themselves, the results of the weighted-choice schedules were quite similar to those of the original sentences, while the values for majority-choice schedules were much further. If we consider the original schedules to be the target "gold standard" output for this task, then these automated results are sending conflicting signals: some metrics suggest that the majority schedules are better than the weighted schedules (i.e., they are more similar to the corpus examples), while others suggest the reverse.

In this experiment, we addressed this conflict through an evaluation in which human judges compared videos of three different schedules for a range of sentences: resynthesised versions of the original sentences, as well as schedules created by both of the corpus-based generation strategies. The relative rankings of the two generation strategies gives an indication as to which of the automated predictions agree with human preferences.

Based on previous comparisons of corpus-based and human evaluations of generated output (Stent *et al.*, 2005; Belz and Reiter, 2006), as well as the results of the study in Chapter 5, we had the following hypotheses for this experiment:

1. The weighted-choice schedules will be preferred over the majority-choice schedules.
2. The original schedules will be preferred over the output of either of the corpus-based generation strategies.

8.3.1 Participants

As in most of the previous experiments, this experiment was also run through a web page, and participants were also recruited via an email to the Informatics departmental student mailing list and by posting the experiment at the Web Experiments Portal; participants were also entered into a prize draw. The details of the participants who took part in this experiment are as follows:

Total number	56			
Gender	Female: 22	Male: 34		
Age	Under 20: 5	20–29: 39	30 and over: 12	
Computer experience	Beginner: 3	Middle: 22	Expert: 31	
Native language	English: 24	German: 9	Spanish: 5	Other: 18

8.3.2 Methodology

This experiment used the same basic methodology as the study in Section 7.4: again, participants saw videos of two possible facial displays accompanying the same synthesised speech and were asked to select which of the two versions they preferred. In this case, there were 24 sentences in total, with three videos for each sentence: the schedule specified by each of the majority-choice and weighted-choice strategies, as well as the original schedule. All participants saw the same 24 sentences, in an individually randomly-selected order. The versions were balanced so that each participant made each of the three pairwise comparisons between schedules eight times, four times in each order; the allocation of comparisons was made randomly for each participant. The details of the interface and instructions for this experiment are given in Appendix C.5.

8.3.3 Materials

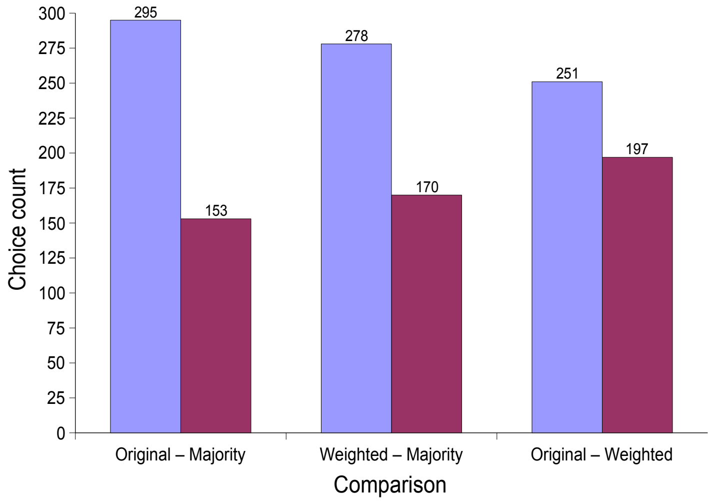
To create the materials for this study, we randomly selected 24 sentences from the corpus and generated RUTH videos of three face-display schedules for each sentence: the schedules generated by the majority-choice and weighted-choice strategies in the cross-validation study from the previous section, as well as the original corpus annotations. The original schedules were modified to remove motions on short words such as *but* and *is*, for the reasons discussed at the end of Section 8.1. The RUTH videos were generated as described in Section 7.2.

The results of the automated metrics on the schedules chosen for the study are generally in line with the overall results presented in Section 8.2. The original schedules have an average of 5.45 total displays per sentence (3.17 display types) and use 16 different display combinations in total, the weighted-choice schedules have 4.21 displays per sentence (2.67 types) and also use 16 different combinations, while the majority-choice schedules have 2.83 displays per sentence (1.17 types) and use 4 different combinations. The mean F score across the selected sentences is 0.31 for the weighted-choice schedules and 0.41 for the majority-choice schedules.

8.3.4 Results

All participants in this experiment gave responses for all of the items, yielding 448 responses for each pairwise comparison. The overall results of this study are shown in Figure 8.7. Participants chose the original schedules and the schedules generated by the weighted strategy over those generated by the majority strategy (66% and 62%, respectively; $p < 0.0001$ on a binomial test for both). They also showed a somewhat weaker preference for the corpus schedules

Figure 8.7: Human preferences among corpus-based strategies



over those generated by the weighted strategy (56%; $p < 0.05$). None of the demographic factors had an influence on these results.

8.3.5 Discussion

The results of this study confirm both of our hypotheses: the weighted-choice schedules were indeed preferred to the majority-choice schedules, and the original schedules were preferred to both. This agrees with both the results of previous studies and with the findings from Chapter 5: we now have more evidence that metrics of corpus-based similarity are not good predictors of human preferences for the output of a generation system that incorporates variation. Also as in Chapter 5, the preferences of the human judges agreed with the result of metrics that directly assess the range of different possibilities included in the output.

The human judges in this study preferred the original sentences to those generated by either strategy, although the preference over the weighted strategy was less pronounced. This indicates that, while the weighted generation strategy produces more highly-rated outputs than the majority strategy, its output is still perceptibly worse than the data in the corpus. This provides further evidence that the metrics that compute the number and range of displays per sentence and the range of displays across the full set of sentences are good predictors of human preferences: all of these metrics scored the original corpus schedules somewhat higher than the weighted-choice schedules with the majority-choice schedules much further down, which agrees with the results of this study.

8.4 Human evaluation: Data-driven vs. rule-based strategies

The previous study compared human preferences for the schedules generated by the two corpus-based strategies and showed that participants strongly disliked the schedules generated by the majority-choice strategy. In this experiment, we turn our attention to the schedules generated by the rule-based strategy from Chapter 7. On nearly all of the automated metrics from Section 8.2, the rule-based strategy scored much lower than either of the corpus-based strategies: its output was less similar to the original data when measured by sentence-level similarity, it used a much smaller range of displays overall, and also included fewer displays per sentence than any of the other strategies. The only metric where this strategy was at all comparable to the others is the standard deviation of the distribution of displays; that is, the distribution of displays generated by this strategy was in some ways similar to the distributions of the original sentences and the weighted strategy.

In this second study, we again gathered user preferences on three pairwise comparisons of schedules. In this case, though, the three versions were the original schedule, the schedule generated by the weighted strategy, and the schedule generated by the rule-based strategy. Based on both the automated metrics and the results of the preceding experiment, we had the following two hypotheses for this experiment:

1. The preferences between both the original and weighted schedules and the rule-based schedules will be the same as the preferences between these two schedules and the majority-choice schedules.

2. The pairwise preference between the original and weighted schedules will be the same as in the previous study.

8.4.1 Participants

This experiment was also run through a web page, and participants were also recruited via email to the department student mailing list and through the Web Experiments Portal; participants were also entered into a prize draw. The details of the participants are as follows:

Total number	36		
Gender	Female: 20	Male: 16	
Age	Under 20: 4	20–29: 23	30 and over: 9
Computer experience	Beginner: 1	Middle: 14	Expert: 21
Native language	English: 18	Chinese: 4	Other: 14

We also asked participants whether they had taken part in in any of the previous talking-head judgement experiments (Sections 7.3, 7.4, or 8.3): two had done at least one previous study.

8.4.2 Methodology

The methodology for this experiment was identical to the previous experiment, except for two factors. First, instead of majority-choice, the third schedule option was the schedule generated by the rule-based strategy; second, participants judged 18 sentences instead of 24, meaning that they made each pairwise comparison six times instead of eight. The details of the instructions and interface are given in Appendix C.5.

8.4.3 Materials

For this study, we randomly selected a new set of 18 sentences from the corpus, ensuring that none of the sentences that were used in the earlier study were included. We generated RUTH videos of the following three schedules for each: the original annotations (modified as in the preceding study to remove motions on short words), the weighted-choice schedule, and the rule-based schedule.

As in Section 8.3.3, the results of all of the automated metrics on the schedules chosen for this study are generally in line with the overall results presented in Section 8.2. The original schedules have an average of 4.83 total displays per sentence (3.17 display types) and use 15 different display combinations in total, the weighted-choice schedules have 5.06 displays per sentence (3.0 types) and use 12 different combinations, while the rule-based schedules have 1.78 displays per sentence (1.17 types) and use 3 different combinations. The mean F score across the selected sentences is 0.40 for the weighted-choice schedules and 0.13 for the rule-based schedules.

8.4.4 Results

Not all of the participants responded to all of the items for this study, so there were a total of 648 judgements: 216 comparing original to rule-based, 217 for weighted vs. rule-based, and 215 for original vs. weighted. The results for this study are presented in Figure 8.8. In these results, there is a mildly significant preference for both the original schedules over both the weighted and the rule-based ones (57% in both cases; $p < 0.05$), and a trend to prefer the weighted schedules over the rule-based ones (55%; $p \approx 0.14$). None of the demographic factors affected these results.

To compare these results to the choices made in the preceding study, we used a χ^2 test on each of the individual pairwise comparisons: that is, we compared the choices made in this study with the corresponding choices made in the preceding study, replacing majority-choice schedules with rule-based schedules. Figure 8.9 shows the data that was used for the χ^2 comparison. The results are as follows:

Figure 8.8: Human preferences between rule-based and weighted strategies

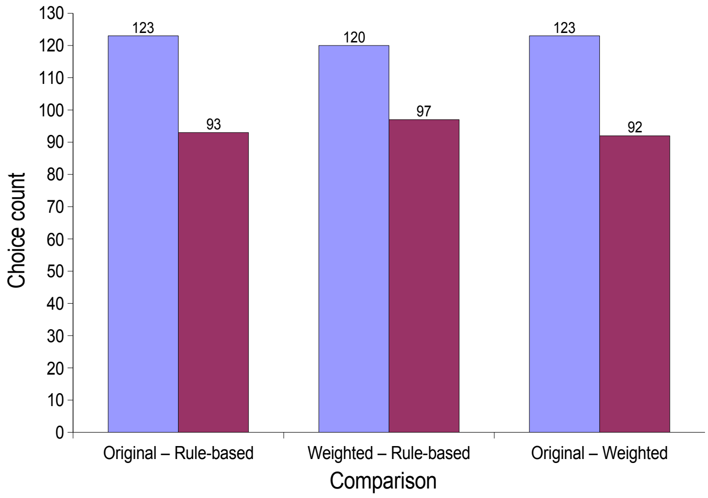


Figure 8.9: Tables for χ^2 comparisons

	Original	***Other***	**Weighted**	***Other***	**Original**	**Weighted**
Other = *Majority*	295	153	278	170	251	197
Other = *Rule-based*	123	93	120	97	123	92

- Original vs. *other*: $\chi^2 = 4.58$, $df = 1$, $p < 0.05$
- Weighted vs. *other*: $\chi^2 = 2.50$, $df = 1$, $p \approx 0.11$
- Original vs. weighted: $\chi^2 = 0.042$, $df = 1$, $p \approx 0.84$

In other words, the judgements on the original vs. weighted comparison were essentially the same in the two studies, the judgements comparing the weighted schedules to the other option (majority or rule-based) were more different but not significantly so, while the difference on the comparisons between the original schedules and the other option is significant.

8.4.5 Discussion

The results of this study contradict our first hypothesis that the results for the rule-based schedules would be the same as for the majority-choice schedules in the preceding study. In this study, the participants showed a tendency to prefer both the weighted and original schedules over the rule-based schedules, but the trend is mildly significant in one case ($p < 0.05$) and does not reach significance in the other ($p \approx 0.14$). This contrasts with the very significant preference ($p < 0.0001$) for weighted and original over majority in the previous study. If we compare the responses from the previous study to the corresponding responses on this study, the responses with the original schedules are significantly different, while the difference comparing against weighted-choice schedules has marginal significance ($p \approx 0.1$).

On the other hand, the second hypothesis is confirmed: participants' responses on the trials comparing original schedules against weighted-choice schedules were similar to the responses in the preceding experiment on corresponding trials. Again, participants expressed a mild preference ($p < 0.05$) in favour of the original schedules. The results of this χ^2 test confirm that the pattern of responses on these trials is essentially identical across the two experiments. This provides evidence that the materials used in this study were similar to those used in the preceding study and that judges were making similar judgements, and suggests that difference in the other responses is likely due to the difference between the majority-choice schedules and the rule-based ones rather than due to some difference in the participants or materials.

In addition to disagreeing with our first hypothesis, the judgements on the rule-based schedules also disagree with the predictions of nearly all of the automated metrics, which generally scored the weighted schedules significantly higher than the rule-based ones. The only metric on which the scores of the rule-based and weighted schedules are comparable is the standard deviation of the class sizes; and on that metric, the result for the majority-choice schedules is much higher. This suggests that user preferences for face-display schedules depend more on the evenness of the distribution of the different facial displays than they do on the overall variety of the displays used.

8.5 Summary

In this chapter, we presented two different methods for making direct use of the corpus data for selecting facial displays for the talking head based on the derivation tree for a sentence and the contextual information from the output planner. One method chooses the single highest-probability option in every context, while the other makes a random choice among all of the options available in a context, weighted by the corpus probabilities. These implementations are representative of the two main methods for selecting body language for embodied agents (Section 2.4.2); they are also comparable to techniques for generating text such as those supported by *p*CRU (Belz, 2006). We compared these methods to each other and to the simple rule-based method from Chapter 7 in two ways: using a range of automated metrics, and by gathering the preferences of human judges.

The majority-choice strategy scored highest on all metrics of corpus similarity, while the scores of the weighted-choice strategy were closest to those of the corpus on all metrics that count the number and range of facial displays selected. The rule-based choice was furthest from the corpus on all automated metrics except for the standard deviation of the face-display frequencies: on that metric, it was actually the majority-choice schedules that diverged the furthest from the corpus. On the human evaluations, the participants preferred the original and weighted-choice schedules very significantly over the majority-choice schedules, and had a mild preference for the original schedules to the weighted-choice schedules. However, there were no significant preferences expressed when the weighed-choice and original schedules are compared to the rule-based schedules.

The results in this chapter address all of the research questions of this thesis. Regarding the first and second questions—the impact of variation on generated output, and the utility of making stochastic choice in generation—they demonstrate again that modifying the generation strategy to choose options other than the highest-scoring one in the corpus produces output that is preferred by judges over output that uses the single highest-probability option. The nature of the corpus and its role in this study both differ from those of the corpus used in Chapter 5. In that study, the corpus was created by hand to represent correct syntactic structures in order to control the overgeneration of the OpenCCG realisation process; the grammar still does most of the work in specifying the content. Here, the corpus is based on annotated human behaviour and is used to select the content to be used, not just to score the content selected by some other process. Despite that difference, in both studies the human judges tended to prefer output which showed more variation and which did not use the majority corpus option in all cases.

In terms of the second research question, although the overall result of this chapter is that making a stochastic choice is better than making a majority choice, the results are not as clear-cut here as they were in Chapter 5. The results of the first human evaluation indicate that making selections that reproduce more of the range of behaviour in the corpus produces results that are preferred by judges to those that reproduce less of the corpus. However, the results of the second study indicate that the selections need not be based directly on the corpus data: participants in that study did not express a significant preference between the schedules generated by weighted random choice and those based on a simple rule-based strategy.

The comparison between the automated evaluations and the human preferences in this chapter contributes to the third question of the thesis, and provides additional evidence to support the findings of the study in Chapter 6. Just as in that study, the predictions of all of the corpus-based similarity metrics disagreed with the actual judgements of the human participants in the experiments; also as in that study, metrics that assess the degree of variability of the output were better predictors of the judgements. This provides further evidence that corpus-based similarity metrics are not suitable to be used in isolation to evaluate the output of generation systems that include variation in their output.

Chapter 9

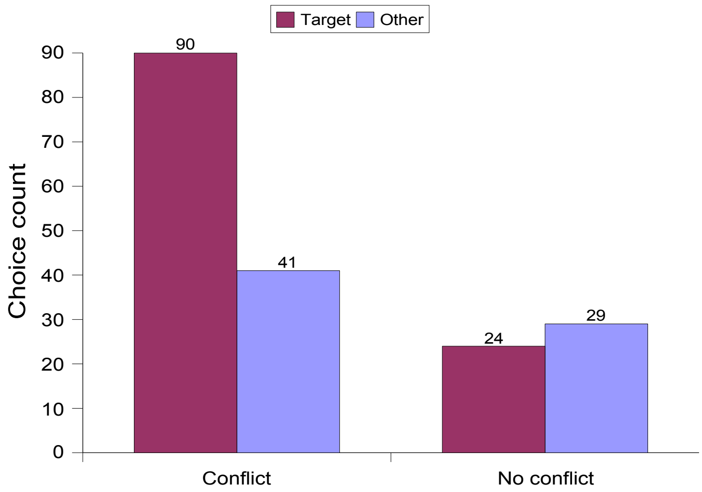
Evaluating facial displays in context

> Perdita thought that not obeying rules was somehow *cool*. Agnes thought that rules like "Don't fall into this huge pit of spikes" were there for a purpose.
>
> Terry Pratchett, *Carpe Jugulum*

THE STUDIES described in the preceding two chapters have demonstrated that human judges were able to recognise the difference between the characteristic positive and negative facial displays made by the speaker recorded for the corpus, and that they strongly preferred generated face-display schedules that made use of more of the available space of motions included in the corpus. However, the results of the final experiment in the previous chapter indicate—somewhat unexpectedly—that the judges did not have a significant subjective preference between schedules generated using a simple corpus-derived rule and those generated using stochastic choice based on the full corpus data.

In all of the talking-head experiments to this point in the thesis, the output has been presented in isolation: participants made judgements based only on videos of isolated sentences. However, the goal of the embodied agent in COMIC, as in many previous systems, is to enhance users' interactions with the overall system. To get the full picture of the impact of different generation strategies, the output must be presented in context. Some previous studies have actually found a difference between the impact of embodied output presented in isolation and presented in context: for example, Rehm and André (2005) found that participants were able to perceive deceptive facial expressions in isolation, but that the displays did not affect participants' behaviour in the context of a task where detecting deception was essential.

Figure 9.1: Results of speech-only user-model study (repeated from Figure 4.6(b))



To add a context to the generated facial displays, we now return to the study from Section 4.4 on the effect of user-model tailoring in language. Recall that participants in that experiment were asked to distinguish descriptions correctly tailored to a given user model from those tailored to the preferences of some other user. The results of that study indicated that participants were able to make this distinction reliably only when the difference between the two descriptions included conflicting concessions to negative preferences. If the choice was between a description intended to be positive and one intended to be neutral, their judgements did not differ significantly from chance. The results for descriptions presented as text and those presented as speech were nearly identical; the results for the speech modality are reproduced in Figure 9.1.

In this chapter, we combine the materials and techniques from this study with the techniques for generating talking-head facial displays from the preceding chapters, with the goal of addressing two issues. First, we test whether the results of the previous speech-based experiment are reproduced when the output is presented by an animated talking head, or whether there is a difference in the participants' responses. Second, we compare participants' responses using two different versions of the talking-head displays: displays selected by the rule-based technique from Chapter 7, and displays selected by the corpus-based weighted-choice technique from Chapter 8.

Like the experiments in the previous chapter, this study again contributes to all three of the research questions of this thesis. In terms of the impact of variation on generated output, we are—as in Chapter 4—measuring participants' ability to detect output that is correctly tailored to the conversational context. We are also making a different sort of direct comparison between rule-based and stochastic methods of selecting behaviour for the talking head, which extends the results of the final study in the preceding chapter and thus contributes more data to the second research question. Finally, this study provides yet another human evaluation comparing two different strategies for adding variation; we can compare the results to the automated measures from the preceding chapter to shed more light on the relationship between the two types of evaluation.

The materials presented to the participants in this evaluation were the same as those presented in the Section 4.4 study, with the addition of the talking head: participants therefore had access to all of the information that the original participants had to make their decisions, plus the additional user-preference information provided by the displays of the talking head. Since both strategies for selecting facial displays are based on the characteristic behaviour of the corpus speaker—which was demonstrated in Chapter 7 to be recognisable—this information should not hurt the participants' ability to detect the user-model tailoring.

For this study, we therefore have the following two hypotheses:

1. Participants' performance with the addition of the talking head will be no worse than their performance with no talking head.
2. The strategy for selecting the talking-head displays will not make a difference to the results.

Note that both of these are essentially null hypotheses: we are predicting no effect on the results of either of the manipulations in this study.

9.1 Participants

Once again, this experiment was also run through a web page, and participants were also recruited via an email to the Informatics departmental student mailing list and by posting the experiment at the Web Experiments Portal; participants were also entered into a prize draw. For this experiment, participants who were native speakers of English were especially encouraged to take part. The details of the participants who took part in this experiment are as follows:

Total number	32		
Gender	Female: 19	Male: 13	
Age	Under 20: 4	20–29: 18	30 and over: 10
Computer experience	Beginner: 2	Middle: 15	Expert: 15
Native language	English: 30	Other: 2	

9.2 Methodology

The methodology for this experiment was essentially the same as that for the experiment described in Section 4.4, but modified to run over the web and to include the talking head. As before, participants observed an eight-turn dialogue between the system and a user with specific likes and dislikes. They were shown two possible versions of each system turn in the dialogue; after viewing both versions of a system turn, they had to respond to the question *Which system output was more appropriate for this user?*

The target user model was shown at all times on the screen and participants were reminded to take it into account when making their choices. The simulated user turns were presented as text on the screen, and the participant clicked a button to play a RUTH video of the two possible system turns. The output was presented as a RUTH video in a pop-up window; there was no textual presentation. The full interface for this study is shown in Appendix C.6.

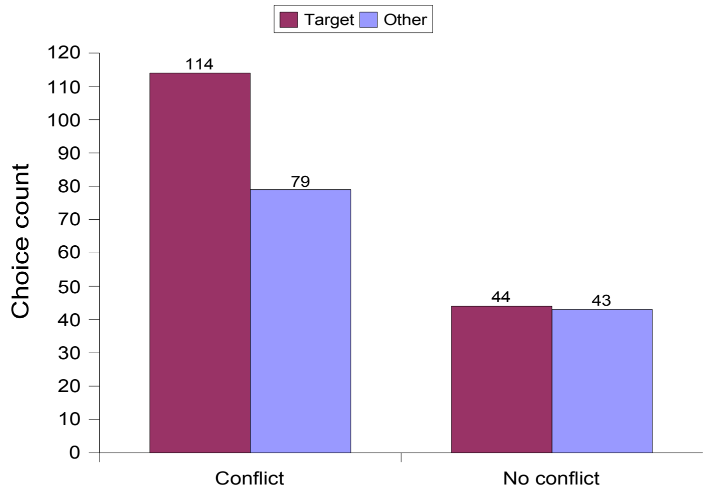
There were four different user models, and each participant was assigned to one of them in rotation. Across the eight system turns in a dialogue, one version in each pair was always tailored to the target model, while the other was tailored to one of the other models or to a neutral model with no likes or dislikes. Each non-target model was used as the "other" model twice in the experiment, once as the first option and once as the second. The assignment of models to turns was made randomly for each participant.

As an additional between-participants factor for this experiment, participants were assigned in rotation to use either videos generated by the weighted-choice strategy, or videos generated by the rule-based strategy. The textual content for both versions of the videos was the same; only the facial displays were different. This factor was held constant for a participant's entire interaction with the system: a participant would see either all weighted videos or all rule-based videos.

9.3 Materials

For this study, we used the same four randomly-generated user models as in the experiment Section 4.4, as well as the same sets of description sequences. To create the videos, we first added all of the necessary contextual information to the derivation trees for the sentences in each description. We then used those trees to generate facial-display schedules using both the rule-based strategy and the corpus-based weighted strategy. Videos were generated for all of the sentences using the RUTH head as described in Section 7.2.

Figure 9.2: Overall results

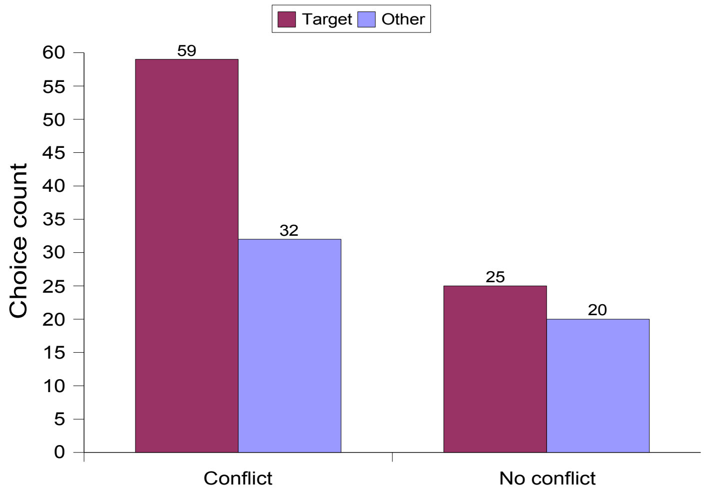


9.4 Results

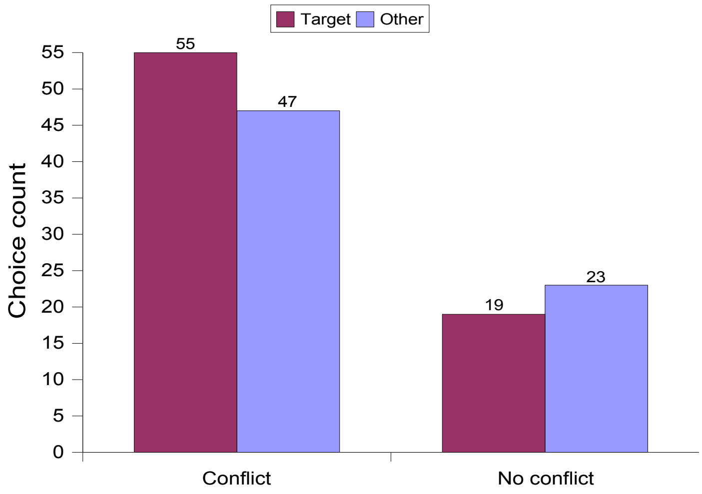
All participants in this experiment gave a response for all of the items. The overall results for all 32 of the participants in this experiment are shown in Figure 9.2. The general shape of these results is the same as those of the speech-only experiment: when the difference between the two versions in a trial included a conflicting concession, participants chose the targeted version significantly more often than the other version (59%; $p < 0.05$); when there was no such conflict, they chose essentially at chance (50%; $p = 1$). None of the demographic factors affected these results. The pattern of responses on the conflicting-concession trials was not significantly different from the responses of participants on such trials in the speech-only condition: $\chi^2 = 2.71$, $df = 1$, $p \approx 0.10$.

If we compare the results for the participants using the weighted schedules with those of the participants using the rule-based schedules, however, there is a difference between these two groups. As shown in Figure 9.3, when there were conflicting concessions, the participants using the rule-based schedules chose the target version over the other version more often (65%; $p < 0.01$) than did the participants using the weighted schedules (54%; $p \approx 0.49$). Both groups still performed essentially at chance on the trials with no conflicting concessions.

Figure 9.3: Results divided by face-display schedule type



(a) Rule-based schedules



(b) Weighted schedules

To assess the significance of this difference, we compared the results from each of the groups of participants on the conflicting-concession trials to the corresponding results of the participants from the speech-only experiment. The results were as follows:

- Rule-based videos: $\chi^2 = 0.21$, $df = 1$, $p = 0.65$
- Weighted videos: $\chi^2 = 4.72$, $df = 1$, $p < 0.05$

In other words, the results for the participants using the rule-based videos were very similar to those of the participants using only speech, while there was a significant difference between the responses of the speech-only participants and those of the participants using the weighted schedules.

9.5 Discussion

We had two hypotheses for this study: that the ability of participants using the talking head to identify correctly tailored output would be no worse than that of the participants who used only speech, and that there would be no difference between the performance of the participants using the two versions of the facial-display schedules. The first of these hypotheses holds: the overall results of the participants using the talking head were not significantly different from the results of the participants who used the speech-only presentation. However, the second hypothesis is not confirmed: the participants using the weighted-choice schedules actually performed significantly worse than the speech-only participants.

If we examine the schedules generated by the weighted strategy for this evaluation, we see that they include a variety of facial displays; sometimes these displays are actually the opposite of what would be selected by the rule-based strategy. The head moves to the right when describing a negative fact in 23 of the 520 schedules, and moves to the left when describing a neutral or positive fact in 20 cases. A description included up to three sentences, and an experimental trial involved comparing two descriptions, so a total of 75 of the trials (52%) for the weighted-choice participants involved at least one of these these potentially misleading displays. There were 38 conflicting-concession trials that had neither of the conflicting head movements, and the performance on these trials was essentially the same as on the full set of trials: the correctly targeted description was chosen 20 times, and the other version 18 times. So the worse performance with the weighted-choice schedules cannot be attributed only to the selected facial displays conflicting with the linguistic content.

Another possibility is that the study participants who used the weighted-choice schedules were distracted by the expressive motions and failed to pay attention to the content of the speech. This appears to have been the case in the COMIC whole-system evaluation (White *et al.*, 2005), for example, where the performance of the male participants on a recall task was significantly worse when an expressive talking head was used. On this study, there was no effect of gender (or any of the other demographic factors) on the pattern of responses; however, it could be that a similar effect occurred in this study for all of the participants.

In Section 8.4, there was no significant preference between the two methods of selecting facial displays, although participants expressed a non-significant preference in favour of the weighted-choice schedules; this generation strategy also outscored the rule-based strategy on all of the automated evaluation measures. However, the results of this study tell a different story: in this more task-based context, neither version of the talking head helped participants to select the correctly tailored descriptions any better than they could with speech alone, and the displays selected by weighted choice actually decreased participants' performance.

In the context of this thesis, the results of this study are relevant to all of the research questions. For the first question, they show that whatever prevented the judges in Section 4.4 from detecting the context tailoring in the output, it was not mitigated by adding facial displays—instead, it was actually made worse in some cases. For the second question, this study adds more information about how rule-based and stochastic generation strategies compare, this time in the context of a more task-based evaluation. Finally, these results extend our findings on evaluation methods by showing that subjective preferences on outputs presented in isolation may not always be sufficient to compare generation strategies: to get the full picture of the performance of a generation system, in some cases it can be necessary to embed the generated output in a task.

Chapter 10

Conclusions

Inigo Montoya: Let me explain. [pause] No, there is too much. Let me sum up.

William Goldman, *The Princess Bride* (screenplay)

As described in Chapter 1, there is a trade-off in the design of any generation system between increased representation and processing effort involved in making dynamic choices on one hand, and the impact on the resulting generation system on the other hand. In this thesis, we have explored the nature of this trade-off through a series of modifications to the output-generating components of a multimodal dialogue system. For each implementation, we performed an evaluation using human judges to see whether the added variation was perceptible to and appreciated by users; in some studies, we also computed a range of automated evaluation metrics no the output.

In this chapter, we summarise the contributions of the thesis as follows. In Section 10.1, we revisit the research question set out in Section 1.1 and combine the results of all of these experiments to propose an answer to each. In Section 10.2, we then discuss how the findings described here based on modifications to the output components of the COMIC multimodal dialogue system are applicable to other similar generation systems. Finally, in Section 10.3, we propose possible extensions to this work to explore further the issues raised in this thesis.

10.1 Research questions revisited

At the start of the thesis, we set out to answer three related research questions, as summarised in Section 1.1. In this section, we summarise the results that address each question.

10.1.1 Evaluating the impact of variation

All of the human evaluations in this thesis addressed this question in some way. The evaluations were of two distinct types. In Chapter 4, Chapter 7, and Chapter 9, the user task was to identify the intended tailoring of the output: either directly by choosing the perceived context from a list, or indirectly by selecting the alternative that was more appropriate to the conversational situation. In the experiments in Chapter 5 and Chapter 8, on the other hand, participants were asked to choose between alternative outputs on the basis of subjective preference.

In almost all of these studies, the added variation had a significant effect on users' responses: users generally perceived a difference between output generated under different parameter settings and, when asked, always selected the version with added variation as their preferred version or as the one most appropriate to the conversational situation. This result holds across both of the output modalities and all of the implementation strategies that were considered. The demographic differences of the participants had almost no effect on these results: the only effect of this type that was found was on the male and female subjects' opinions on the understandability of texts generated with anti-repetition measures (Section 5.2).

These results confirm the frequently-stated advantage of dynamic generation over static methods: that a system that creates its output dynamically is able to produce higher-quality output because it can adapt its output to the context and can incorporate naturalistic variation. In terms of the trade-off between implementation effort and output quality, it appears that it is almost always worth the effort to incorporate dynamic choice into a generation system, because it will almost always have a perceptible positive effect on the output.

There were no cases where users selected the default version as better or more appropriate than the version with added variation; however, there were a small number of cases where no significant effect was found, suggesting that users were unable to perceive the modifications to the output. Users were generally able to detect tailoring to negative user preferences more easily than they could detect tailoring to other preference types (Section 4.4, Section 7.3, Section 7.4, and Chapter 9); in particular, they often had difficulty distinguishing between positive and neutral tailoring. This suggests that the default mode in COMIC is that descriptions are positive unless explicitly marked as negative—in retrospect, this is not surprising, as COMIC is essentially acting as a "recommender" system. Features of the output modalities also sometimes prevented tailoring from being detected: in the dialogue-history evaluation (Section 4.3), the synthesised speech prevented users from noticing the more subtle forms of tailoring, while in Chapter 9 the weighted motions of the embodied agent appeared to have a similar effect.

10.1.2 Comparing implementation strategies

The second research question dealt with generation systems that use a corpus to help in the decision-making process, and asked whether higher-quality output could be created in such a system by selecting the majority option based on the corpus data, or whether modifying the generation process to select options other than the top one would be appreciated by users.

The studies in Chapter 5 and Chapter 8 both measured users' subjective preferences between outputs generated by taking the single highest-scoring option based on corpus similarity and outputs generated using other techniques: Chapter 5 looked at generated texts, while Chapter 8 used facial displays. The comparisons made in Chapter 5 and Section 8.3 are directly analogous: in both cases, participants were choosing between minimal pairs of outputs generated by a stochastic generation system, where one version was generated by selecting the highest-probability option in all cases and the other was created by modifying the selection process to choose other options. Participants in each of these studies showed a significant preference for the outputs that included lower-probability choices. This result agrees with several recent findings on user preferences for variation in generated text.

The results of the experiment in Section 8.4 provide an interesting counterpart to the above results. In this experiment, participants chose between facial displays generated by a stochastic system in weighted-choice mode and displays generated by a simple rule. While the results indicate a mild preference for the weighted-choice schedules over the rule-based ones, this preference did not reach significance; this contrasts with the very significant preference for the weighted-choice schedules over the majority-choice schedules. This contrast indicates that there is more to the subjective preferences than just the absolute range of options in the output: it is also relevant how evenly the options are distributed.

The experiment in Chapter 9 provides another comparison of the weighted and rule-based generation strategies: in this case, the differential impact of the two types of facial displays is measured on users' ability to detect the intended user-model tailoring of generated descriptions. The results of this study indicate that the schedules generated by the weighted strategy actually decrease performance on this task compared to the speech-only presentation, while the rule-based schedules do not have any impact; this holds even if we consider only those trials where the weighted strategy did not select conflicting head motions.

In summary, then, the findings on the relative merits of the two generation strategies are mixed. For text generation, the participants clearly preferred output generated by the modified system to that generated by the basic system. For non-verbal behaviour, on the other hand, the results are less clear: users weakly preferred the schedules generated by the weighted-choice strategy over those generated by the rule-based strategy; however, the weighted-choice schedules appeared to reduce users' ability to detect correctly tailored descriptions. In Section 10.3, we discuss potential follow-up studies addressing this issue.

In both Chapter 5 and Chapter 8, we also evaluated the various generation strategies using a range of automated metrics and compared the predictions of the metrics to the actual user preferences. We discuss the results of these studies in Section 10.1.3; that section also includes a further discussion of the user preferences for the rule-based facial displays.

10.1.3 Comparing evaluation techniques

The final research question asked whether the results of automated evaluation metrics correspond to the preferences of human judges when assessing the quality of generated output that includes variation. Two experiments in this thesis addressed this question directly: in Section 5.3 and Section 8.2, we used automatic evaluation metrics to compare the same generation strategies that were compared in the human preference studies in Section 5.2 and Sections 8.3–8.4, respectively. These two sets of evaluation results allow us to compare the rankings of the automated metrics to the actual preferences of the human judges, and therefore to test which of the automated metrics tend to agree with the human preferences.

By definition, automated metrics that directly compute the similarity of generated output to corpus examples—n-gram scores in Section 5.3; precision, recall, F score, and β in Section 8.2—will prefer any generation technique that selects the highest-probability option based on the corpus data. As described in Section 10.1.2, these are exactly the sorts of outputs that human judges tend to dislike the most: this result holds both for the experiments in this thesis and for other previous studies of text generation. This suggests that metrics based on simple corpus similarity are not suitable for being used alone to evaluate the output of a generation system, especially one that incorporates variation into its output. In Section 10.3, we discuss other corpus-based metrics that could be more useful for evaluating generated output.

In addition to the metrics based on corpus similarity, we also used several automated metrics that computed inherent properties of the generated texts themselves. For the texts in Chapter 5, we measured the mean edit distance across a set of descriptions, while for the facial displays in Chapter 8 we computed several values that indicated the number and variety of facial displays that were selected by each strategy. Metrics of this type generally corresponded more closely to the preferences in the human evaluations. For the texts, the preferred versions in the human study were those that had higher mean edit distance; for the facial displays, the weighted-choice schedules scored higher than the majority-choice schedules on all such metrics, which corresponded to the ranking of the human judges.

A particularly interesting case is the results for the rule-based strategy for selecting facial displays. This strategy scored by far the lowest on nearly all of the automated metrics; however, on the human evaluation, there was a weak, non-significant preference for the weighted strategy over this strategy, and only a mildly significant preference between it and the original corpus examples. This contrasts with the result for the majority-choice strategy, which was strongly dispreferred to both the weighted schedules and the original schedules in a similar evaluation. The only automated metric where the rule-based strategy was at all comparable to the others is the standard deviation of the distribution of generated facial displays: the values for the original schedules, the weighted schedules, and the rule-based schedules were all similar, while the value for the majority schedules was much higher. As mentioned in Section 10.1.1, this suggests that what was important to the participants in this study was the evenness of the facial-display distribution, rather than the absolute range of different displays.

10.2 Implications

The results summarised in the preceding section are all based on modifications to the output-generation components of the COMIC multimodal dialogue system. As discussed in Section 2.7 and Section 3.3, the output generated by this system—in both its linguistic content and its use of an embodied agent—is similar to that generated by a number of other systems. Indeed, in cases where an evaluation in this thesis uses similar materials or techniques to previous studies, the results are generally in agreement. The results on user-model tailoring in Section 4.4 agree with those of Walker *et al.* (2004) and Carenini and Moore (2006); the findings on subjective preferences for variation in stochastically-generated output confirm and extend those of Stent *et al.* (2005) and Belz and Reiter (2006) on text generation; while the influence that we found of affective non-verbal agent behaviour on the interpretation of embodied speech agrees with several recent results in the area including those of Rehm and André (2005) and Marsi and van Rooden (2007).

In addition to confirming previous results and in, some cases, extending them—to different modalities, different application domains, or different types of affective behaviours—these results also suggest some considerations for the design of future generation systems. In general, variation had a perceptible positive effect on the quality of generated output, whatever the implementation strategy and the specific modalities used. However, we did find that it is possible for linguistic tailoring that is perceptible in some presentation modes to cease to be perceptible in other modes. This means that it is important to consider the actual mode of presentation for a system and not to assume, for example, that subtle tailoring that is effective in written text will be effective when presented by a virtual agent using a synthetic voice.

Another consistent finding is that contextual tailoring was noticeable only when it was a marked departure from the default mode of the system: the default interpretation of output in COMIC appears to be that the agent believes it is describing something that the user likes unless explicit signals are given to the contrary. In other contexts, the default mode may be different. For example, when communicating error messages or warnings, the default interpretation may be that the message is something to be concerned about, and a marked variation would be to say that the warning is in some way benign.

These studies have provided a direct comparison of the two main current implementation techniques for selecting non-verbal behaviour for an embodied agent: using rules derived from the observation of human behaviour, and directly using recorded human motions. The results indicate that both of these techniques can be successful in the right circumstances. The output of corpus-based weighted choice was mildly preferred to the output of the rule-based strategy on a subjective choice experiment; however, users of the rule-based schedules performed better than users of the weighted-choice schedules at identifying the intended linguistic tailoring. This suggests that, for other embodied-agent implementations, a consideration for choosing between these two strategies should be whether the emphasis of the system is on subjective user satisfaction with the agent or on full understanding of the generated content.

The embodied-agent studies also demonstrate the utility of a *semantically transparent* corpus (van Deemter *et al.*, 2006) in which the full contextual information for every corpus entry is available. Such a corpus allows the full context available to the generation system to be used to make rich data-driven selection, in contrast to a surface-only corpus which only permits decisions to be made at the surface level. As discussed in Section 6.1, such deeper contextual information is not always available in current large-scale multimodal corpora; this fact has often led developers of generation systems to build their own application-specific corpora. If current large-scale multimodal corpora are to be useful for generation—in addition to the currently popular application areas of summarisation, analysis, and retrieval—then it is necessary to include the extra contextual information; this can be a non-trivial annotation task.

The results on the relationship between human and automated evaluations of generated output contribute to the current debate in the NLG community on the utility of shared-task evaluations (see Section 2.6.3). Like others, we found that automated metrics based purely on corpus similarity tend to prefer generated outputs that do not diverge from the average examples of the corpus; and, again like others, we found that it was exactly these outputs that were most disliked by the human judges. This provides additional evidence that, if a shared task is to be adopted—particularly the competitive shared tasks proposed by some—then the evaluation metrics must include more than simple recall/precision-style similarity measures. In machine-translation evaluation metrics such as BLEU, multiple reference translations are used to mitigate exactly this problem; however, it is not clear that expanding the space of correct answers will necessarily improve evaluations if simple similarity measures are used.

The results in this thesis indicate that other types of evaluation metrics—those that measure inherent properties of the generated outputs, rather than comparing them directly against the corpus examples—correspond better to human preferences. However, such metrics are necessarily more specific to the generation task than simple corpus-based similarity: the edit-distance-based variability metric used in Section 5.3 is not one that will provide useful information in every situation, for example. Also, these measures are only applicable in cases where the generation system is constrained to a space of basically "sensible" outputs: a strategy that chose entirely at random would score high on variability, which is not really what is intended. In Section 10.3.2, we discuss other potential metrics that make more indirect use of the corpus data.

10.3 Possible extensions

The results of the experiments have contributed to answering the three research questions set out at the start. In this section, we outline several possibilities for future work to follow up the results presented in this thesis. These fall into three main categories: other human evaluation techniques, other automated evaluation metrics, and other embodied-agent implementations.

10.3.1 Human evaluations

All of the human evaluations described in this thesis evaluated the generated output directly, either by gathering subjective preferences or by asking various questions designed to measure if the intended contextual tailoring was perceptible and appreciated. The results of these studies are that users are generally able to perceive variation—and also appreciate it—except under specific circumstances.

A possible extension to this work is to perform *task-based* evaluations: embedding the generated output into a system designed to address some larger task, and then comparing the impact on various system- and task-level measures of output generated with and without different types of variation. Several studies that have used this technique are described at the start of Section 2.6.1. While it is to be expected that perception is likely to translate into behavioural differences, this has not always been the case. In a meta-analysis of studies in HCI, for example, Nielsen and Levy (1994) found that while there was a strong positive correlation between users' subjective preferences between interfaces and their task performance using them, there were also a number of outliers. More recently, Rehm and André (2005) found that users were able to detect deceptive facial signals of an embodied agent in isolation, but in the context of a task where deception was relevant, the facial signals made no difference.

Another style of evaluation that can be used to compare users' reactions to different versions of the output is to record physiological measures of the participants. Measures of this type that have been used for evaluation include eye tracking (Prendinger *et al.*, 2005a), galvanic skin response (Prendinger *et al.*, 2005b), and interpersonal distance in a virtual environment (Bailenson *et al.*, 2004). These measures provide another dimension of comparison for the output, and are particularly suitable for evaluating interfaces that include embodied agents.

10.3.2 Automated evaluations

Automated evaluation metrics make the task of evaluating a generation system much easier: there is no need to recruit subjects, and large amounts of data can easily be processed. This means that evaluations can be run as frequently as necessary during the development process, and—if a shared-task evaluation framework is adopted for NLG—that systems can easily be compared against one another on a range of metrics. However, it is critical that any metric to be used in this way be grounded in actual human preferences.

In this thesis, we investigated a range of automated evaluation metrics. Those based on sentence-level similarity against a corpus tended to favour exactly those outputs that human judges most disliked: output that was based on the "average" examples in the corpus with little variation. In a task where there is a single right answer, this type of metric is useful; however, in generation—where there can often be a range of correct answers, sometimes with degrees of appropriateness—treating the corpus data as a gold standard is not appropriate. In machine translation and other related areas, the normal technique to get around this problem is to compute similarity against a corpus that includes multiple alternatives for each sentence, rather than a single "gold-standard" answer. This type of measure might help to mitigate the issues with corpus-based similarity metrics.

Another possibility is to move away from strict corpus-based similarity and towards metrics that compute properties of the generated outputs themselves. In the experiments in this thesis, metrics that computed inherent properties (such as the distribution of facial displays or the variability across a set of descriptions) corresponded more closely to the preferences of the human judges; however, this type of metric is necessarily more domain- and task-specific than one that compares against a corpus, and it is difficult to imagine such a metric being useful for a wide range of generation systems. Another alternative is to use techniques such as those of Karamanis and Mellish (2005) and Walker (2005, 2007) in which the corpus data is used more indirectly. Walker used machine learning on a corpus to derive a scoring function based on reward functions such as dialogue success, and then used the learned policy to score possible outputs in generation; Karamanis and Mellish compared text-ordering techniques by comparing how highly they rated the corpus examples.

10.3.3 Embodied conversational agents

The embodied-agent studies in this thesis all used the RUTH talking head (DeCarlo *et al.*, 2004), which has no body and, while human in appearance, is not particularly realistic. We used this agent to investigate two of the types of non-verbal behaviour described in Section 2.4.1: expressions of affect (in this case, user-preference evaluations), and visual correlates of prosody and information structure. More information about the relative utility of different techniques for selecting non-verbal behaviour for embodied agents can be gathered by experimenting with a wider range of agents and of non-verbal behaviours. Other possible agent types include photorealistic animated agents (Graf *et al.*, 2002; Cunningham, 2005), agents with fully articulated virtual bodies (Cassell *et al.*, 2001a; Stone *et al.*, 2004; Rehm and André, 2005), and physically embodied robot agents (Burghart *et al.*, 2005; de Ruyter *et al.*, 2005; Sidner *et al.*, 2005). The possibilities for non-verbal behaviours include deictic, iconic, and beat gestures, body posture, gaze behaviour, and expressions of other types of affect. Experimenting with other combinations of agent properties and behaviours can improve our knowledge of the relative utility of different mechanisms for selecting non-verbal behaviour.

In both of the user-preference evaluations of generated non-verbal behaviour in this thesis, the resynthesised versions of the original displays annotated in the corpus were rated more highly than any of the regenerated versions. This indicates that there is still room for improvement on the strategies used to select the non-verbal behaviours. In the implementations described here, the linguistic content was selected first, and then the displays were selected to accompany that content; however, as described in Section 2.4.1, there is evidence from studies of humans that content on the verbal and non-verbal channels is created simultaneously rather than in sequence as in this implementation. It is possible that a more integrated selection of content on the two channels—i.e., multimodal “unit selection,” as was implemented by Stone *et al.* (2004)—could produce higher-quality output. One possible implementation technique is to build n-grams models based on the data and the corpus and to use them to integrate the selection of facial displays directly into the OpenCCG realiser’s n-gram-based search for a good realisation. Such an implementation would have a better chance at capturing the complex interactions between the two output channels.

Appendix A

Sample interaction with COMIC

THE FOLLOWING is a transcript of a full interaction with COMIC. Figure A.1 shows how the screens of the COMIC demonstrator are normally configured: the talking head runs on a vertical screen, while the design application runs on a tablet display set horizontally on the table. The user sits in front of the configuration wearing a head-mounted, close-talk microphone.

A.1 Phase 0: ASR calibration and introduction

This initial phase is needed to calibrate the speech-recognition module for the rest of the interaction. Once the recogniser has been successfully calibrated, COMIC gives the user an overview of the system.

USER *[Taps the screen to wake up the system]*

COMIC Hello, and welcome to the COMIC system. Please say something, so that we can measure the level of background noise.

USER *[Speaks into the microphone]*

COMIC Cheers.

This system will help you to design your new bathroom. First, you will draw the shape of your room. Next you can choose how the sanitary ware will be arranged. After that, I will show you some choices for the tiles. When we are finished, you can take a three dimensional tour of your new room.

Figure A.1: COMIC demo configuration

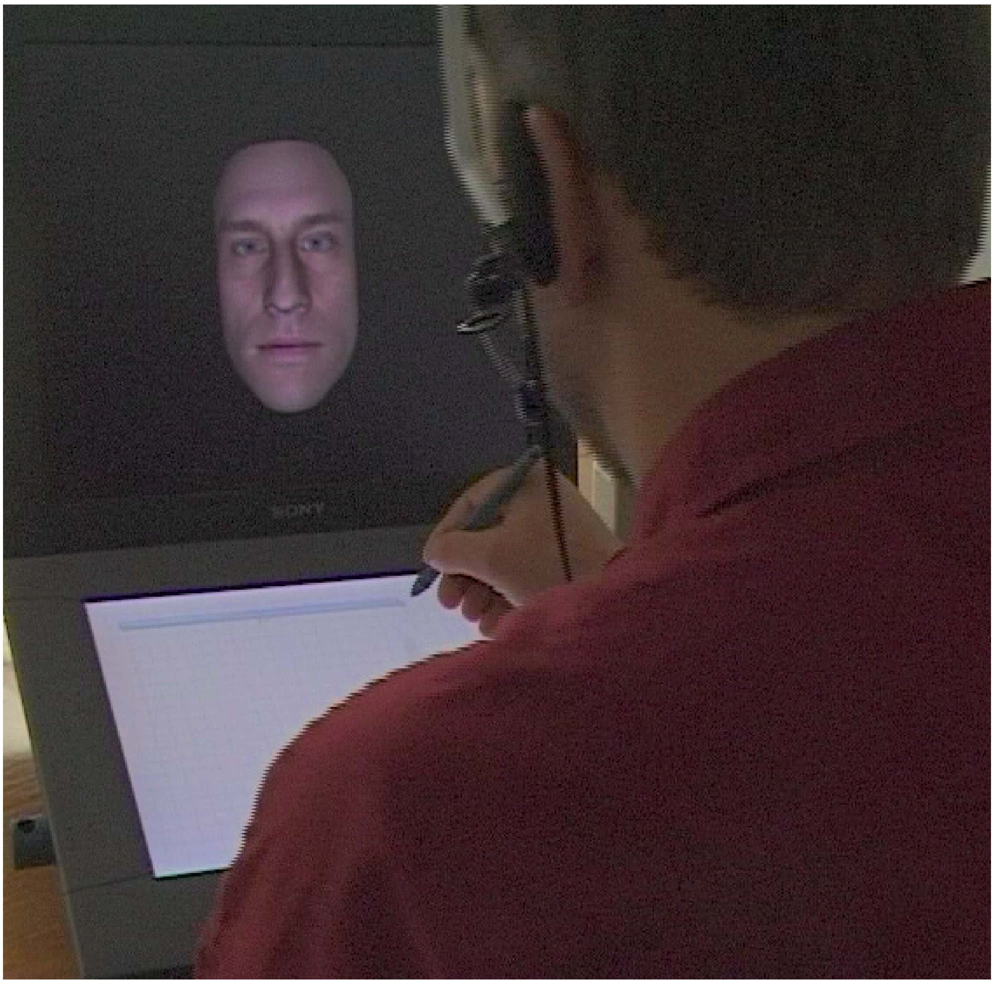
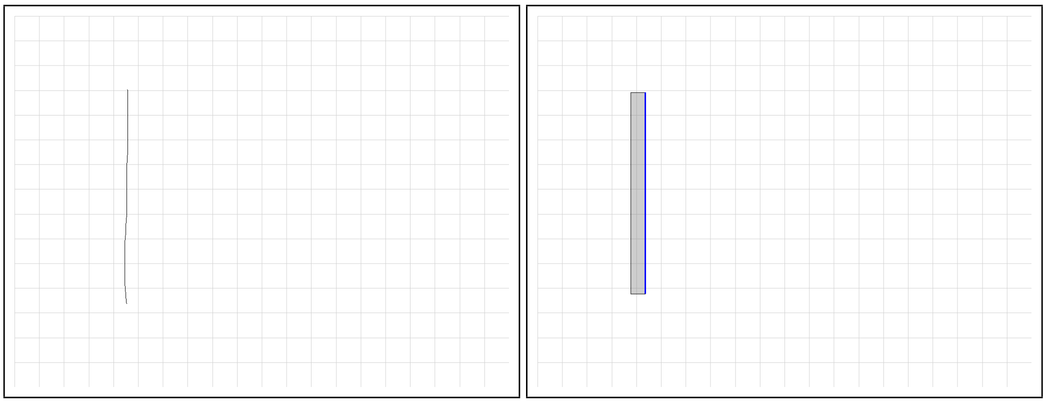


Figure A.2: Object recognition and beautification



(a) Wall drawn by user **(b)** Beautified wall

A.2 Phase 1: Specifying the shape of the room

In this phase, the user draws shapes and writes numbers with the pen on the screen. Once COMIC has recognised the input, in erases the ink and "beautifies" the recognised object on the screen, as in Figure A.2. The complete blueprint for the room created in this part of the interaction is in Figure A.3.

COMIC Please draw the walls of your room.

USER *[Draws a wall on the screen (Figure A.2(a))]*

COMIC *[Beautifies the wall (Figure A.2(b))]*

Please tell me the length of the first wall.

USER Three meters *[Writes "3m" on screen as well]*

COMIC *[Adds "300cm" label to wall]*

Please draw the rest of the walls.

USER *[Draws the second wall]*

COMIC *[Beautifies the wall]*

Please tell me the length of the second wall.

USER *[Writes "4m"]*

COMIC *[Adds "400cm" label to second wall; fills in the other two walls]* Thanks.

Please show me where the door is and which way it opens.

USER *[Draws the door]*

COMIC *[Beautifies the door]* Great.

Please show me where the window is.

USER *[Draws the window]*

COMIC *[Beautifies the window]* Cheers.

Please tell me the width of the window.

USER Ninety six centimetres. *[Recognised as "three"]*

COMIC What units is that size in?

USER Centimetres.

COMIC *[Thinks the user has said "three centimetres"]* I'm sorry, that value was too small.

Please tell me the width of the window.

USER Ninety six centimetres.

COMIC *[Adds "96cm" label to window]* Cheers.

What is the height of the window ?

USER Ninety two centimetres.

COMIC *[Adds "92cm" label to window]* Thanks.

What is the height of the window sill ?

USER Eighty two centimetres.

COMIC *[Adds "82cm" label to window]* Great.

You can move your door or window now. Or, you can say continue.

USER Erase this. *[Makes an erasing gesture on the door]*

COMIC *[Erases the door]* Okay, I've erased the door.

Please show me where the door is and which way it opens.

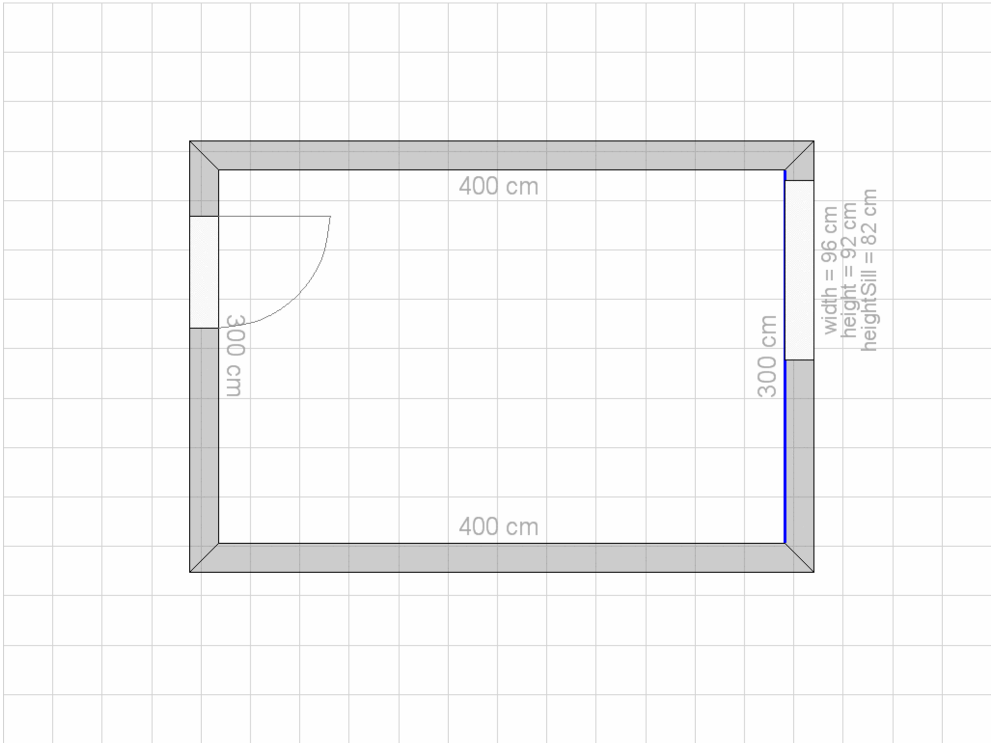
USER *[Draws a new door]*

COMIC *[Beautifies the door]* Cheers.

You can move your door or window now. Or, you can say continue.

USER Move this there. *[Draws a line from the window to a new location]*

Figure A.3: Completed room blueprint

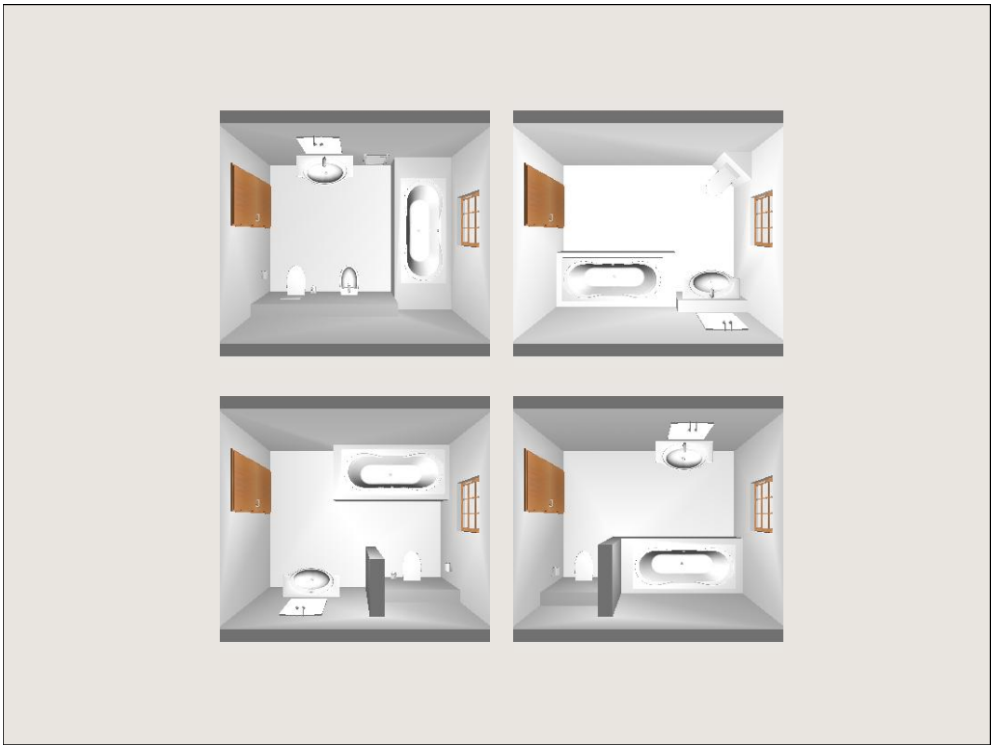


COMIC *[Moves the window to the new location]* Okay, I've moved the window.

You can move your door or window now. Or, you can say continue.

USER Continue.

Figure A.4: Sanitary-ware layouts



A.3 Phase 2: Choosing a layout

In this phase, the user chooses one of four possible layouts for the sanitary ware in their room. Figure A.4 shows the interface for this phase.

COMIC Now, let's take a look at the bathroom in three dimensions. *[Shows four possible sanitary-ware layouts]*

Please choose one of these layouts to use in your room.

USER *[Circles the top-right template]*

COMIC All right.

Figure A.5: Tile-browsing interface

A.4 Phase 3: Browsing tile designs

The interface for this phase is as shown in Figure A.5: there are up to five thumbnails of tiling designs at the bottom of the screen, and one of those designs is shown in the three-dimensional view at the top of the screen. COMIC also uses an on-screen pointer to draw the user's attention to what it is talking about in this phase; the pointer can be seen next to the leftmost thumbnail image in Figure A.5.

COMIC *[Shows the selected layout in three dimensions]* The next step is to choose the tiles for your bathroom. We have a range of designs for you to look at, which show a variety of styles, colours, and decorative ideas. Each design includes a small number of tiles that have been carefully selected to go together well. I can help you find a design that you like, and then you can see a three dimensional tour of the room.

Are you ready?

USER Yes.

COMIC All right.

[Circles first thumbnail] Here we have a design in the classic style. It has geometric

shapes and floral motifs on the decorative tiles. Its tiles are from the Alt Mettlach collection by Villeroy and Boch. This is one of our most popular collections.

[Circles other thumbnails as it describes them] This design is in the modern style, while this one is in the family style. This design is in the modern style, while this one is country. Please choose one of these designs to look at next.

USER *[Circles rightmost design]*

COMIC *[Shows selected design in 3D]* This design draws from Smart, by Gardenia Orchidea. The colours are green and white. There are geometric shapes on the decorative tiles.

Would you like to see more country designs?

USER Show me designs with yellow tiles. *[Not recognised]*

COMIC I'm sorry, I couldn't understand what you just said.

We have more country designs. Would you like to see them?

USER Show me designs with yellow tiles.

COMIC All right. *[Shows a new design selection]* Here are some designs with yellow tiles.

[Circles first thumbnail] This design features blue and yellow in the colour scheme. Once again it is in the country style, but here the tiles are from the Colorado collection by Aparici.

[Circles other thumbnails as it describes them] This design features white, red and yellow in the colour scheme, while this one features green, yellow and red. This design features rose, yellow and white in the colour scheme, while this one features blue and yellow. Please choose one of these designs to look at next.

USER *[Circles rightmost thumbnail]*

COMIC *[Shows selected design in 3D]* This design has comic motifs on the decorative tiles. It is in the family style. The tiles draw from Asterix, by Villeroy and Boch.

We have more family designs. Would you like to see them?

USER Show me more decoration. *[Not recognised]*

COMIC I'm sorry, I didn't understand you completely.

We have more family designs. Would you like to see them?

USER Show me more decoration.

COMIC All right. *[Increases the number of decorative tiles shown]*

Would you like to see more family designs?

USER Show me a 3D tour.

COMIC Great.

Figure A.6: Three-dimensional tour (still frame)

A.5 Phase 4: Three-dimensional tour

In this phase, the system gives a three-dimensional animated walkthrough of the finished bathroom. Figure A.6 shows a still frame from the animation.

COMIC *[Shows animated 3D walkthrough of the selected room]* Let's take a three dimensional look at these tiles.

[Pauses for several seconds]

Would you like to go back and look at more tiles?

USER Yes.

Figure A.7: Dialogue box for borders and decoration

A.6 Return to phase 3 and exit

This part of the interaction shows one more phase-3 feature: in addition to choosing among tiling designs, the user can also fine-tune the borders and decoration of the selected design. This can be done either using speech (as in the *Show me more decoration* interaction in Section A.4), or using a dialogue box as shown in this section (Figure A.7).

COMIC You can tell me the colour, style, or designer you would like to see , or choose one of these designs here. You can also change the borders or decoration, or see a three D tour of the current bathroom. Let me know what you want to do.

USER Show me more family designs.

COMIC Sure. *[Shows a new selection of designs]* Here are some designs in the family style.

[Circles first thumbnail] This design has animal motifs on the decorative tiles. The tiles are from the Creative Amazonas collection by Villeroy and Boch.

Please choose one of these designs to look at next.

USER *[Circles third thumbnail]*

COMIC *[Shows selected design in 3D]* This design has an orange, blue and green colour scheme. The tiles draw from Funny Day, by Villeroy and Boch. It has floral motifs on the decorative tiles.

We have more designs by Villeroy and Boch. Would you like to see them?

USER I would like to change the decoration. *[Not recognised]*

COMIC I'm sorry, I didn't understand that.

Would you like to see more designs by Villeroy and Boch?

USER I would like to change the decoration.

COMIC *[Shows a dialogue box (Figure A.7)]* Please use the buttons to change the amount of borders and decoration. Press "Done" when you are finished.

USER *[Clicks on the buttons to change the borders and decoration.]*

USER *[Closes the dialogue box]*

COMIC Would you like to see more designs by Villeroy and Boch?

USER I want to quit.

COMIC Do you really want to quit?

USER Yes.

COMIC Thank you for using the COMIC system. Good-bye.

Appendix B

Technical description of the COMIC fission module*

In Section 3.1.5.1, we gave a short description of the role of the fission module—that is, the presentation planner—within the COMIC multimodal dialogue system, and in Section 3.2 we described how it chooses the content of dynamically-created turns. In this appendix, we describe how the module prepares the parts of a presentation and outputs them incrementally, and give some technical details of exactly how the module works.

B.1 Introduction

When the fission module receives input from the dialogue manager, it selects and structures multimodal content to create an output plan, using a combination of scripted and dynamically-generated output segments. The fission module addresses the tasks of low-level content selection, text planning, and sentence planning; surface realisation of the sentence plans is done by the OpenCCG realiser. The fission module also controls the output of the planned presentation by sending appropriate messages to the output modules including the text realiser, speech synthesiser, talking head, and bathroom-design GUI. Coordination across the modalities is implemented using a two-step technique: the synthesised speech is prepared in advance, and the timing information from the synthesiser is used to create the schedule for the other modalities.

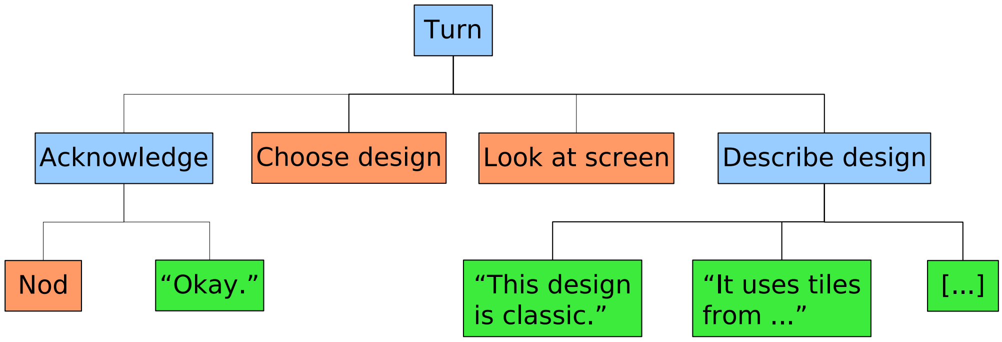
The plan for an output turn in COMIC is represented in a tree structure; for example, Figure B.2 shows part of the plan for the output in Figure B.1. A plan tree like this is created

*This appendix is based on Foster (2005).

Figure B.1: Sample COMIC input and output

User	Tell me about this design *[circle Alt Mettlach]*
COMIC	*[Nod]*
	Okay!
	[Change screen so selected design is shown in room]
	[Look at screen]
	THIS DESIGN is CLASSIC. *[circle tiles]*
	It uses tiles from the ALT METTLACH collection by VILLEROY AND BOCH. *[point at manufacturer name]*
	As you can see, the colours are DARK RED and OFF WHITE. *[point at tiles]*

Figure B.2: Output plan



from the top down, with the children created left-to-right at each level, and is executed in the same order. The planning and execution processes for a turn are started together and run in parallel, which makes it possible to begin producing output as soon as possible and to continue planning while output is active. In the following section, we describe the set of classes and algorithms that make this interleaved preparation and execution possible.

The COMIC fission module is implemented in a combination of Java and XSLT. The final module consists of 18 000 lines of Java code in 88 source files, and just over 9000 lines of XSLT templates. In the diagrams and algorithm descriptions that follow, some non-essential details are omitted for simplicity.

B.2 Representing an output plan

Each node in a output-plan tree such as that shown in Figure B.2 is represented by an instance of the Segment class. The structure of this abstract class is shown in Figure B.3; the fields and methods defined in this class control the preparation and output of the corresponding segment of the plan tree, and allow preparation and output to proceed in parallel.

Figure B.3: Structure of the Segment class

Segment
parent : Sequence # ready : boolean # skip : boolean # active : boolean
+ *plan()* + *execute()* # reportDone()

Each Segment instance stores a reference to its parent in the tree, and defines the following three methods:

- `plan()` Begins preparing the output.
- `execute()` Produces the prepared output.
- `reportDone()` Indicates to the Segment's parent that its output has been completed.

`plan()` and `execute()` are abstract methods of the Segment class; the concrete implementations of these methods on the subclasses of Segment are described later in this section. Each Segment also has the following Boolean flags that control its processing; all are initially false.

- `ready` This flag is set internally once the Segment has finished all of its preparation and is ready to be output.
- `skip` This flag is set internally if the Segment encounters a problem during its planning, and indicates that the Segment should be skipped when the time comes to produce output.
- `active` This flag is set externally by the Segment's parent, and indicates that this Segment should produce its output as soon as it is ready.

The activity diagram in Figure B.4 shows how these flags and methods are used during the preparation and output of a Segment. Note that a Segment may send asynchronous queries to other modules as part of its planning. When such a query is sent, the Segment sets its internal state and exits its `plan()` method; when the response is received, preparation continues from the last state reached. Since planning and execution proceed in parallel across the tree, and the planning process may be interrupted to wait for responses from other modules, the `ready` and `active` flags may be set in either order on a particular Segment. Once both of these flags have been set, the `execute()` method is called automatically. If both `skip` and `active` are set, the Segment instead automatically calls `reportDone()` without ever executing; this allows Segments with errors to be skipped without affecting the output of the rest of the turn.

Figure B.4: Segment preparation and output

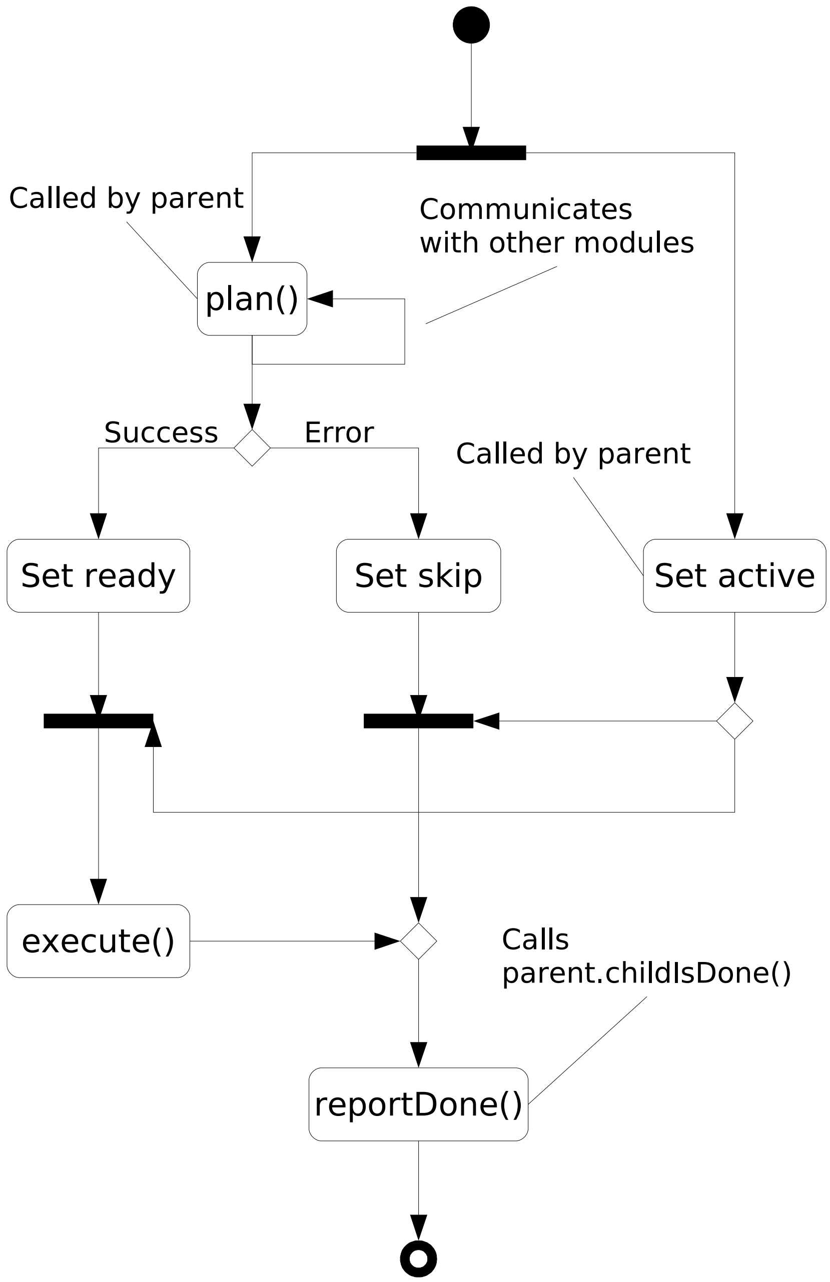
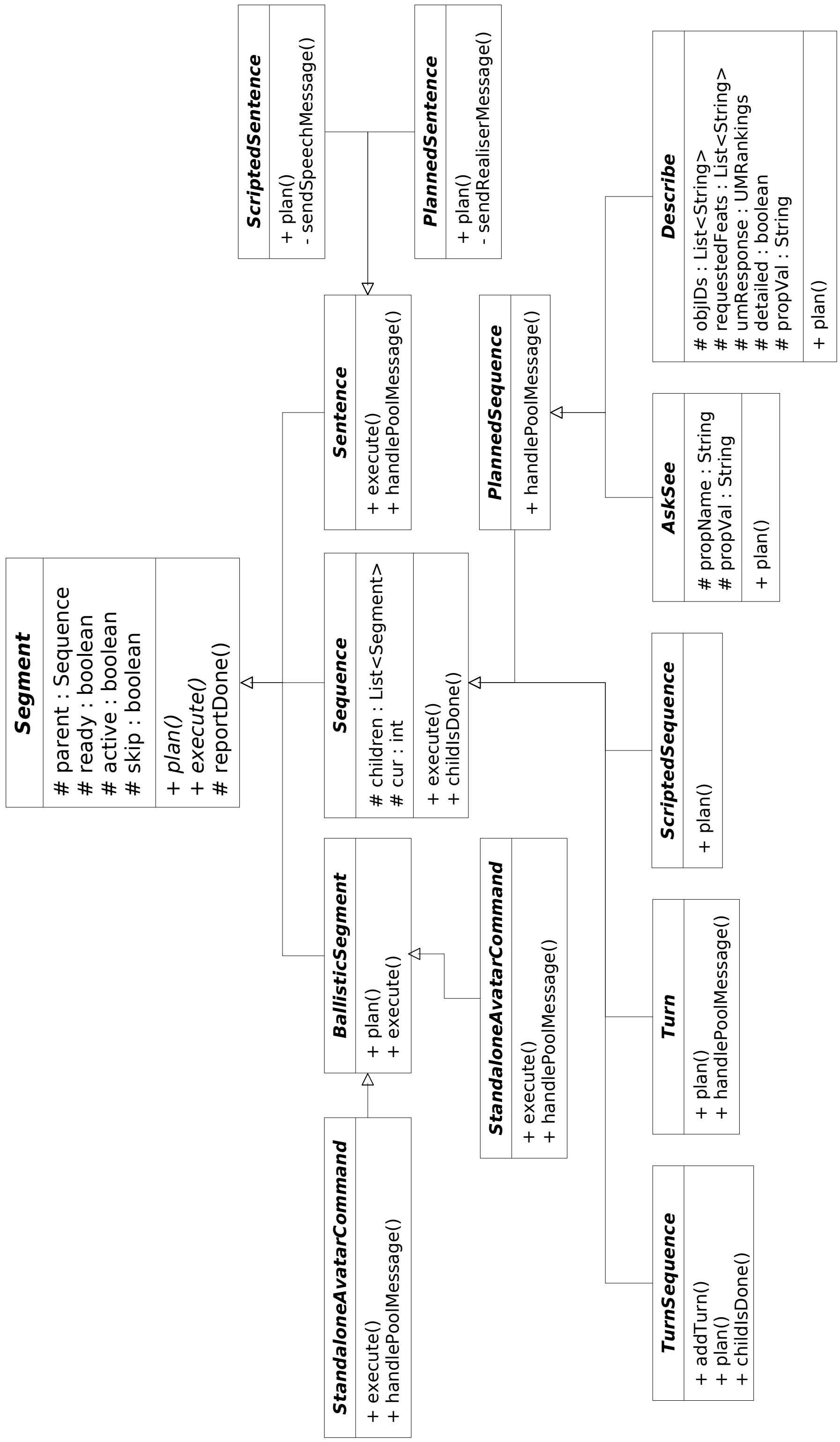


Figure B.5: Segment class hierarchy



The full class hierarchy under Segment is shown in Figure B.5. There are three top-level subclasses of Segment, which differ mainly based on how they implement `execute()`:

Sequence An ordered sequence of Segments. It is executed by activating each child in turn.

BallisticSegment A single command whose duration is determined by the module producing the output. It is executed by sending a message to the appropriate module and waiting for that module to report back that it has finished.

Sentence A single sentence, incorporating coordinated output in all modalities. Its schedule is computed in advance, as part of the planning process; it is executed by sending a "go" command to the appropriate output modules.

In the rest of this section, we discuss each of these classes and its subclasses in more detail.

B.2.1 Sequence

All internal nodes in a presentation-plan tree are instances of some type of Sequence. A Sequence stores a list of child Segments, which it plans and activates in order, along with a pointer to the currently active Segment. Figure B.6 shows the pseudocode for the main methods of a typical Sequence.

Note that a Sequence calls sets its `ready` flag as soon as all of its necessary child Segments have been created, and only then begins calling `plan()` on them. This allows the Sequence's `execute()` method to be called as soon as possible, which is critical to allowing the fission module to begin producing output before the full tree has been created.

When `execute()` is called on a Sequence, it calls `activate()` on the first child in its list. All subsequent children are activated by calls to the `childIsDone()` method, which is called by each child as part of its `reportDone()` method after its execution is completed. Note that this ensures that the children of a Sequence will always be executed in the proper order, even if they are prepared out of order. Once all of the Sequence's children have reported that they are done, the Sequence itself calls `reportDone()`.

The main subclasses of Sequence, and their relevant features, are as follows:

TurnSequence The singleton class that is the parent of all Turns. It is always active, and new children can be added to its list at any time.

Figure B.6: Pseudocode for Sequence methods

```
public void plan() {
    // Create child Segments

    cur = 0;
    ready = true;

    for( Segment seg: children ) {
        seg.plan();
    }
}

public void execute() {
    children.get( 0 ).activate();
}

public void childIsDone() {
    cur++;
    if( cur >= children.size() ) {
        reportDone();
    } else {
        children.get( cur ).activate();
    }
}
```

Turn Corresponds to a single message from the dialogue manager; the root of the output plan in Figure B.2 is a Turn. Its `plan()` implementation creates a Segment corresponding to each dialogue act from the dialogue manager; in some cases, the Turn adds additional children not directly specified by the DAM, such as the verbal acknowledgement and the gaze shift in Figure B.2.

ScriptedSequence A sequence of canned output segments stored as an XSLT template. A ScriptedSequence is used anywhere in the dialogue where dynamically-generated content is not necessary; for example, instructions to the user and acknowledgements such as the left-most subtree in Figure B.2 are stored as ScriptedSequences.

PlannedSequence In contrast to a ScriptedSequence, a PlannedSequence creates its children dynamically depending on the dialogue context. The principal type of PlannedSequence is a description of one or more tile designs, such as that shown in Figure B.1. To create the content of such a description, the fission module uses information from the system ontology, the dialogue history, and the model of user preferences to select and structure the facts about the selected design and to create the sequence of sentences to realise that content. This process is described in detail in Section 3.2.

B.2.2 BallisticSegment

A BallisticSegment is a single command for a single output module, where the output module is allowed to choose the duration at execution time. In Figure B.2, the *Nod*, *Choose design*, and *Look at screen* nodes are examples of BallisticSegments. In its `plan()` method, a BallisticSegment transforms its input specification into an appropriate message for the target output module. When `execute()` is called, the BallisticSegment sends the transformed command to the output module and waits for that module to report back that it is done; it calls `reportDone()` when it receives that acknowledgement.

B.2.3 Sentence

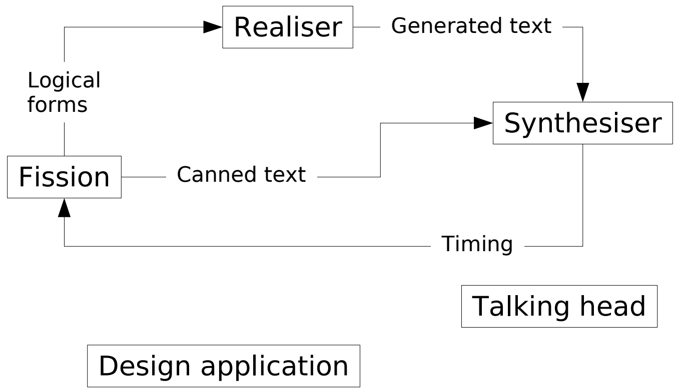
The Sentence class represents a single sentence, combining synthesised speech, lip-synch commands for the talking head, and possible coordinated behaviours on the other multimodal channels. The timing of a sentence is based on the timing of the synthesised speech; all multimodal behaviours are scheduled to coincide with particular words in the text. Unlike a BallisticSegment, which allows the output module to determine the duration at execution time, a Sentence must prepare its schedule in advance to ensure that output is coordinated across all of the channels. In Figure B.2, all of the leaf nodes containing text are instances of Sentence.

There are two types of Sentences: ScriptedSentences and PlannedSentences. A ScriptedSentence is generally created as part of a ScriptedSequence, and is based on pre-written text that is sent directly to the speech synthesiser, along with any necessary multimodal behaviours. A PlannedSentence forms part of a PlannedSequence, and is based on a logical form for the OpenCCG realiser (White, 2005, 2006b).

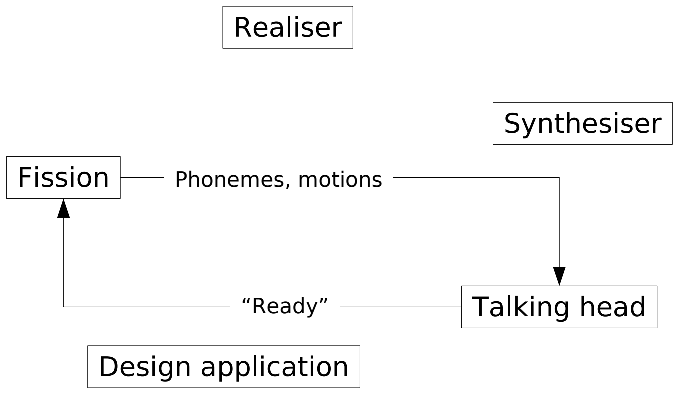
The first step in preparing either type of Sentence is to send the text to the speech synthesiser (Figure B.7(a)). For a ScriptedSentence, the canned text is sent directly to the speech synthesiser; for a PlannedSentence, the logical forms are sent to the realiser, which then creates the text and sends it to the synthesiser. In either case, the speech-synthesiser input also includes marks at all points where multimodal output is intended. The speech synthesiser prepares and stores the waveform based on the input text, and returns timing information for the words and phonemes, along with the timing of any multimodal coordination marks.

The fission module uses the returned timing information to create the final schedule for all modalities. It then sends the animation schedule (lip-synch commands, along with any coordinated expression or gaze behaviours) to the talking-head module so that it can prepare its animation in advance (Figure B.7(b)). Once the talking-head module has prepared the animation for a turn, it returns a *ready* message. The design application does not need its schedule

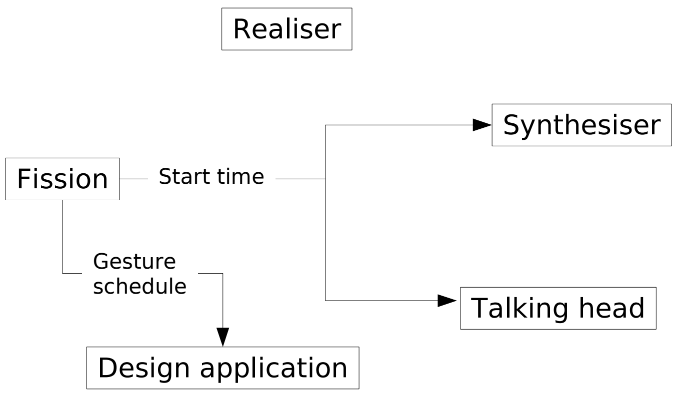
Figure B.7: Planning and executing a Sentence



(a) Preparing the speech



(b) Preparing the animation



(c) Producing the output

in advance, so once the response is received from the talking head, the Sentence has finished its preparation and is able to set its `ready` flag.

When a Sentence is executed by its parent, it selects a desired start time slightly in the future and sends two messages, as shown in Figure B.7(c). First, it sends a *go* message with the selected starting time to the speech-synthesis and talking-head modules; these modules then play the prepared output for that turn at the given time. The Sentence also sends the concrete schedule for any coordinated gesture commands to the bathroom-design application at this point. After sending its messages, the Sentence waits until the scheduled duration has elapsed, and then calls `reportDone()`.

B.3 Robustness and configurability

In the preceding section, we gave a description of the data structures and methods that are used when preparing and executing and output plan. In this section, we describe two other aspects of the module that are important to its functioning as part of the overall dialogue system: its ability to detect and deal with errors in its processing, and the various configurations in which it can be run.

B.3.1 Error detection and recovery

Since barge-in is not implemented in COMIC, the fission module plays an important role in turn-taking for the whole COMIC system: it is the module that informs the input components when the system output is finished, so that they are able to process the next user input. The fission module therefore incorporates several measures to ensure that it is able to detect and recover from unexpected events during its processing, so that the dialogue is able to continue even if there are errors in some parts of the output.

Most input from external modules is validated against XML schemas to ensure that it is well-formed, and any messages that fail to validate are not processed further. As well, all queries to external modules are sent with configurable time-outs, and any Segment that is expecting a response to a query is also prepared to deal with a time-out.

If a problem occurs while preparing any Segment for output—either due to an error in internal processing, or because of an issue with some external module—that Segment immediately sets its `skip` flag and stops the preparation process. As described in section B.2, any Segments with this flag set are then skipped at execution time. This ensures that processing is able to continue as much as possible despite the errors, and that the fission module is still able to produce output from the parts of an output plan unaffected by the problems and to perform its necessary turn-taking functions.

B.3.2 Configurability

The COMIC fission module can be run in several different configurations, to meet a variety of evaluation, demonstration, and development situations. The fission module can be configured not to wait for *ready* and *done* responses from either or both of the talking-head and design-application modules; the fission module simply proceeds with the rest of its processing as if the required response had been received. This allows the whole COMIC system to be run without those output modules enabled. This is useful during development of other parts of the system, and for running demos and evaluation experiments where not all of the output channels are used; the experiments in Chapter 4 were run using this mode. The module also has a number of other configuration options to control factors such as query time-outs and the method of selecting multimodal coarticulations.

As well, the fission module has the ability to generate multiple alternative versions of a single turn, using different user models, dialogue-history settings, or multimodal planning techniques; this is useful both as a testing tool and as part of a system demonstration. The module can also store all of the generated output to a script, and to play back the scripted output at a later time using a subset of the full system. This allows alternative versions of the system output to be directly compared in user evaluation studies such as those described in Chapter 4 of this thesis.

B.4 Output speed

In the final version of the COMIC system, the average time[1] that the speech synthesiser takes to prepare the waveform for a sentence is 1.9 seconds, while the average synthesised length of a sentence is 2.7 seconds. This means that, on average, each sentence takes long enough to play that the next sentence is ready as soon as it is needed; and even when this is not the case, the delay between sentences is still greatly reduced by the parallel planning process.

The importance of beginning output as soon as possible was demonstrated by a user evaluation of an interim version of COMIC (White *et al.*, 2005). Participants in that study used the full COMIC system in one of two configurations: an "expressive" condition, where the talking head used all of the expressions it was capable of, or a "zombie" condition where all of the behaviours of the head were disabled except for lip-synch. One effect of this difference was that

[1] On a Pentium 4 1.6GHz computer.

the system gave a consistently earlier response in the expressive condition—a facial response was produced an average of 1.4 seconds after the dialogue-manager message, while spoken input did not begin for nearly 4 seconds. Although both versions of the system were very slow, the participants in the expressive condition were significantly less likely to mention the overall slowness than the participants in the zombie condition.

After this interim evaluation, effort was put into further reducing the delay in the final system. For example, we now store the waveforms for acknowledgements and other frequently-used texts pre-synthesised in the speech module instead of sending them to Festival, and other internal processing bottlenecks were eliminated. Using the same computers as the interim evaluation, the fission delay for initial output is under 0.5 seconds in the final system.

Appendix C

Instructions and interfaces for human evaluations

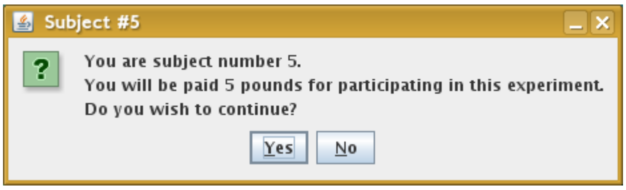
THIS APPENDIX describes the interfaces that were used for each of the experiments described in this thesis, including full instructions that were given to the participants. Most of the experiments were run over the web and were listed on the Language Experiments Portal (`http://www.language-experiments.org/`). All of the web-based experiments used the same PHP+JavaScript test harness, with modifications as necessary for the specific experiments. For each of the web-based experiments, we show the full content of the web pages that were used and describe how the experiment progressed from page to page.

For the experiments in Chapter 4, we used a cut-down, output-only version of the full COMIC system to play canned output and controlled it programmatically. The user interface for this evaluation was a series of windows with buttons and check-boxes. For this experiment, we present screenshots of the user interface and describe how the experiment progressed.

C.1 Context-tailored textual descriptions

These two studies (Section 4.3 and Section 4.4) were run consecutively in a single session, so we describe the interface for the two experiments together. This experiment used a cut-down, output-only version of the full COMIC system that was able to play back scripted output. User input was provided by playing recordings of a user making requests to the system. This allowed us to ensure that every participant saw and heard exactly the same version of each system turn. The output modules in these experiments were the speech synthesiser and the ViSoft emulator; the talking head and full bathroom application were not used.

Figure C.1: Initial window for dialogue history/user model study



C.1.1 Initial steps

The test harness was installed on the Edinburgh informatics network and could be run on any centrally-managed computer with the correct sound hardware. A participant ran a shell script to start the evaluation. This shell script recorded the userid of the participant to ensure that they received their compensation and to make sure that no person did the experiment more than once. After making an initial sanity check on the computer's sound hardware[1] and verifying that the current user had not already taken part in the experiment, the test harness was started.

Participants were compensated for taking part in the experiment. The first 20 participants received £5 for taking part, while the next 5 received £3. The first window that was shown after the sanity check presented this information to the participants, as shown in Figure C.1.

If the user clicked *No*, the test harness exited; If they clicked *Yes*, a window was displayed that gave a short introduction to the COMIC system and to the experiment (Figure C.2). This window also allowed participants to play a test sentence using the ViSoft emulator to ensure that everything worked properly on the computer they were using. This window also requested several pieces of demographic information: the participant's age range (Under 20, 20-24, 25-29, 30-34, or 35+), gender, first language (English or other), and level of computer experience (beginner, intermediate, or expert). The experiment could not be started unless the participant entered all of this information.

After the user had verified that the program worked properly on their computer and had entered the necessary demographic information, they clicked *Start experiment* and the dialogue-history study began.

C.1.2 Evaluation of dialogue-history tailoring

After the initial steps listed in the preceding section were completed, the main window for the dialogue-history study (Figure C.3) was displayed. This window described the details of the dialogue-history study: the type of output that would be displayed and the judgements that

[1]Not all of the computers in the student labs had working sound cards.

Figure C.2: Initial instructions

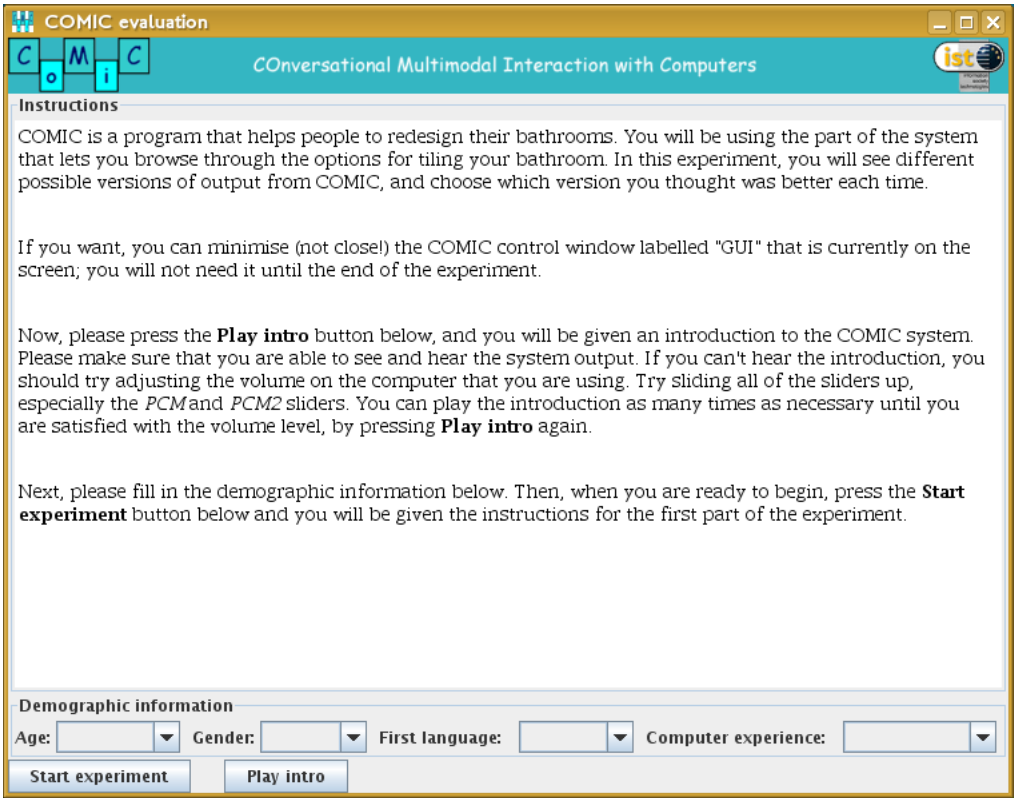


Figure C.3: Instructions for dialogue-history study

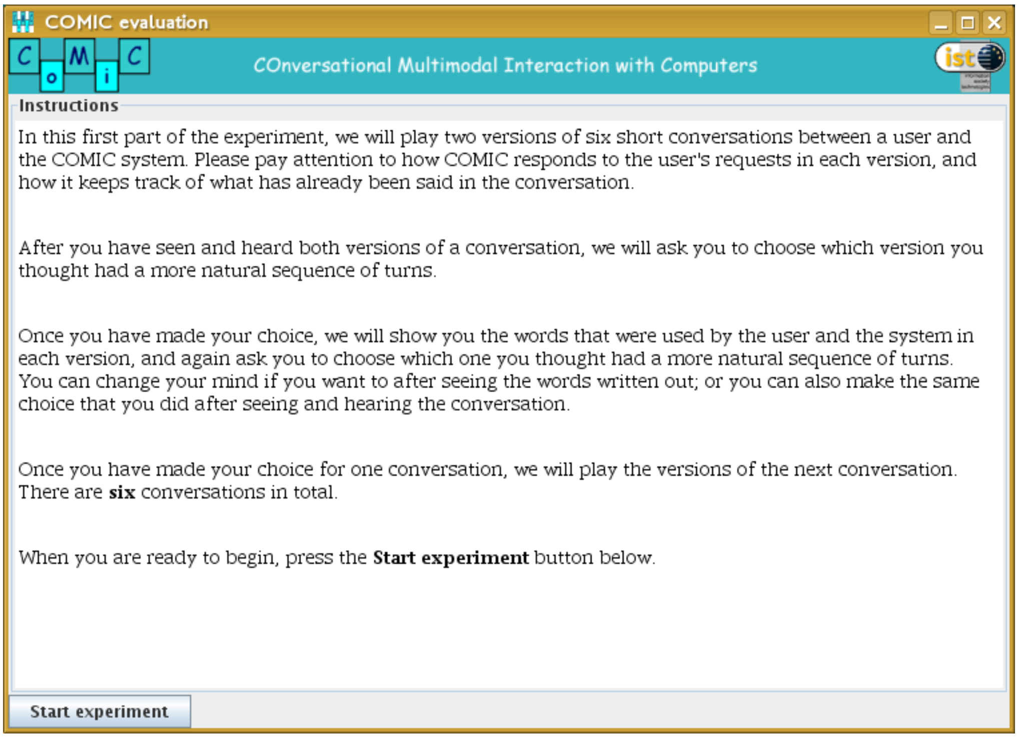


Figure C.4: Windows for playing dialogue-history options

(a) First *Play* window

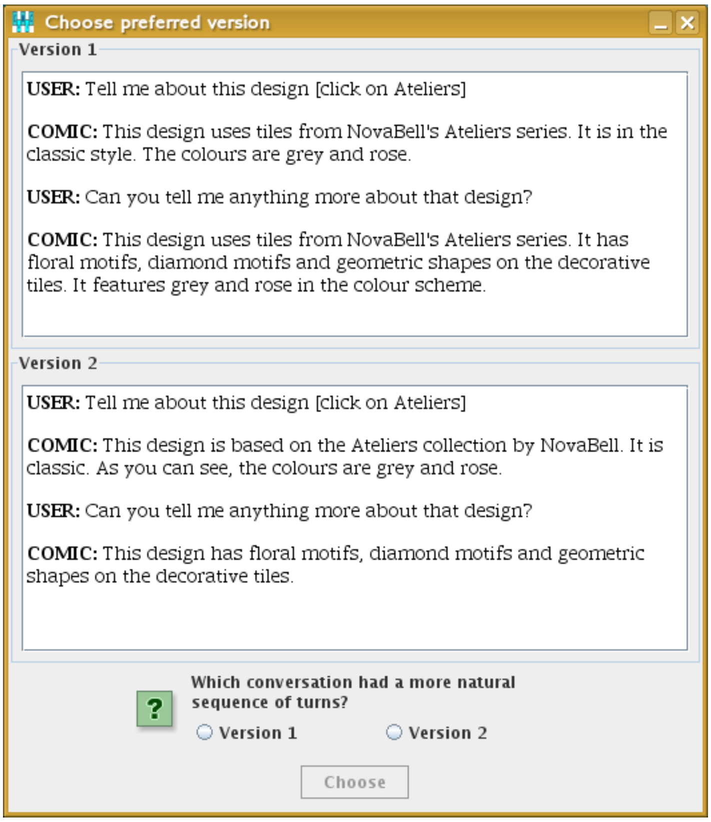
(b) Second *Play* window

the participant should make regarding the output. The trials all involved selecting between two versions of a short dialogue, one in which the system took into account the preceding context when producing later turns and one in which it did not. See Section 4.3 for details.

Once a participant had read through the instructions, he or she pressed *Start experiment* to begin the experiment. After *Start experiment* was clicked, the instruction window was closed and a dialogue box asking the user to click to play the first version of the first dialogue was displayed (Figure C.4(a)). When the *Play* button was clicked, the dialogue box disappeared and the ViSoft emulator (Section 3.1.2.1) was loaded. A short dialogue consisting of two user requests and system responses was then played. The user input was simulated by playing a recorded speech clip and simulating clicks on the emulator interface, while the output was

Figure C.5: Windows for making selections

(a) Speech selection window

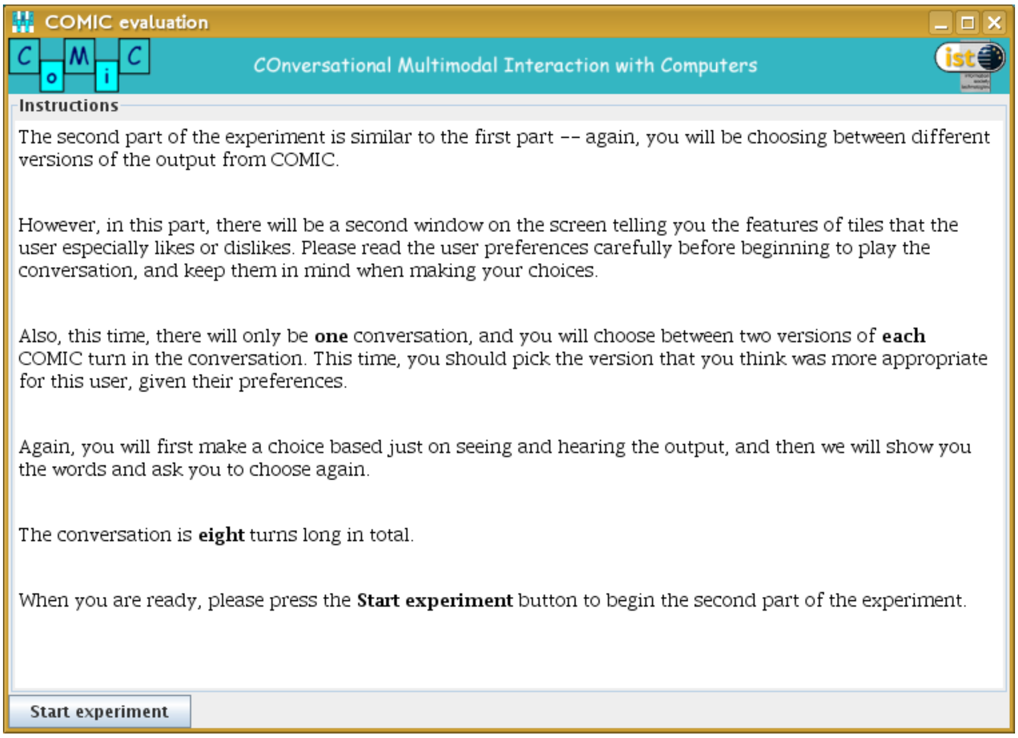


(b) Text selection window

produced by playing back scripted output including synthesised speech and ViSoft emulator actions. After the first version of the dialogue was played, the emulator was hidden and a dialogue box (Figure C.4(b)) was displayed asking the participant to click for the second version of that dialogue. When *Play* was clicked in this window, the second version was played.

After both versions of the dialogue were played, the emulator was closed and a window was displayed asking the user to choose which of the two versions had the more coherent sequence of turns (Figure C.5(a)). The *Choose* button could not be clicked until either version 1 or

Figure C.6: Instructions for user-model study



version 2 was selected, and neither was selected by default. After the preferred version had been chosen, the transcript of both versions was displayed, and the user was again asked to select the preferred version (Figure C.5(b)).

Once both selections had been made based for a dialogue, the responses were logged and the *Play* window (Figure C.4(a)) was displayed for the next conversation. This continued until all six of the dialogue-history dialogues had been processed; note that the same six dialogues were used for all participants, but they were presented in an individually-chosen random order each time.

Once the dialogue-history study was completed, the experiment continued with the user-model study.

C.1.3 Evaluation of user-preference tailoring

Directly after the user chose the preferred version for the sixth and final dialogue-history trial (as described in the preceding section), the instructions for the user-mode study were displayed (Figure C.6). Again, these instructions describe the sort of output that would be presented and the judgement that the user was intended to make about the outputs. In this case,

Figure C.7: User-preference window

Figure C.8: Windows for playing user-model options

(a) First *Play* window

(b) Second *Play* window

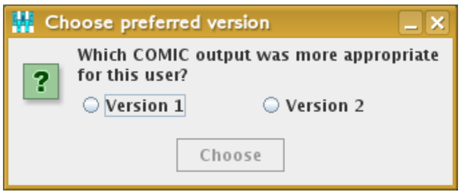
the task was to select which of the two versions of each system turn was correctly tailored to a target user model: each trial involved two versions, one correctly tailored to the preferences of a hypothetical user and one tailored to the preferences of some other user. Section 4.4 has full details of the methodology and materials for this study.

Once *Start experiment* was clicked, the instruction window disappeared and the target user model for this study was displayed on screen (Figure C.7). This window remained on screen throughout the experiment. There were four different user models, and participants were assigned to each as the target in turn. As in the first experiment, a window was also displayed asking the user to click a button to play the first version of the first turn in the dialogue (Figure C.8(a)).

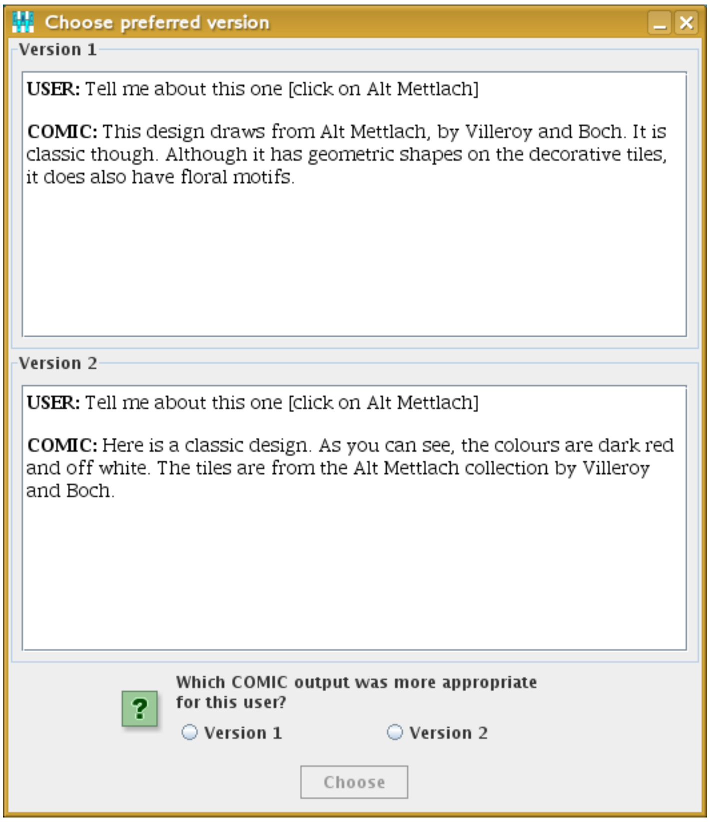
As in the first experiment, when this button was clicked, the first user request and the first version of the system response to that request were played, using recorded speech and simulated clicks for the user input and the ViSoft emulator and the speech synthesiser to produce the system output. After the first version was played, the ViSoft emulator was hidden and another window (Figure C.8(b)) was displayed asking the user to click to play the second version of the turn.

After the second version was played, a window similar to that used in the first experiment was displayed asking the user to choose which of the two versions they felt was more appropriate, based on the spoken presentation (Figure C.9(a)). After the user had made their selection based on the full COMIC presentation, the transcripts of the two versions were shown and the user was asked to chose again (Figure C.9(b)). As in the previous study, the choice was forced: the *Choose* button could not be clicked unless the user made a selection.

Figure C.9: Windows for making selections

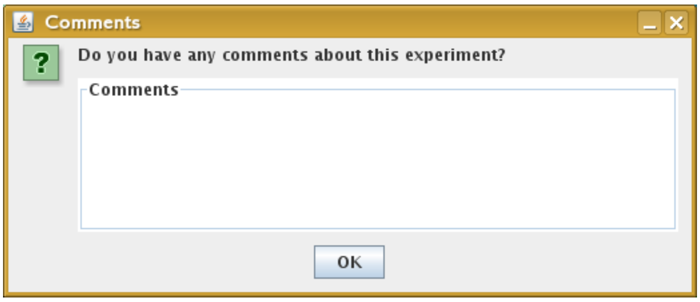


(a) Speech selection window



(b) Text selection window

Figure C.10: Comment form for textual-tailoring evaluation



After the user made both choices on a turn, the responses were logged and the user was prompted again (Figure C.8(a)) to play the first version of the next turn. There were eight turns in total: all participants with the same target model saw the same turn sequence, but the alternative versions of each turn were assigned individually for each subject.

C.1.4 Comments

After all the user made a choice for all eight of the turns, they were given the opportunity to leave comments on the experiment (Figure C.10). After that window was closed, the experiment ended.

C.2 Periphrastic variation in text

This experiment compared user preferences between texts generated with and without measures to avoid repetitiveness; see the details are given in Section 5.2. This experiment was run over the web, and all participants were entered into a draw for a gift certificate from Amazon.

The initial web page for the experiment is shown in Figure C.11. This web page introduces the experiment, gives a sample pair of description sequences—neither of which uses the no-variation generation strategy, and neither of which was included in the evaluation—and asks for demographic information. The demographic information that was requested included age range (under 20, 20-24, 25-29, 30-34, or 35+), gender, level of computer experience (beginner, intermediate, expert). We also requested an email address so that participants could be entered into the prize draw. All of the information was required except for the email address.

After this form was submitted, the experiment began. The experiment materials consisted of eight description sequences, each made up of four individual tile-design descriptions. Participants were shown both versions of a sequence on a web page (Figure C.12), with a small

Figure C.11: Instructions

Tile-design descriptions

In this experiment, we are comparing different ways of creating descriptions of bathroom-tile designs. It should take about **15 minutes** in total to do the whole experiment. Unfortunately, due to the nature of this experiment, you can only participate if you are a **native speaker of English**.

In the experiment, you will see eight sequences of descriptions of tile designs. There will be two versions of each sequence: both versions will give the **same facts** about the **same designs**, but will describe them in different ways. For each sequence, we will ask you a few questions comparing the two versions. Here is an example of the sort of descriptions that you will see:

Sequence 1

Here we have a design in the classic style. It uses tiles from NovaBell's Abbazie collection. It features ochre and beige in the colour scheme. The decorative tiles feature geometric shapes.

Here is a country design. It's based on Aramis, by Aparici. The colour scheme features cream and blue. The decorative tiles feature cross motifs and geometric shapes.

Sequence 2

This design is classic. It is based on the Abbazie collection by NovaBell. The colours are ochre and beige. There are geometric shapes on the decorative tiles.

Here we have a design in the country style. It draws from Aramis, by Aparici. It features cream and blue. The decorative tiles have cross motifs and geometric shapes.

Information about you

Please enter some information about yourself below. The information about your age, gender, and computer experience are required. I will be doing a draw for a **£15 Amazon gift voucher** after the experiment is done; if you want to be entered for the draw, please also enter your email address in the bottom field below. The email address will *only* be used for the prize draw; it will not be associated with your responses in any way, and I promise not to send you any spam. :)

Remember that you can only take part in this experiment if you are a **native speaker of English**.

Age range	(Please choose)	*Required*
Gender	(Please choose)	*Required*
Computer experience	(Please choose)	*Required*
Email address		*Optional, but necessary if you want to be entered in the prize draw*

Begin experiment

When you are ready, please press the button below to begin the experiment.

Start experiment!

Figure C.12: Description-sequence presentation

Text-generation experiment (1 of 8)

Please read these two sequences of descriptions of tile designs. Note that both sequences tell you **same things** about the **same set of designs**; the only difference is in how they describe them.

Sequence 1

This design is classic. It is based on the Opus Romano collection by Bisazza. The colours are beige, black and white. It has mosaics on the decorative tiles.

This design is country. It is based on the Altamira collection by Villeroy and Boch. The colours are sandstone, green and blue. There are floral motifs on the decorative tiles.

This design is classic. It is based on the Blue Jeans collection by Steuler. The colours are white, blue and grey. There are jeans motifs and belt buckle motifs on the decorative tiles.

This design is modern. It is based on the Century collection by Villeroy and Boch. The colours are blue, white and terracotta. There are geometric shapes and floral motifs on the decorative tiles.

Sequence 2

Here we have a design in the classic style. It uses tiles from the Opus Romano collection by Bisazza. The design features beige, black and white. There are mosaics on the decorative tiles.

Here we have a design in the country style. It draws from Villeroy and Boch's Altamira series. The colour scheme features sandstone, green and blue. It has floral motifs on the decorative tiles.

This design is classic. It is based on Steuler's Blue Jeans series. It features white, blue and grey in the colour scheme. It has jeans motifs and belt buckle motifs on the decorative tiles.

Here we have a modern design. It uses tiles from Villeroy and Boch's Century collection. It features blue, white and terracotta. The decorative tiles feature geometric shapes and floral motifs.

Please press this button to answer the questions about these tile descriptions. Answer questions

Figure C.13: Questions about description sequences

Questions about part 1

Please answer all of following questions about the current descriptions:

Which sequence is **easier to understand**?	(Please choose)
Which sequence is **more repetitive**?	(Please choose)
Which sequence is **better written**?	(Please choose)

Submit answers

Figure C.14: Sequence presentation with question window

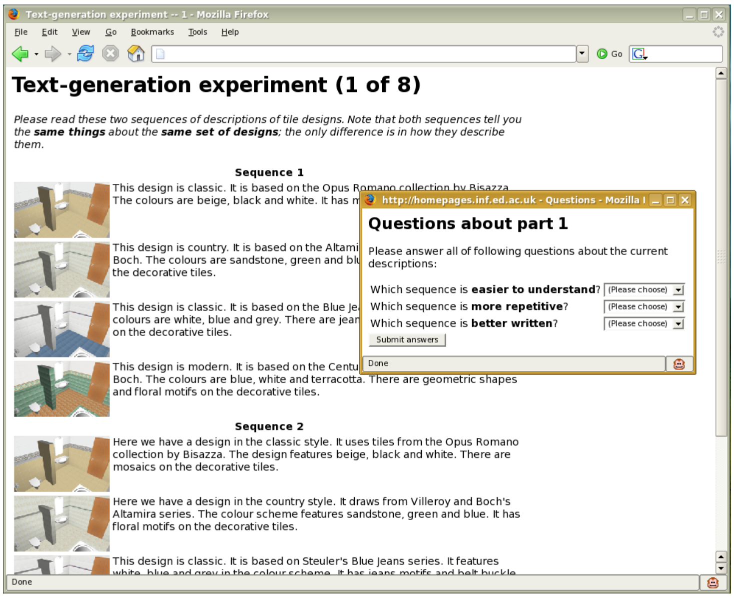


image of the tile design being described next to each description. At the bottom of the page was a *Answer questions* button that the participant pressed to pop up a window displaying three questions about the descriptions.

When the participant pressed the **Answer questions** button, a window popped up on the screen asking three questions about the current pair of description sequences: which was easier to understand, more repetitive, and better written (Figure C.13). For each question, there was a pull-down list containing *Sequence 1* and *Sequence 2*, with no default selected; the form would not submit unless a choice was made for all three of the questions. This window popped up above the main window displaying the description sequences, as shown in Figure C.14.

Figure C.15: Comment form

Text-generation experiment
Final comments

Thank you for taking part in this experiment! If you have any final comments, please enter them here.

Submit comments

After a participant submitted his or her responses for a pair of sequences, the popup window disappeared and the main window showed the two versions of the next sequence. Once a participant had gone through all eight of the sequence pairs, the responses were sent by email and a final web page requested comments about the experiment (Figure C.15). Once the comment form had been submitted, the experiment was over.

C.3 Recognisability of facial displays

This experiment (Section 7.3) measured participants' ability to detect the intended user-model evaluation of a piece of generated text based on the facial displays that were used to present it. It was run over the web.

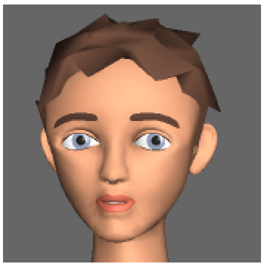
The initial web page for this study is shown in Figure C.16. With minor variations (different experiment numbers, additional demographic questions), this same page was also used as the introductory page for all of the other talking-head experiments. This page first describes the general purpose of the experiment and specifies the technical requirements (a computer with working sound hardware, and a web browser with Javascript and Macromedia Flash installed)). It then allows participants to play a test video to make sure that everything is configured correctly. At the bottom, the page requests demographic information including age range (under 20, 20-24, 25-29, 30-34, or 35+), gender, computer experience (beginner, intermediate, expert), native language (a free-text field), and the participants' email address. All of this information was required except for the email address, which was used only for the prize draw.

When the *Play test* button was clicked, a window popped up playing an embedded Flash video of the RUTH head saying a sentence. The video played in a loop, and the transcript of the sentence was displayed under the video window as shown in Figure C.17. The window could be closed either using the normal window-manager close button or by clicking *Close window*. This basic presentation method was used for all of the talking-head experiments in this thesis.

Figure C.16: Initial page for the talking-head experiments

Talking-head experiment #2

Introduction



In this experiment, we are comparing different ways of generating the movements for a "talking head". It should take about **15-20 minutes** in total to do the whole experiment. The experiment requires the following:

- A computer with a working sound card, and either speakers or earphones
- A web browser with JavaScript enabled, and with the Macromedia Flash plugin installed

In the next section, we will give links to a test video so that you can make sure that your computer is configured properly to run the experiment.

Everyone who completes this experiment will be entered into a draw for **one of two £15 gift certificates from Amazon.co.uk**. If you want to be entered in the draw, please make sure that you enter a valid email address into the form. I promise that the only thing I'll use the address for is the prize draw!

Video test

To play the videos in this experiment, you must have **JavaScript enabled** in your web browser, and you must have the **Macromedia Flash plugin** installed. You might also want to try stopping any other sound-playing programs before playing the video.

Please click the **Play test** button below to make sure that your computer is configured correctly for this experiment. If all goes well, a window should pop up, and you should be able to see and hear a short video of a talking head. The video should play through and then keep repeating. Once you have made sure that you can see and hear the video, you can close the popup window.

Play test

If no window pops up, if you cannot hear or see the video, or if you have any technical questions, please send me email (M.E.Foster@ed.ac.uk). **Please do not start the experiment unless the test video plays correctly.**

Begin experiment

If the test video played successfully, then you are ready to begin the experiment! First, please enter your demographic information below to begin the experiment. I will be doing a draw for two £15 Amazon gift vouchers after the experiment is done; if you want to be entered for the draw, please also enter your email address in the bottom field below. The email address will only be used for the prize draw; it will not be associated with your responses in any way, and I promise not to send you any spam. :)

Age range	(Please choose)	
Gender	(Please choose)	
Computer experience	(Please choose)	
Native language		
Email address		*Optional, but necessary if you want to be entered in the prize draw*

Start experiment!

Figure C.17: Video-playing interface

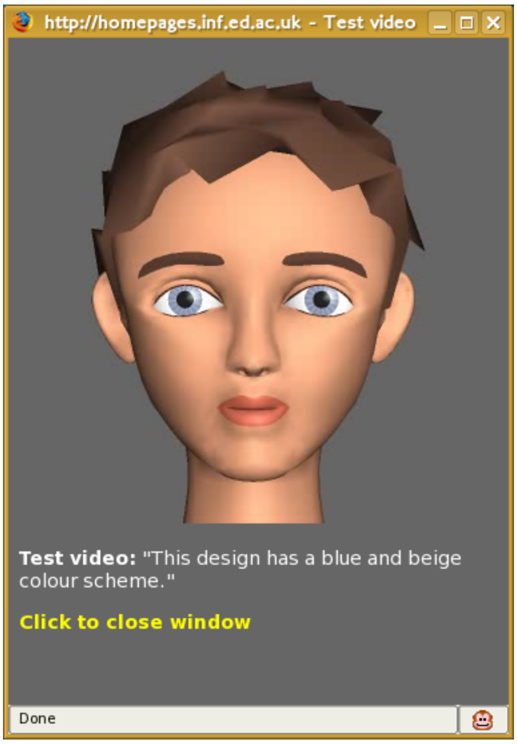
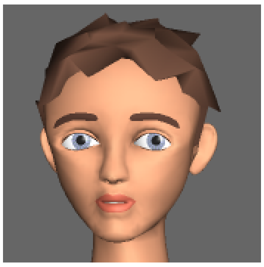


Figure C.18: Main interface for the recognisability study

Talking-head experiment #2



Below are 16 sentences spoken by a talking head; in all of the sentences, the head is describing and comparing different tile designs that you might buy for your bathroom.

You will see a single version of each sentence, which will play in a loop. For each sentence, you should decide whether you believe that the speaker is telling you about something he thinks you **like** or something he thinks you **dislike**, or if you can't tell. For example, if he says *This design has green tiles*, you should decide whether he believes you like or dislike the colour green.

Don't think about it too hard! Go with your first instincts; you shouldn't have to watch a video more than once or twice. Once you have made your decision about a video, please close the window and then choose your response from the list next to that video.

Once you have made a decision on all of the videos, please click the **Submit** button at the bottom of the page to submit your responses. You can also provide any comments you have about the experiment in the Comments box.

Thanks for doing this experiment! Please send me email (M.E.Foster@ed.ac.uk) if you have any questions or concerns.

Video		Like/Dislike	Transcript
1	Play	(Please choose)	"This design features orange in the colour scheme."
2	Play	(Please choose)	"It uses tiles from the Colorado series."
3	Play	(Please choose)	"This design is in the family style."
4	Play	(Please choose)	"It is classic."
5	Play	(Please choose)	"This design draws from Darlington."
6	Play	(Please choose)	"It is by Bisazza."
7	Play	(Please choose)	"This design uses tiles from the Lollipop series."
8	Play	(Please choose)	"It uses tiles from Alessi's Viso series."
9	Play	(Please choose)	"These designs are based on the Altamira collection by Villeroy and Boch."
10	Play	(Please choose)	"It is in the modern style."
11	Play	(Please choose)	"There are abstract shapes and floral motifs on the decorative tiles."
12	Play	(Please choose)	"These designs have a blue and green colour scheme."
13	Play	(Please choose)	"These designs are country."
14	Play	(Please choose)	"These designs use tiles from the Armonie series."
15	Play	(Please choose)	"It has face motifs on the decorative tiles."
16	Play	(Please choose)	"The colour is white."

Comments:

Submit

After the user had entered all of the demographic information and pressed *Start experiment*, the main interface for the experiment was loaded (Figure C.18). This page gave a summary of the experiment at the top and then displayed a list of 16 sentences. For each sentence, there was a *Play* button: when that button was pressed, the selected version of the sentence (with positive, negative, neutral, or no facial displays) was played, using the pop-up window interface shown in Figure C.17. All participants in this experiment saw the same 16 sentences, but each in an individually-chosen random order. As described in Section 7.3, the videos were also randomly allocated so that each participant saw four videos with a positive facial display, four with a negative display, four with a neutral display, and four with no display.

After the video had been played for a sentence, the pulldown list for that sentence was enabled. That list offered three options: *Good*, *Bad*, and *Don't know*; participants were asked to select which of those evaluations the speaker was expressing. The sentences could be processed in any order. There was also a box at the bottom for comments from the participants.

The *Submit* button at the bottom of the page submitted the results of the experiment. If a participant had not made a selection on all of the videos, they were asked to confirm that they intended to submit the partially-completed form. After the responses were submitted, the experiment was over.

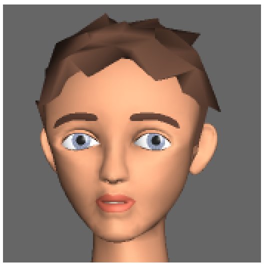
C.4 Consistency of facial displays

This experiment (Section 7.4) measured participants' preferences between videos where the facial displays match the user preference expressed in the text and those where the language and facial displays are inconsistent. Again, it was run over the web.

The initial screen for this experiment was identical to that for the recognition experiment described in the preceding section (Figure C.16). After the participant entered the demographic information and started the experiment, the main page shown in Figure C.19 was displayed. This is again similar to the recognition-study interface shown in Figure C.18, but the task is somewhat different. When the user clicked on *Play*, two different videos were played: both had the same linguistic content, but the facial displays were different. The task for the participant was to decide which of the two versions he or she preferred. As described in Section 7.4, there were three different possible types of facial displays in this study (positive, negative, and neutral), and each participant made every pairwise choice between these types twice in each context (positive and negative). All participants saw the same 12 sentences, but in an individually chosen random order and with a random assignment of comparisons to items.

Figure C.19: Main interface for the consistency study

Talking-head experiment #3



Below are 12 sentences spoken by a talking head; in all of the sentences, the head is describing and comparing different tile designs that you might buy for your bathroom.

For each sentence, we have generated two different possible sequences of head motions. Please click the **Play** button to play the two versions of each video – they should play repeatedly. Watch the two versions as many times as you want until you have decided which one you like better.

Don't think about it too hard! Go with your first instincts; you shouldn't have to watch a video more than once or twice. Once you have made your decision about a video, please close the window and then choose your response from the list next to that video.

Once you have made a decision on all of the videos, please click the **Submit** button at the bottom of the page to submit your responses. You can also provide any comments you have about the experiment in the Comments box.

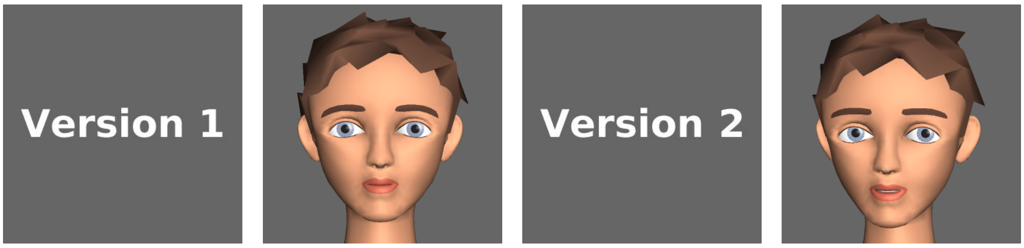
Thanks for doing this experiment! Please send me email (M.E.Foster@ed.ac.uk) if you have any questions or concerns.

Video		**Preferred**	**Transcript**
1	Play	(Please choose)	"You will not like this: the colours are royal blue and white."
2	Play	(Please choose)	"You will like this: it uses tiles from Aparici's Colorado series."
3	Play	(Please choose)	"You will like this: the tiles draw from Creative Amazonas."
4	Play	(Please choose)	"You will like this: it uses tiles from the Century series."
5	Play	(Please choose)	"You will like this: it has a off white and black colour scheme."
6	Play	(Please choose)	"You will not like this: the tiles draw from Armonie, by Sphinx."
7	Play	(Please choose)	"You will not like this: it is classic."
8	Play	(Please choose)	"You will like this: there are geometric shapes and animal motifs on the decorative tiles."
9	Play	(Please choose)	"You will not like this: it has diamond motifs on the decorative tiles."
10	Play	(Please choose)	"You will not like this: it is by Steuler."
11	Play	(Please choose)	"You will like this: it is in the country style."
12	Play	(Please choose)	"You will not like this: it features cream, blue and grey in the colour scheme."

Comments:

Submit

Figure C.20: Video sequence for evaluation



(a) Introduction 1 **(b)** Version 1 **(c)** Introduction 2 **(d)** Version 2

The two videos for each sentence were played as Flash videos using the interface in Figure C.17. To make it clear which version of the video was which, “title cards” were used between the videos as shown in Figure C.20.

As in the previous study, there also was text field for comments at the bottom of the page. When a participant clicked *Submit*, the responses to all of the questions were sent over email. If a participant had not made a selection for all of the items, a confirmation window popped up to ensure that incomplete forms were not inadvertently submitted. After the form was submitted, the experiment was over.

C.5 Comparing methods of selecting facial displays

The instructions and interface for the two experiments described in Section 8.3 and Section 8.4 are essentially identical to those described in the preceding section: again, participants saw two versions of a number of videos and selected which of the two versions they preferred.

For the experiment in Section 8.3, the differences were as follows: on the initial page (Figure C.16), the estimated time was 15–20 minutes and the prize draw was for one of two £25 gift certificates. On the experiment itself, there were a total of 24 sentences, and the pairwise comparisons were between the original corpus annotations, the displays selected by the weighted-choice strategy, and those selected by the majority-choice strategy.

For Section 8.4, the initial estimate was the same, but the prize draw was for one £15 gift certificate. We also added a note to the initial page noting that participants who had not taken part in any of the preceding talking-head evaluations were particularly encouraged and added a mandatory field to the demographic section asking if the participant had taken part in any other studies.

C.6 Facial displays in context

In the experiment in Chapter 9, we examine whether the addition of the embodied agent makes a difference to participants' ability to detect text that is correctly tailored to a specific user model. Again, this experiment was run over the web. The initial page is again similar to that shown in Figure C.16, with three exceptions:

- A note was added to the top indicating that native English speakers were particularly encouraged to take part.
- The experiment is described as *looking at how people respond to descriptions presented by a "talking head"* rather than *comparing different ways of generating the movements for a "talking head"*.
- The prize draw was for one £15 gift certificate.

After a participant had started the experiment, a set of instructions was given as shown in Figure C.21. These instructions described the purpose of the COMIC system and explained that participants were supposed to consider the two possible versions of the system output and select the one that was more appropriate for the user. The *Begin experiment* at the bottom of this window was used to load the main experiment interface.

The interface for the main experiment is shown in Figure C.22. The target user model was shown in the top-right corner throughout the experiment, and a mock-up of the COMIC interface was shown at the left. At the right, each user request was shown in turn, in text. Participants pressed the *Play system responses* button to play the system responses to the user requests. Pressing this button updated the tile-designs displayed at the left and also popped up a window in which the two possible outputs were played. The videos were played using the same Flash-based popup window as in the other experiments (Figure C.17), except that the text of the output was not shown and the video was not played in a loop. Participants could play the videos as many times as they wished.

Below the user request is a pulldown list that the participants used to select the version that they thought was more appropriate—either the first or the second. This pulldown was not enabled until after the videos had been played for a given turn. Once a participant had selected one of the versions for a turn, they pressed *Choose* to store the selection. After a selection was made, the screen updated to show the new user request and the experiment continued in this way until all eight of the turns had been processed.

At the end of the experiment, the responses on all of the items were sent by email, and then a comment form identical to the one shown in Figure C.15 was displayed.

Figure C.21: Instructions

Talking-head experiment #4: Instructions

This experiment is based on a computer system that describes and compares bathroom tile designs. The system can display several tiling designs at once, as shown at the left. There are "thumbnails" of up to five designs across the bottom, while at the top, the user's own bathroom is shown tiled with one of the designs.

Users of the system can ask to get more details about specific designs by choosing one of the thumbnails and saying *Tell me about this one*; they can also ask for designs with specific features by saying things like *Show me designs with blue tiles*. The system responds to requests by changing the selection of designs and by giving descriptions of the designs using a talking head.

In this experiment, you will observe a user interacting with this system. This user has specific tile-design features that they like and dislike; for example, they might like the colour yellow, and dislike designs in the classic style and designs by the manufacturer Viso. These likes and dislikes are displayed on the screen as shown to the right. The user makes a series of 8 requests to the system. For each request, you will see two possible responses, and each time you should choose which of the two is more appropriate for this user.

User preferences

Likes Colour: yellow

Dislikes Style: classic
Manufacturer: Viso

When you are ready to begin the experiment, please click the button below.

[Begin experiment]

Figure C.22: User interface for evaluation experiment

Talking-head experiment #4

Read the request from the user, and then press *Play system responses* to play two possible system responses to that request. Watch and listen to both possible versions and decide which one you think was more appropriate for this user. Remember to look at the **User Preferences** displayed at the right!

Select *Version 1* or *Version 2* from the list once you have decided, and then press *Choose*. After you have made your selection, the next user request will be displayed. There are 8 requests in total.

User preferences

Likes Manufacturer: Aparici
Colour: green
Style: family

Dislikes Colour: brown
Colour: blue
Style: classic
Decoration: geometric shapes
Decoration: mosaics

User request #1 (of 8)

"Tell me about this design *[chooses first design]*"

[Play system responses]

Which system output do you think was **more appropriate for this user**? [(Please choose)] [Choose]

Appendix D

Annotator instructions

The following are the written instructions that were given to the second annotator who processed the face-display corpus as described in Section 6.3.3. Note that these are the initial instructions that were sent; there was additional verbal feedback given on how best to annotate various things after the annotator had processed an initial set of 20 sentences.

D.1 Running the annotation tool

There are two tapes to be annotated. You can start with either one, and go back and forth between them if you want; just use a different output filename for the first and second tapes.

To start the tool: log into a DICE machine and run `/group/project/comic/head-annotation/annotate.sh` with two arguments, as follows:

- First argument: the tape to process, `1` or `2`.
- Second argument: the output file. If the file doesn't exist, it will be created; the parent directory does need to exist in advance, though. You should use a different output file for tape 1 and tape 2.

Note that you can—and probably should!—annotate a tape in multiple sessions. Your partial results will be saved to the output file that you specify, and if you give the same filename the next time you process a tape, you'll be able to pick up where you left off.

When the tool is running, there are four buttons on the bottom. *Play original* plays the original video clip at full speed; *Play original (slow)* plays the original clip at a slower speed; *Play RUTH* plays back the current annotations on the talking head; while *Save* saves the current annotations and loads the next sentence.

D.1.1 Playback mode

I've provided 10 (randomly-selected) sentences from my own annotation that you can take a look at. To see them, give only the single argument `--playback` instead of the tape number and output file. In this mode, you can only look at the annotations; you can't modify them or save the file. The *Save* button is relabelled *Next*; press it to go through the set of sentences one at a time. Unfortunately, there wasn't an easy way to add scrolling backwards, so if you want to take a look at a sentence again, you'll need to close the tool and re-open it ... sorry about that.

D.2 Performing annotations

To process a sentence, you should first play the original video once or twice, to get a feel for what the speaker does with his face. Then, you can right click on any of the nodes in the tree both leaf nodes and internal nodes to attach a facial display to that node (I'll talk more about the facial displays in a second). When you click on a node, you get a pop-up menu with five possible display types:

- Brow: up, down
- Nod: up, down
- Lean: left, right
- Turn: left, right
- Squint: yes

You should attach whatever combination of these that you see in the video. If a motion covers more than one word, then attach it at whatever point it is most appropriate. Ideally, if there is a single node that covers exactly the words that the display is attached to, you should annotate the display on that node. If there is no single node that covers the displays adequately, you can select multiple nodes in the tree by clicking on the first and then using control-click to select the others; if you then right-click on one of the selected nodes and choose a facial display, it will be attached to all of the selected nodes.

If you want to remove a display from a node, you can right click on it and choose *(None)* as its value from the popup menu.

You are allowed to nest displays within one another—e.g., a lean on a phrase and a nod on a word within that phrase. However, you are *not* allowed to nest motions of the same type—e.g., a lean on a word within a phrase where there's already a lean. The tool won't let you do this; the relevant menu item will be disabled in these situations.

Nodes that have some motion associated with them are bold in the tree, while those that don't are in a plain font. You can see what the exact annotations are on a node by holding the mouse over it and reading the tool tip. If the tool tip contains a number (e.g. *BROW=up[0]*), this means that this annotation is part of a multi-node annotation (produced with control-click as described above).

D.3 Playback on talking head

Once you have a proposed annotation, you should press *Play RUTH* to play it back on the RUTH talking head. The goal of this whole process is to make the playback on the talking head as close as possible to the original. There are several factors that make this difficult; the biggest one is that the intonation in the speech synthesis is often not the same as what the speaker used, so—for example—big motions on words like *but* and *and* can look silly when played back with synthesised speech. Please do mark everything that the speaker does (even motions on such words).

After playing back the annotation on the talking head, you should make any modifications that are necessary (e.g., changing the span of annotations, or adding or subtracting displays), and then play back the revised annotation on RUTH.

When you are satisfied that the playback is as close to the original as possible, press *Save* to save the current annotation and load the next one.

D.4 Facial displays

Figure D.1: Neutral expression

There are five facial displays that you can mark. This section gives more detail on each. Figure D.1 shows a sample neutral expression of the speaker. Motions should always be annotated relative to the speaker's position at the start of the sentence; if his head is leaning left at the start of a sentence, as in the figure, and does not lean in either direction during the sentence, then no lean should be annotated.

Motions should be annotated to cover the whole sequence of words where they occur, starting on the first word where there is motion and ending on the word where the motion ends. If a motion covers the entire span under an internal node, please mark it on that internal node, rather than on all of the children (unfortunately, the tool doesn't enforce this; sorry).

The speaker has an annoying tendency to bounce around and fidget while he's speaking; you don't need to capture every single little twitch, just aim for the gross motions, especially with turning and leaning.

D.4.1 Eyebrows

Figure D.2: Eyebrow raise

Eyebrow motions can be annotated as *up* and *down*. Figure D.2 shows a sample eyebrow raise. These are usually fairly easy to spot.

D.4.2 Nodding

Nodding can also be up or down. Sometimes the speaker leans forwards while speaking; this should also be annotated as a downward nod.

D.4.3 Leaning

This can be the most difficult one to annotate, as the speaker has a default left lean. Only annotate leaning if he moves to a different position during a sentence; if his head stays at the same angle for a whole sentence, don't mark any leaning. Left and right refer to the *speaker's* left and right; for example, the motion shown in Figure D.4 should be annotated as a left lean.

Figure D.3: Nodding

(a) Downward nod

(b) Upward nod

Figure D.4: Left lean (+ eyebrow lower)

D.4.4 Turning

Turning is rotating the head from side to side. Again, left and right are with reference to the speaker—Figure D.5 shows a right turn.

D.4.5 Squinting

This is a fairly rare motion, but it does occur. It's sometimes hard to see when it starts and ends because of the speaker's glasses, but do your best. Figure D.6 shows an example.

D.5 Tips

- Occasionally, the original video plays with the audio and video unsynchronised. If this happens, just press *Play Original* again and it should play properly.
- On some sentences, the synthesised words are slightly different than those in the speaker's

Figure D.5: Right turn (+ eyebrow raise)

Figure D.6: Squint (+ brow raise)

script. For example, the speaker says *Alessi* while the talking head says *Alessi Tiles*. In such cases, attach any motions to the corresponding node in the tree. As well, there are some sentences where the synthesis is bad; the word *Palace*, in particular, sometimes comes out sounding like *Paladus*. Again, try to get the motions right and don't worry about the synthesis.

- The nodes in the tree sometimes contain several words; e.g., *Villeroy and Boch*, *geometric shapes*, *once again*, and *decorative tiles* are all single nodes in the tree. If the speaker moves on one part of such a multi-word node, please attach the display to the relevant node.

- Please make sure that, when you add multi-node annotations, you have selected the correct set of nodes; it's somewhat easy to accidentally have extra nodes selected.

Appendix E

Computing agreement measures

In this appendix, we give the detailed computations used to compute the observed and expected disagreement measures for corpora processed by the two annotators as described in Section 6.3.

E.1 Observed disagreement

For a sentence S, we let $A_1(S)$ and $A_2(S)$ represent the sets of facial displays proposed by for that sentence by the two annotators. For an annotation $a_i \in A_i(S), i \in \{1,2\}$, we define $v(a_i)$ as the value of that annotation (e.g., a brow raise), and $L(a_i)$ as the set of leaf nodes covered by that annotation. Then we can compute the observed disagreement $D_o(a_i)$ for annotation a_i as follows (where $j \in \{1,2\}, j \neq i$):

$$D_o(a_i) = \begin{cases} 0 & \text{if } \exists a_j \in A_j(S) : (v(a_j) = v(a_i)) \wedge (L(a_j) = L(a_i)) \\ \frac{1}{3} & \text{if } \exists a_j \in A_j(S) : (v(a_j) = v(a_i)) \\ & \qquad \wedge ((L(a_i) \subset L(a_j)) \vee (L(a_j) \subset L(a_i))) \\ \frac{2}{3} & \text{if } \exists a_j \in A_j(S) : (v(a_j) = v(a_i)) \wedge ((L(a_i) \cap L(a_j)) \neq \emptyset) \\ 1 & \text{otherwise} \end{cases}$$

That is, if the second annotator proposes exactly the same display as the first, the disagreement is 0; if the second annotator has an annotation that covers a superset or subset of the nodes covered by the first annotator's proposal, that is $\frac{1}{3}$; if the displays proposed by the two annotators overlap but with no super/subset relation, that is $\frac{2}{3}$ disagreement; while if there is no corresponding display from the second annotator, the disagreement is 1.

The total disagreement $D_o(S)$ on a sentence S is then computed by summing the disagreement for all annotations proposed by both annotators:[1]

$$D_o(S) = \sum_{i=1}^{2} \sum_{a_i \in A_i(S)} D_o(a_i)$$

E.2 Expected disagreement

To compute the expected disagreement $D_e(S)$, we first compute the probability $p_i(v|l)$ of annotator i assigning a display with value v to a sequence of leaf nodes of of length $l+1$; these probabilities can be straightforwardly counted from the annotated sentences. For a leaf node span (a, b) within a sentence of length n, we compute the probabilities of annotator i assigning a corresponding display on a superset or subset, an overlapping display, and no corresponding display:

$$\begin{array}{rcll}
p_{ss,j}(v,a,b,n) & = & \sum_{c=a}^{b} \sum_{d=a}^{b} p_j(v|(d-c)) & \text{(except } c=a, d=b\text{)} \\
& + & \sum_{c=1}^{a} \sum_{d=b}^{n} p_j(v|(d-c)) & \text{(except } c=a, d=b\text{)} \\
p_{ov,j}(v,a,b,n) & = & \sum_{c=1}^{a-1} \sum_{d=a}^{b-1} p_j(v|(d-c)) & \\
& + & \sum_{c=a+1}^{b} \sum_{d=b+1}^{n} p_j(v|(d-c)) & \\
p_{m,j}(v,a,b,n) & = & 1 - [p_j(v|(b-a)) + p_{ss_j}(v,a,b,n) + p_{ov,j}(v,a,b,n)] &
\end{array}$$

The expected disagreement $D_e(j, v, a, b, n)$ for annotator j on an annotation v assigned by annotator i (where $i \neq j$) is then the sum of the above individual probabilities, weighted by the degree of disagreement:

$$\begin{array}{rcl}
D_e(j,v,a,b,n) & = & \frac{1}{3}[1 - p_j(v|(b-a))] \cdot p_{ss,j}(v,a,b,n) \\
& + & \frac{2}{3}[1 - p_j(v|(b-a)) - p_{ss,j}(v,a,b,n)] \cdot p_{ov,j}(v,a,b,n) \\
& + & 1 \cdot p_{m,j}(v,a,b,n)
\end{array}$$

The total expected disagreement for a sentence S of length n is then computed by summing the expected disagreement for each annotator and for each possible annotation value $v \in V$ (where $i \neq j$):

$$D_e(S) = \sum_{i=1}^{2} \sum_{a=1}^{n} \sum_{b=a}^{n} \sum_{v \in V} p_i(v|(b-a)) \times D_e(j,v,a,b,n)$$

[1] Note that this measure double-counts every subset and overlapping annotation pair; however, D_e is much easier to estimate with disagreement defined in this way.

Appendix F

Detailed analysis of the patterns in the face-display corpus

This appendix describes the contextual features that had the largest effect on each of the facial displays annotated in the corpus described in Chapter 6. To determine these factors, we performed multinomial logit regression on the full set of corpus data, using the R statistical package (Ihaka and Gentleman, 1996), following the procedure described by Fox (2002, chap. 5). For each display, we describe the features that had the largest effect as measured by a χ^2 test on the Wald statistic (regression coefficient divided by standard error); we report only those contextual features whose significance is at least $p < 0.0001$ on this measure.

We considered the features and annotated displays for each node in the OpenCCG derivation tree. The contextual features that were considered for each node were as follows:

- The user-model evaluation of that part of the description (positive, negative, or neutral);
- The predicted pitch accent for the node;
- Whether the fact being described had been said before or not;
- Whether the fact was explicitly compared or contrasted with a feature of a preceding tile design; and
- Whether the node appeared in the first or second clause of a two-node sentence, or whether it was in a clause on its own; and
- The number of leaves spanned by the node.[1]

[1] We used this measure as a proxy for the surface string associated with the node, as the range of possible surface strings is too large for statistical tests to be run.

For each facial display in turn, we describe which of these contextual features had the largest influence on the distribution of the feature, and give an overview of the influence of each of the features. The graphs for each feature indicate the relative frequency of each type of the display in each of the contexts; that is, the percentage of nodes of each type on which that display occurred.

In each section, we describe in detail the significant factors from the corpus annotated by the first annotator, as that is the corpus that was used in the experiments described in this thesis. The significant factors in the sub-corpus processed by the final annotator were generally similar to those of the first annotator; at the end of each section, we summarise the significant factors in this sub-corpus.

F.1 Eyebrow motions

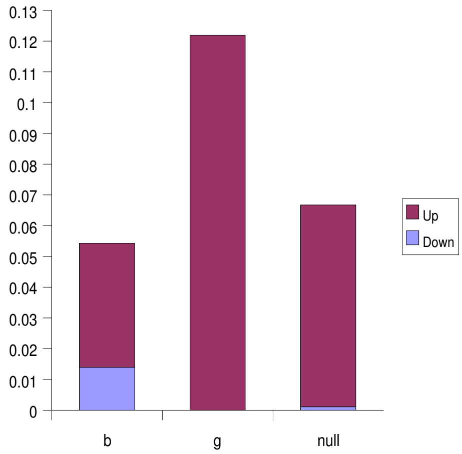
The significant predictors for eyebrow motions were the user-model evaluation, the predicted pitch accent, and the clause of the sentence; there was no significant interaction between these factors. Figure F.1(a) illustrates the influence of the user-model evaluation. In a positive user-model context, eyebrow motions overall were more frequent than in any other context; in negative and neutral contexts, brow motions were approximately equally frequent, but in a negative context the relative frequency of lowering was much higher. As shown in Figure F.1(b), eyebrow motion—especially raising—was more frequent on a node with any sort of predicted pitch accent; the type of pitch accent did not make any difference to the eyebrow motions. When it comes to different clauses of a sentence (Figure F.1(c)), brow motion is most frequent in the first clause and least frequent in the second clause, with single-clause contexts between these two; brow lowering is most frequent in the second clause.

For the final annotator, the user model and the sentence clause were both significant predictors of eyebrow motion, with the same patterns as the first annotator. The pattern of brow raises according to pitch accents is similar for the final annotator, but this was not a significant trend.

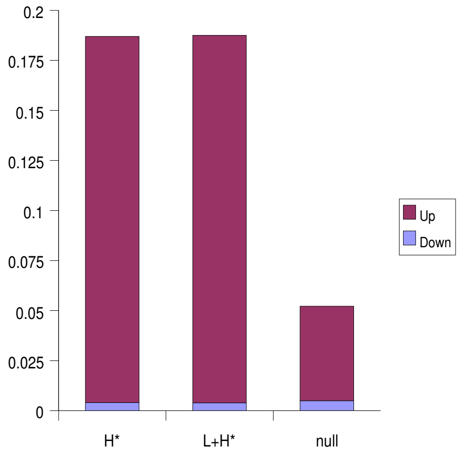
F.2 Eye squinting

Squinting was a very infrequent motion overall; it was annotated on only 0.6% of the nodes in the corpus. It was affected by two main factors: the user-model evaluation of the fact being described (Figure F.2(a)), and the number of leaf nodes spanned by a node (Figure F.2(b)). Squinting was more than four times as frequent in contexts with a negative user-model evaluation than in any other context, and tended to occur on nodes spanning relatively few leaves.

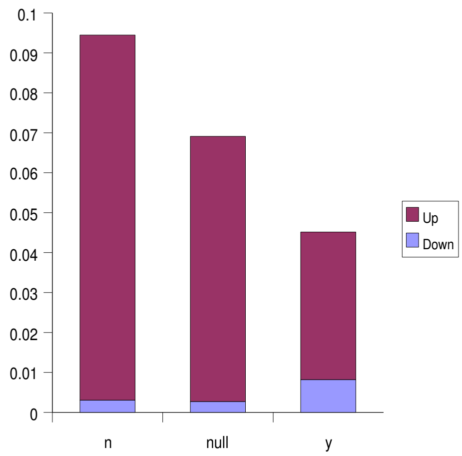
Figure F.1: Distribution of eyebrow motions in different contexts



(a) Brow frequency by user-model evaluation

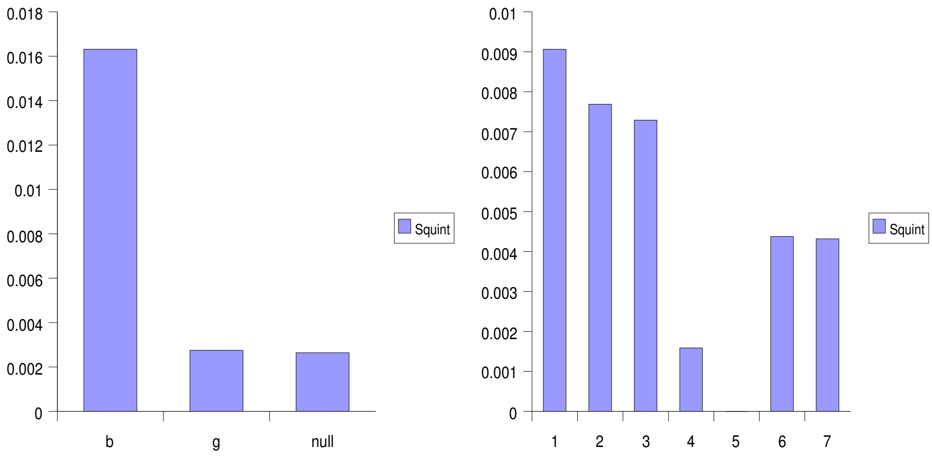


(b) Brow frequency by pitch accent



(c) Brow frequency by sentence clause

Figure F.2: Distribution of squinting



(a) Squint by user model **(b)** Squint by leaves

The final annotator marked even fewer squints than the first—only 0.3% of the nodes; there were so few that no pattern is discernible, but there appear to be relatively more squints on nodes with any sort of user-model evaluation, positive or negative.

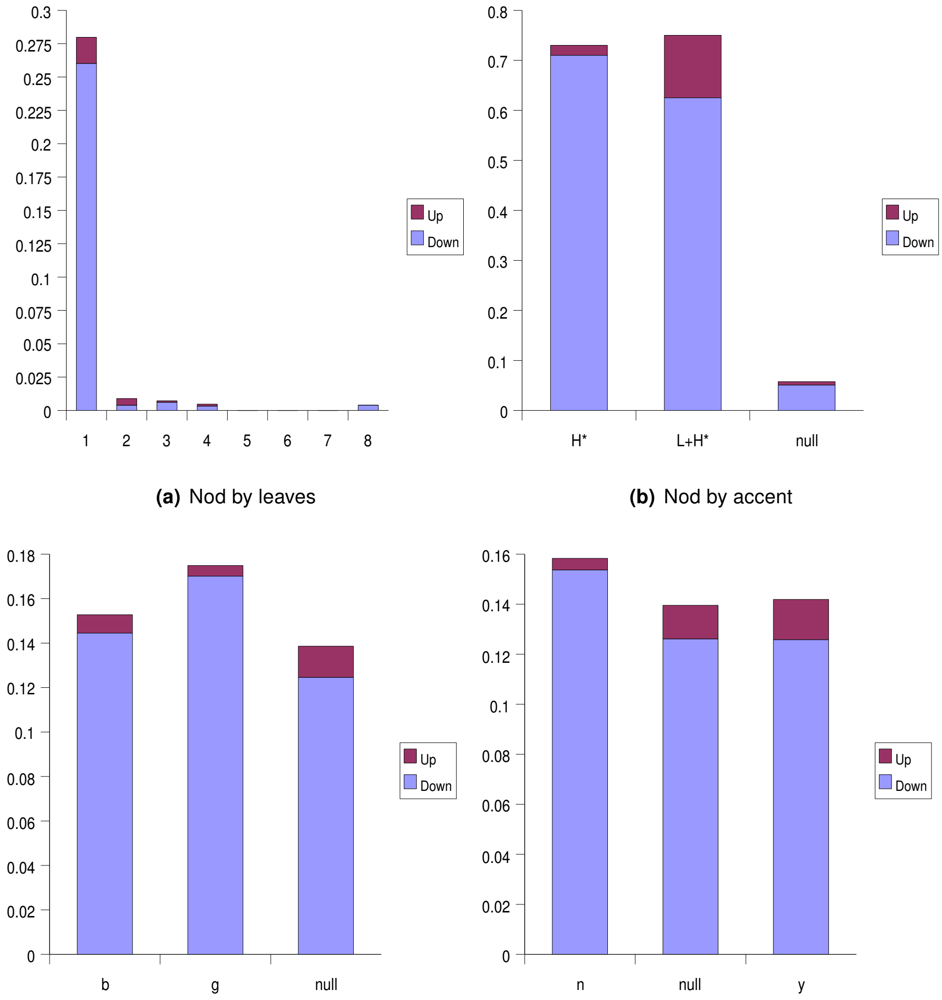
F.3 Nodding

In contrast to squinting, nodding was a relatively frequent behaviour, occurring on 17% of all of the nodes in the corpus. The rate of nodding was influenced by a number of factors. Almost all of the nods annotated in the corpus occurred on single leaf nodes (Figure F.3(a)) and on words with predicted pitch accents (Figure F.3(b)), with relatively more upward nods on $L+H*$ accents. The user model (Figure F.3(c)) and clause (Figure F.3(d)) also had significant effects on the distribution (although less dramatic ones), with more nods on words with positive evaluations and in the first clauses of sentences.

There were also two significant interactions among these factors when it comes to nodding. Figure F.4 shows the interaction of the effects of user-model evaluation and pitch accent on the speaker's nodding behaviour:[2] in a neutral context, upward nods were much more likely on predicted $L+H*$ accents, while the same effect was not seen in a negative context. The clause of the sentence had a similar effect on nodding (Figure F.5); note that, due to the rules used to generate the sentences, $L+H*$ accents tend strongly to be in the first clause of sentences planned by COMIC.

[2] There were no sentences with $L+H*$ accents in a positive user-model context in the corpus.

Figure F.3: Influence of single factors on nodding rate



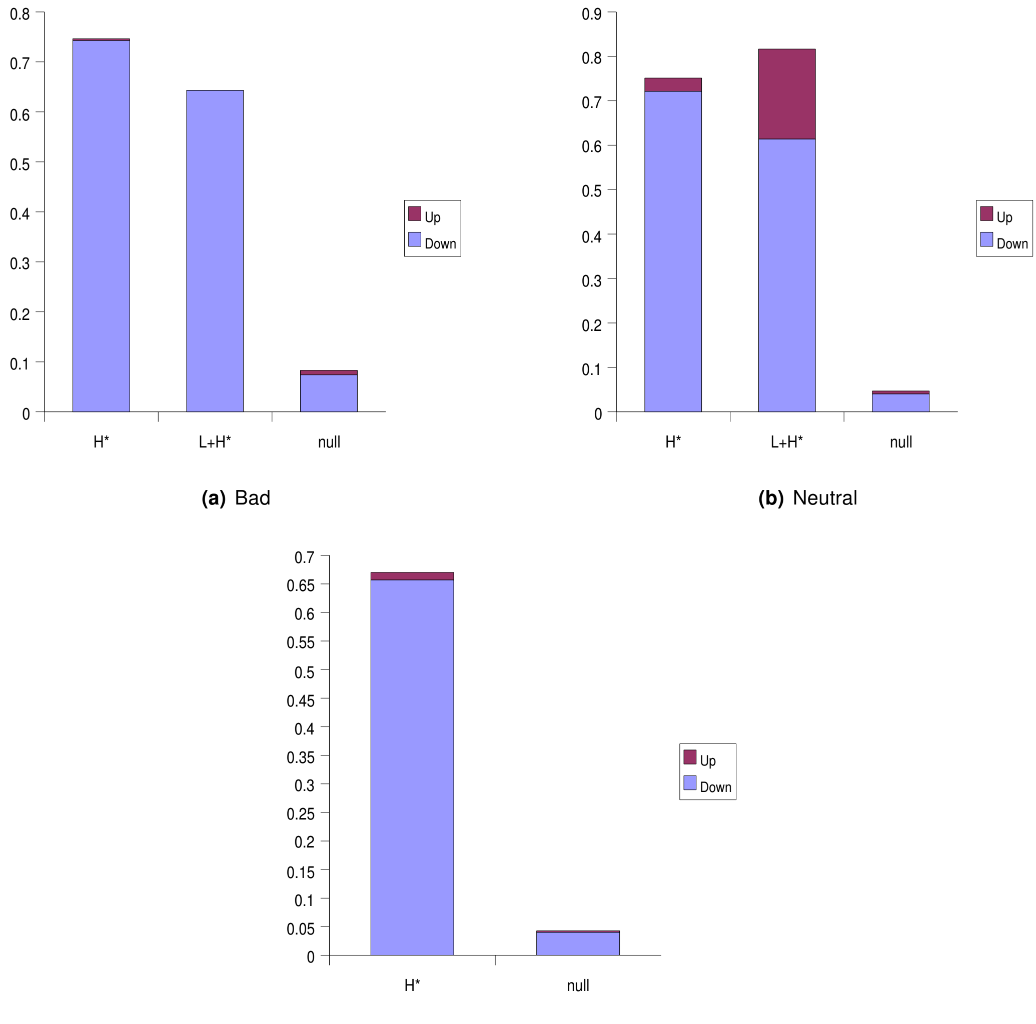
(a) Nod by leaves

(b) Nod by accent

(c) Nod by user model

(d) Nod by clause

Figure F.4: Interaction of user model and accent on nodding



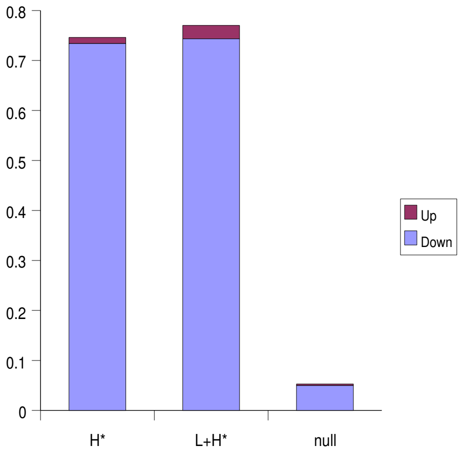
(a) Bad

(b) Neutral

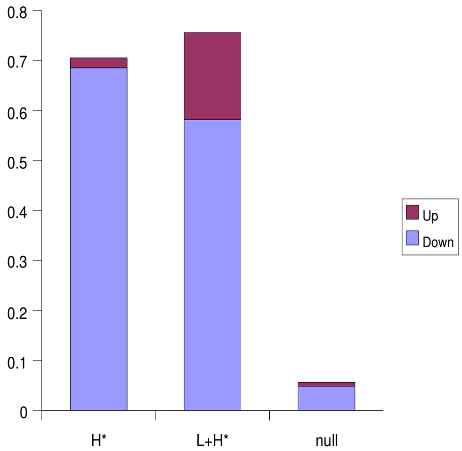
(c) Good

In the final annotator's sub-corpus, the pitch accent and the number of nodes both had a significant effect on the annotated nodding, exactly as described above for the first annotator; the user-model evaluation and the clause had similar effects as for the first annotator, but the effects were not significant.

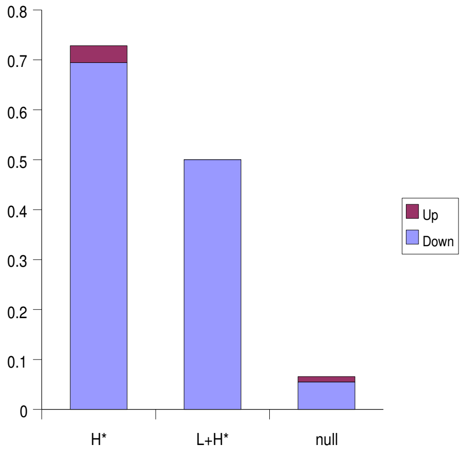
Figure F.5: Interaction of clause and accent on nodding



(a) First

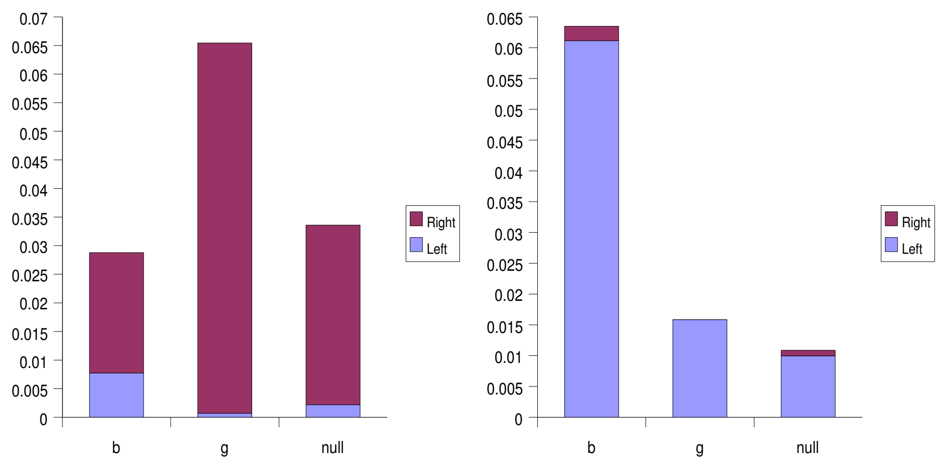


(b) Second



(c) N/A

Figure F.6: Influence of the user model on turning and leaning



(a) Turning by user model **(b)** Leaning by user model

F.4 Turning and leaning

The rates of turning and leaning were both significantly affected by a single contextual factor: the user-model evaluation. This factor had a complementary effect on the two motions: turning was generally to the right and occurred mostly in positive user-model contexts (Figure F.6(a)), while leaning was generally to the left and occurred more frequently in negative user-model contexts (Figure F.6(b)).

For the final annotator, the user model was also a significant factor in the same way for both of these motions; he also annotated mostly right turns and left leans, and in the same user-model distribution. For turns, the number of leaves was also a significant predictor in this sub-corpus, with most turns covering less than three nodes.

Bibliography

Where possible, Digital Object Identifiers (DOIs) or references to the ACL Anthology have been given for all documents that are available online. The DOIs can be resolved at `http://dx.doi.org/`, while the ACL Anthology references can be resolved at `http://www.aclweb.org/anthology/`. In the electronic version of this thesis, the DOIs and ACL references are clickable links.

ÅHLFELDT H, BORIN L, DAUMKE P, GRABAR N, HALLETT C, HARDCASTLE D, KOKKINAKIS D, MANCINI C, MARKÓ K, MERKEL M, PIETSCH C, POWER R, SCOTT D, SILVERVARG A, TOPOROWSKA GRONOSTAJ M, WILLIAMS S, and WILLIS A (2006). Literature review on patient-friendly documentation systems. Technical Report 2006/04, Department of Computing, The Open University.

OP DEN AKKER R, BUNT H, KEIZER S, and VAN SCHOOTEN B (2005). From question answering to spoken dialogue: towards an information search assistant for interactive multimodal information extraction. In *Interspeech 2005*, pp. 2793–2796.

ALLEN J F, BYRON D K, DZIKOVSKA M, FERGUSON G, GALESCU L, and STENT A (2001). Toward conversational human-computer interaction. *AI Magazine*, 22(4):27–38.

ANDRÉ E (2000). The generation of multimedia presentations. In Dale *et al.* (2000), pp. 305–327.

— (2003). Natural language in multimedia/multimodal systems. In R Mitkov, editor, *Handbook of Computational Linguistics*, pp. 650–669. Oxford University Press.

ANDRÉ E and RIST T (1993). The design of illustrated documents as a planning task. In Maybury (1993), pp. 94–116.

ANDROTSOPOULOUS I, OBERLANDER J, and KARKALETSIS V (2007). Source authoring for multilingual generation of personalised object descriptions. *Natural Language Engineering*, 13:191–233. `doi:10.1017/S1351324906004268`.

ARENS Y, HOVY E, and VOSSERS M (1993). On the knowledge underlying multimedia presentations. In Maybury (1993), pp. 280–306. Reprinted in Maybury and Wahlster (1998), pp. 157–172.

ARTSTEIN R and POESIO M (2005). Kappa^3 = alpha (or beta). Technical Report CSM-437, University of Essex Department of Computer Science.

ASHER N and LASCARIDES A (2003). *Logics of Conversation*. Cambridge University Press.

BAILENSON J, AHARONI E, BEALL A, GUADAGNO R, DIMOV A, and BLASCOVICH J (2004). Comparing behavioural and self-report measures of embodied agents' social presence in immersive virtual environments. In *Proceedings of the 7th Annual International Workshop on PRESENCE*.

BAKER R, CLARK R, and WHITE M (2004). Synthesizing contextually appropriate intonation in limited domains. In *Proceedings of the 5th ISCA Workshop on Speech Synthesis*.

BANGALORE S and RAMBOW O (2000a). Corpus-based lexical choice in natural language generation. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*. ACL Anthology P00-1059.

— (2000b). Exploiting a probabilistic hierarchical model for generation. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*. ACL Anthology C00-1007.

BANGALORE S, RAMBOW O, and WHITTAKER S (2000). Evaluation metrics for generation. In *Proceedings of the 1st International Conference on Natural Language Generation (INLG 2000)*. ACL Anthology W00-1401.

BATEMAN J, KAMPS T, KLEINZ J, and REICHENBURGER K (2001). Towards constructive text, diagram, and layout generation for information presentation. *Computational Linguistics*, 27(3):409–449. ACL Anthology J01-3004.

BATEMAN J A and HENSCHEL R (2007). Generating text, diagrams and layout appropriately according to genre. In Theune *et al.* (2007).

BAVELAS J B and CHOVIL N (2000). Visible acts of meaning: An integrated message model of language in face-to-face dialogue. *Journal of Language and Social Psychology*, 19(2):163–194. doi:10.1177/0261927X00019002001.

BECKMAN M E, HIRSHBERG J, and SHATTUCK-HUFNAGEL S (2005). The original ToBI system and the evaluation of the ToBI framework. In S A Jun, editor, *Prosodic Typology: The Phonology of Intonation and Phrasing*. Oxford University Press.

BELZ A (2006). *p*CRU: Probabilistic generation using representational underspecification. Technical Report NLTG-06-01, Natural Language Technology Group, CMIS, University of Brighton.

BELZ A and DALE R, editors (2006). *INLG 2006 Special Session on Sharing Data and Comparative Evaluation*.

BELZ A and REITER E (2005). NLG evaluation. Presentation at ELRA HLT Evaluation workshop, http://www.elra.info/hltevaluationworkshop/.

— (2006). Comparing automatic and human evaluation of NLG systems. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*. ACL Anthology E06-1040.

BELZ A and VARGES S, editors (2005). *Corpus Linguistics 2005 Workshop on Using Corpora for Natural Language Generation*. http://www.itri.brighton.ac.uk/ucnlg/.

BERRY D C, BUTLER L, DE ROSIS F, LAAKSOLATHI J, PELACHAUD C, and STEEDMAN M (2004). Final evaluation report. Deliverable 4.6, MagiCster project.

BICKMORE T and CASSELL J (2005). Social dialogue with embodied conversational agents. In van Kuppevelt *et al.* (2005). doi:10.1007/1-4020-3933-6_2.

BILMES J and KIRCHHOFF K (2003). Factored language models and general parallelized backoff. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*. ACL Anthology N03-2002.

BINSTED K, CAWSEY A J, and JONES R B (1995). Generating personalised patient information using the medical record. In *Proceedings of Artificial Intelligence in Medicine Europe*.

BINSTED K, PAIN H, and RITCHIE G (1997). Children's evaluation of computer-generated punning riddles. *Pragmatics and Cognition*, 5(2):309–358.

BONTCHEVA K and WILKS Y (2001). Dealing with dependencies between content planning and surface realisation in a pipeline generation architecture. In *Proceedings of International Joint Conference in Artificial Intelligence (IJCAI'01)*. Seattle.

TEN BOSCH L (2005). The ASR module in the COMIC demonstrators: Functionality and robustness. Internal deliverable 3.1, COMIC project.

BREIDT M, WALLRAVEN C, CUNNINGHAM D W, and BÜLTHOFF H H (2003). Facial animation based on 3d scans and motion capture. In N Campbell, editor, *Proceedings of SIGGRAPH 2003 Sketches & Applications*. New York, NY: ACM Press.

BUISINE S, ABRILIAN S, NIEWIADOMSKI R, MARTIN J C, DEVILLERS L, and PELACHAUD C (2006). Perception of blended emotions: From video corpus to expressive agent. In *Proceedings of the 6th International Conference on Intelligent Virtual Agents*.

BURGHART C, MIKUT R, STIEFELHAGEN R, ASFOUR T, HOLZAPFEL H, STEINHAUS P, and DILLMANN R (2005). A cognitive architecture for a humanoid robot: a first approach. In *Proceedings of the 5th IEEE-RAS International Conference on Humanoid Robots*, pp. 357–362. doi:10.1109/ICHR.2005.1573593.

CAHILL L, DORAN C, EVANS R, MELLISH C, PAIVA D, REAPE M, SCOTT D, and TIPPER N (1999). In search of a reference architecture for NLG systems. In *Proceedings of the 7th European Workshop on Natural Language Generation (EWNLG 1999)*.

CAHILL L, EVANS R, MELLISH C, PAIVA D, REAPE M, and SCOTT D (2001). The RAGS reference manual. Technical Report ITRI-01-08, ITRI, University of Brighton.

CALLISON-BURCH C, OSBORNE M, and KOEHN P (2006). Re-evaluating the role of BLEU in machine translation research. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*. ACL Anthology E06-1032.

CARBERRY S, CHU-CARROLL J, and ELZER S (1999). Constructing and utilizing a model of user preferences in collaborative consultation dialogues. *Computational Intelligence*, 15(3):185–217. doi:10.1111/0824-7935.00091.

CARCAGNO D and IORDANSKAJA L (1993). Content determination and text structuring: two interrelated processes. In Horacek and Zock (1993), pp. 10–26.

CARENINI G and MOORE J D (2006). Generating and evaluating evaluative arguments. *Artificial Intelligence*, 170(11):925–952. doi:10.1016/j.artint.2006.05.003.

CARLETTA J (2006). Announcing the AMI meeting corpus. *The ELRA Newsletter*, 11(1):3–5. Corpus available through http://corpus.amiproject.org/.

CARLETTA J, EVERT S, HEID U, and KILGOUR J (2005). The NITE XML toolkit: Data model and query. *Language Resources and Evaluation Journal*, 39(4):313–334. doi:10.1007/s10579-006-9001-9.

DE CAROLIS B, CAROFIGLIO V, and PELACHAUD C (2002). From discourse plans to believable behavior generation. In *Proceedings of the 2nd International Conference on Natural Language Generation (INLG 2002)*.

DE CAROLIS B, PELACHAUD C, POGGI I, and STEEDMAN M (2004). APML, a mark-up language for believable behaviour generation. In H Prendinger, editor, *Life-like Characters, Tools, Affective Functions and Applications*, pp. 65–85. Springer.

CARROLL J, MINNEN G, PEARCE D, CANNING Y, DEVLIN S, and TAIT J (1999). Simplifying text for language-impaired readers. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 269–270.

CASSELL J (2000). Nudge nudge wink wink: Elements of face-to-face conversation for embodied conversational agents. In Cassell *et al.* (2000), pp. 1–27.

CASSELL J, BICKMORE T, VILHJÁLMSSON H, and YAN H (2001a). More than just a pretty face: Conversational protocols and the affordances of embodiment. *Knowledge-Based Systems*, 14(1–2):55–64. doi:10.1016/S0950-7051(00)00102-7.

CASSELL J, NAKANO Y, BICKMORE T W, SIDNER C L, and RICH C (2001b). Non-verbal cues for discourse structure. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL 2001)*. ACL Anthology P01-1016.

CASSELL J, SULLIVAN J, PREVOST S, and CHURCHILL E (2000). *Embodied Conversational Agents*. MIT Press.

CASSELL J, VILHJÁLMSSON H H, and BICKMORE T (2001c). BEAT: The Behavior Expression Animation Toolkit. In *Proceedings of the 28th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2001)*, pp. 477–486. doi:10.1145/383259.383315.

CATIZONE R, SETZER A, and WILKS Y (2003). Multimodal dialogue management in the COMIC project. In *Proceedings of the EACL 2003 Workshop on Dialogue Systems: Interaction, adaptation, and styles of management*.

CAVÉ C, GUAÏTELLA I, BERTRAND R, SANTI S, HARLAY F, and ESPESSER R (1996). About the relationship between eyebrow movements and F_0 variations. In *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP 1996)*.

CAWSEY A J, JONES R B, and PEARSON J (2000). The evaluation of a personalised health information system for patients with cancer. *User Modeling and User-Adapted Interaction*, 10(1):47–72. doi:10.1023/A:1008350913145.

CHOVIL N (1991). Discourse-oriented facial displays in conversation. *Research on Language and Social Interaction*, 25:163–194.

CLARK H H and SCHAEFER E F (1992). Dealing with overhearers. In H H Clark, editor, *Arenas of language use*. University of Chicago Press.

CLARK R A J, RICHMOND K, and KING S (2004). Festival 2 – build your own general purpose unit selection speech synthesiser. In *Proceedings of the 5th ISCA Workshop on Speech Synthesis*.

CLEMEN R T (1996). *Making Hard Decisions: An Introduction to Decision Analysis*. Duxbury Press.

COWIE R, DOUGLAS-COWIE E, TSAPATSOULIS N, VOTSIS G, KOLLIAS S, FELLENZ W, and TAYLOR J (2001). Emotion recognition in human-computer interaction. *IEEE Signal Processing Magazine*, 18(1):32–80. doi:10.1109/79.911197.

COX R, O'DONNELL M, and OBERLANDER J (1999). Dynamic versus static hypermedia in museum education: An evaluation of ILEX, the intelligent labelling explorer. In *Proceedings of the 9th International Conference on Artificial Intelligence in Education (AIED 1999)*, pp. 1–8.

CUNNINGHAM D (2005). Implementation of the presentation agent. Internal deliverable 7.3, COMIC project.

CUNNINGHAM D W, KLEINER M, BÜLTHOFF H H, and WALLRAVEN C (2004). The components of conversational facial expressions. In *Proceedings of the 1st Symposium on Applied Perception in Graphics and Visualization (APGV 2004)*, pp. 143–150. doi:10.1145/1012551.1012578.

CUNNINGHAM D W, KLEINER M, WALLRAVEN C, and BÜLTHOFF H H (2005). Manipulating video sequences to determine the components of conversational facial expressions. *ACM Transactions on Applied Perception (TAP)*, 2(3):251–269. doi:10.1145/1077399.1077404.

DALE R (1993). The initial specifications for generation. In Horacek and Zock (1993), pp. 271–278.

DALE R, GREEN S J, MILOSAVLJEVIC M, PARIS C, VERSPOOR C, and WILLIAMS S (1998). The realities of generating natural language from databases. In *Proceedings of the 11th Australian Joint Conference on Artificial Intelligence*. Brisbane.

DALE R, MOISL H, and SOMERS H, editors (2000). *A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text*. Marcel Dekker Inc.

DALE R and REITER E (1995). Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19(2):233–263. doi:10.1207/s15516709cog1902_3.

DAVIDSON R J, EKMAN P, SARON C, SENULIS J, and FRIESEN W (1990). Approach-withdrawal and cerebral asymmetry: Emotional expression and brain physiology I. *Journal of Personality and Social Psychology*, 58(2):330–341.

DAVIDSON R J and IRWIN W (1999). The functional neuroanatomy of emotion and affective style. *Trends in Cognitive Sciences*, 3(1):11–21. doi:10.1016/S1364-6613(98)01265-0.

DECARLO D, STONE M, REVILLA C, and VENDITTI J (2004). Specifying and animating facial signals for discourse in embodied conversational agents. *Computer Animation and Virtual Worlds*, 15(1):27–38. doi:10.1002/cav.5.

VAN DEEMTER K, KRAHMER E, and THEUNE M (2005). Real versus template-based natural language generation: A false opposition? *Computational Linguistics*, 31(1):15–24.

VAN DEEMTER K, VAN DER SLUIS I, and GATT A (2006). Building a semantically transparent corpus for the generation of referring expressions. In *Proceedings of the Fourth International Natural Language Generation Conference*, pp. 130–132. Sydney, Australia. ACL Anthology W06-1420.

DEMBERG V and MOORE J (2006). Information presentation in spoken dialogue systems. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*. ACL Anthology E06-1009.

DI EUGENIO B, FOSSATI D, YU D, HALLER S, and GLASS M (2005). Aggregation improves learning: experiments in natural language generation for intelligent tutoring systems. In *Proceedings of the 43rd Meeting of the Association for Computational Linguistics (ACL 2005)*. Ann Arbor, MI. ACL Anthology P05-1007.

DIMARCO C and FOSTER M E (1997). The automated generation of web documents that are tailored to the individual reader. In *Proceedings, AAAI Spring Symposium on Natural Language Processing on the World Wide Web*. Stanford University.

DiMarco C, Hirst G, Wanner L, and Wilkinson J (1995). HealthDoc: Customizing patient information and health education by medical condition and personal characteristics. In *Proceedings of the Workshop on Artificial Intelligence in Patient Education*. Glasgow.

Dohen M, Lœvenbruck H, Cathiard M A, and Schwartz J L (2004). Visual perception of contrastive focus in reiterant French speech. *Speech Communication*, 44(1–4):155–172. doi:10.1016/j.specom.2004.10.009. Special Issue on Audio Visual speech processing.

Ech Chafaï N, Pelachaud C, and Pelé D (2006). Analysis of gesture expressivity modulations from cartoons animations. In Martin *et al.* (2006).

Edwards W and Barron F H (1994). SMARTS and SMARTER: Improved simple methods for multiattribute utility management. *Organizational Behavior and Human Decision Processes*, 60:306–325. doi:10.1006/obhd.1994.1087.

Egges A, Kshirsagar S, and Magnenat-Thalmann N (2004). Generic personality and emotion simulation for conversational agents. *Computer Animation and Virtual Worlds*, 15(1):1–13. doi:10.1002/cav.3.

Ekman P (1979). About brows: Emotional and conversational signals. In M von Cranach, K Foppa, W Lepenies, and D Ploog, editors, *Human Ethology: Claims and limits of a new discipline*. Cambridge University Press.

— (1999). Basic emotions. In T Dalgleish and T Power, editors, *The Handbook of Cognition and Emotion*, pp. 45–60. John Wiley & Sons, Ltd.

Ekman P and Friesen W (1978). *Facial Action Coding System*. Palo Alto: Consulting Psychologists Press.

Ekman P, Friesen W V, and Hager J C (2002). *Facial Action Coding System*. Salt Lake City: A Human Face.

Ekman P, O'Sullivan M, and Frank M (1999). A few can catch a liar. *Psychological Science*, 10(3):263–266. doi:10.1111/1467-9280.00147.

Elhadad M (1995). Using argumentation in text generation. *Journal of Pragmatics*, 24(1–2):189–220. doi:10.1016/0378-2166(94)00096-W.

Elhadad N, McKeown K, Kaufman D, and Jordan D (2005). Facilitating physicians' access to information via tailored text summarization. In *Proceedings, AMIA Annual Symposium*. Washington, DC.

Engel R (2002). SPIN: Language understanding for spoken dialogue systems using a production system approach. In *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)*.

Erickson D, Fujimura O, and Pardo B (1998). Articulatory correlates of prosodic control: Emotion and emphasis. *Language and Speech*, 41(3–4):399–417.

Feiner S K and McKeown K R (1991). Automating the generation of coordinated multimedia explanations. *IEEE Computer*, 24(10):33–41. doi:10.1109/2.97249. Reprinted in Maybury and Wahlster (1998), pp. 89–98.

Flecha-García M L (2006). *Eyebrow raising in dialogue: Discourse structure, utterance function, and pitch accents*. Ph.D. thesis, Department of Theoretical and Applied Linguistics, University of Edinburgh.

FOSTER M E (2005). Interleaved planning and output in the COMIC fission module. In *Proceedings of the ACL 2005 Workshop on Software*.

— (2007). Issues for corpus-based multimodal generation. In Theune *et al.* (2007).

FOSTER M E, BY T, RICKERT M, and KNOLL A (2006). Human-robot dialogue for joint construction tasks. In *ICMI '06: Proceedings of the 8th international conference on Multimodal interfaces*, pp. 68–71. doi:10.1145/1180995.1181009.

FOSTER M E and OBERLANDER J (2006). Data-driven generation of emphatic facial displays. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*, pp. 353–360. Trento, Italy. ACL Anthology E06-1045.

— (2007). Corpus-based generation of conversational facial displays. *Language Resources and Evaluation*. To appear.

FOSTER M E and WHITE M (2004). Techniques for text planning with XSLT. In *Proceedings of the 4th Workshop on NLP and XML (NLPXML 2004)*. Barcelona, Spain. ACL Anthology W06-1403.

— (2005). Assessing the impact of adaptive generation in the COMIC multimodal dialogue system. In *Proceedings of the IJCAI 2005 Workshop on Knowledge and Reasoning in Practical Dialogue Systems*. Edinburgh, Scotland.

— (2007). Avoiding repetition in generated text. In *Proceedings of the 11th European Workshop on Natural Language Generation (ENLG 2007)*. Schloss Dagstuhl.

FOSTER M E, WHITE M, SETZER A, and CATIZONE R (2005). Multimodal generation in the COMIC dialogue system. In *Proceedings of the ACL 2005 Demo Session*. ACL Anthology W06-1403.

FOX J (2002). *An R and S-Plus companion to applied regression*. Sage Publications.

GAEBEL W and WÖLWER W (2004). Facial expressivity in the course of schizophrenia and depression. *European Archives of Psychiatry and Clinical Neuroscience*, 254(5):335–342. doi:10.1007/s00406-004-0510-5.

GOLDIN-MEADOW S, ALIBALI M, and CHURCH R B (1993). Transitions in concept acquisition: Using the hand to read the mind. *Psychological Review*, 100(2):279–297.

GRAF H, COSATTO E, STROM V, and HUANG F (2002). Visual prosody: Facial movements accompanying speech. In *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2002)*, pp. 397–401. doi:10.1109/AFGR.2002.1004186.

GROSZ B J, WEINSTEIN S, and JOSHI A K (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225. ACL Anthology J95-2003.

GUÉGAN M and HERNANDEZ N (2006). Recognizing textual parallelisms with edit distance and similarity degree. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*. ACL Anthology E06-1036.

GUSTAFSON J, BELL L, BOYE J, EDLUND J, and WIRÉN M (2002). Constraint manipulation and visualization in a multimodal dialogue system. In *Proceedings of the ISCA workshop on Multimodal Interfaces in Mobile Environments*.

HABEL C and ACARTÜRK C (2007). On reciprocal improvement in multimodal generation: Coreference by text and information graphics. In Theune *et al.* (2007).

HALLETT C and SCOTT D (2005). Structural variation in generated health reports. In *Proceedings of the 3rd International Workshop on Paraphrasing (IWP2005)*, pp. 33–40. Jeju Island, Republic of Korea. ACL Anthology I05-5005.

HARTLEY A, SCOTT D, KRUIJFF-KORBAYOVÁ I, SHAROFF S, SOKOLOVA L, DOCHEV D, STAYKOVA K, ČMEJREK M, HANA J, and TEICH E (2000). Evaluation of the final prototype. Technical Report EVAL2, AGILE project.

HELLWEIG B and VAN UYTVANCK D (2006). EUDICO linguistic annotator (ELAN) version 2.6: Manual. http://www.mpi.nl/tools/.

HERZOG G, NDIAYE A, MERTEN S, KIRCHMANN H, BECKER T, and POLLER P (2004). Large-scale software integration for spoken language and multimodal dialog systems. *Natural Language Engineering*, 10:283–305. doi:10.1017/S1351324904003444.

HOFER G (2006). Head motion generation from speech. Unpublished thesis proposal, School of Informatics, University of Edinburgh.

HORACEK H and ZOCK M, editors (1993). *New Concepts in Natural Language Generation: Planning, Realization and Systems*. Pinter Publishers.

HOVY E H (1988). *Generating natural language under pragmatic constraints*. Erlbaum.

HUNT A J and BLACK A W (1996). Unit selection in a concatenative speech synthesis system using a large speech database. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1996)*, volume 1, pp. 373–376. Atlanta, Georgia. doi:10.1109/ICASSP.1996.541110.

IHAKA R and GENTLEMAN R (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314. doi:10.2307/1390807.

ISARD A, BROCKMANN C, and OBERLANDER J (2006). Individuality and alignment in generated dialogues. In *Proceedings of INLG 2006, the International Natural Language Generation Conference*. ACL Anthology W06-1405.

ISARD A, OBERLANDER J, ANDROTSOPOULOS I, and MATHESON C (2003). Speaking the users' languages. *IEEE Intelligent Systems*, 18(1):40–45. doi:10.1109/MIS.2003.1179192.

JOHNSON L, MAYER R, ANDRÉ E, and REHM M (2005). Cross-cultural evaluation of politeness in tactics for pedagogical agents. In *Proceedings of the 12th Int. Conf. on Artificial Intelligence in Education (AIED)*.

JOHNSTON M, EHLEN P, BANGALORE S, WALKER M, STENT A, MALOOR P, and WHITTAKER S (2002). MATCH: An architecture for multimodal dialogue systems. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. ACL Anthology P02-1048.

KARAMANIS N (2004). *Entity Coherence for Descriptive Text Structuring*. Ph.D. thesis, Institute for Communicating and Collaborative Systems, University of Edinburgh.

KARAMANIS N and MELLISH C (2005). A review of recent corpus-based methods for evaluating information ordering in text production. In Belz and Varges (2005).

KARASIMOS A and ISARD A (2004). Multi-lingual evaluation of a natural language generation system. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*.

KAY J (2006). Scrutable adaptation: Because we can and must. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 11 –19. Springer. doi:10.1007/11768012_2.

KEATING P, BARONI M, MATTYS S, SCARBOROUGH R, and ALWAN A (2003). Optical phonetics and visual perception of lexical and phrasal stress in English. In *Proceedings of the 15th International Congress of Phonetic Sciences (ICPhS)*, pp. 2071–2074.

KENDON A (2004). *Gesture: Visible Action as Utterance*. Cambridge University Press.

KIBBLE R (1999). Cb or not Cb? centering theory applied to NLG. In *Proceedings of the ACL 1999 workshop on Discourse and Reference Structure*. ACL Anthology W99-0109.

KIPP M (2004). *Gesture Generation by Imitation - From Human Behavior to Computer Character Animation*. Dissertation.com.

KIPP M, NEFF M, and ALBRECHT I (2006). An annotation scheme for conversational gestures: How to economically capture timing and form. In Martin *et al.* (2006).

KOBSA A and WAHLSTER W, editors (1989). *User models in dialog systems*. Springer.

KOPP S, JUNG B, LESSMANN N, and WACHSMUTH I (2003). Max - a multimodal assistant in virtual reality construction. *KI Zeitschrift (German Magazine of Artificial Intelligence)*.

KRAHMER E and SWERTS M (2004). More about brows: A cross-linguistic study via analysis-by-synthesis. In Pelachaud and Ruttkay (2004), pp. 191–216. doi:10.1007/1-4020-2730-3_7.

— (2005). How children and adults produce and perceive uncertainty in audiovisual speech. *Language and Speech*, 48(1):29–53.

KRAHMER E and THEUNE M (2002). Efficient context-sensitive generation of referring expressions. In K van Deemter and R Kibble, editors, *Information Sharing: Reference and Presupposition in Language Generation and Interpretation*, pp. 223–264. Stanford: CSLI Publications.

KRANSTEDT A, LÜCKING A, PFEIFFER T, RIESER H, and WACHSMUTH I (2006). Deictic object reference in task-oriented dialogue. In G Rickheit and I Wachsmuth, editors, *Situated Communication*. Berlin: Mouton de Gruyter.

KRANSTEDT A and WACHSMUTH I (2005). Incremental generation of multimodal deixis referring to objects. In *Proceedings of the 10th European Workshop on Natural Language Generation (EWNLG 2005)*, pp. 75–82.

VAN KUPPEVELT J, DYBKJÆR L, and BERNSEN N O, editors (2005). *Advances in Natural Multimodal Dialogue Systems*. Kluwer. doi:10.1007/1-4020-3933-6.

LANGKILDE I and KNIGHT K (1998a). Generation that exploits corpus-based statistical knowledge. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL 1998)*. ACL Anthology P98-1116.

— (1998b). The practical value of *n*-grams in generation. In *Proceedings of the 9th International Natural Language Generation Workshop (INLG 1998)*. ACL Anthology W98-1426.

LANGKILDE-GEARY I (2002). An empirical verification of coverage and correctness for a general-purpose sentence generator. In *Proceedings of the 2nd International Natural Language Generation Conference (INLG 2002)*.

LASCARIDES A and STONE M (2006). Formal semantics of iconic gesture. In *BRANDIAL 06: Proceedings of the 10th Workshop on the Semantics and Pragmatics of Dialogue*, pp. 64–71.

LEMON O, GEORGILA K, HENDERSON J, and STUTTLE M (2006). An ISU dialogue system exhibiting reinforcement learning of dialogue policies: Generic slot-filling in the TALK in-car system. In *Proceedings of the EACL 2006 Demo Session*. ACL Anthology E06-2009.

LEMON O, GRUENSTEIN A, and PETERS S (2002). Collaborative activities and multi-tasking in dialogue systems. *Traitement Automatique des Langues (TAL)*, 43(2):131–154.

LIN C Y (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pp. 74–81. Barcelona, Spain. ACL Anthology W04-1013.

MANNING C D and SCHÜTZE H (1999). *Foundations of Statistical Natural Language Processing*. The MIT Press.

MARCINIAK T and STRUBE M (2005). Using an annotated corpus as a knowledge source for language generation. In Belz and Varges (2005).

MARSI E and VAN ROODEN F (2007). Expressing uncertainty with a talking head. In Theune *et al.* (2007).

MARTIN J C, KÜHNLEIN P, PAGGIO P, STIEFELHAGEN R, and PIANESI F, editors (2006). *LREC 2006 Workshop on Multimodal Corpora: From Multimodal Behaviour Theories to Usable Models*. http://wwwhomes.uni-bielefeld.de/mmc06/.

MARTIN J C, DEN OS E, KÜHNLEIN P, PAGGIO P, BOVES L, and CATIZONE R, editors (2004). *LREC 2004 Workshop on Multimodal Corpora: Models Of Human Behaviour For The Specification And Evaluation Of Multimodal Input And Output Interfaces*. http://lubitsch.lili.uni-bielefeld.de/MMCORPORA/.

MAYBURY M T, editor (1993). *Intelligent Multimedia Interfaces*. AAAI Press.

MAYBURY M T (1995). Research in multimedia and multimodal parsing and generation. *Artificial Intelligence Review*, 9(2):103–127. doi:10.1007/BF00849175.

MAYBURY M T and MARTIN J C, editors (2002). *LREC 2002 Workshop on Multimodal Resources and Multimodal Systems Evaluation*. http://www.lrec-conf.org/lrec2002/lrec/wksh/Multimodality.html.

MAYBURY M T and WAHLSTER W, editors (1998). *Readings in Intelligent User Interfaces*. Morgan Kaufmann.

MCCOY K F (1988). Reasoning on a highlighted user model to respond to misconceptions. *Computational Linguistics*, 14(3):52–63. ACL Anthology J88-3005.

MCKEOWN K, ROBIN J, and TANENBLATT M (1993). Tailoring lexical choice to the user's vocabulary in multimedia explanations. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL 1993)*. ACL Anthology P93-1031.

MCKEOWN K R (1985). *Text generation: Using discourse strategies and focus constraints to generate natural language text*. Cambridge University Press.

MCKEOWN K R, CHANG S F, CIMINO J, FEINER S, FRIEDMAN C, GRAVANO L, HATZIVASSILOGLOU V, JOHNSON S, JORDAN D A, KLAVANS J L, KUSHNIRUK A, PATEL V, and TEUFEL S (2001). Persival, a system for personalized search and summarization over multimedia healthcare information. In *JCDL '01: Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries*, pp. 331–340. doi:10.1145/379437.379722.

MCNEILL D (2000a). Catchments and contexts: Non-modular factors in speech and gesture production. In McNeill (2000b), pp. 312–328.

MCNEILL D, editor (2000b). *Language and Gesture: Window into Thought and Action*. Cambridge University Press.

MEERBEEK B, HOONHOUT J, BINGLEY P, and TERKEN J (2006). Investigating the relationship between the personality of a robotic TV assistant and the level of user control. In *Proceedings of RO-MAN 2006 – The 15th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 404–410.

MELLISH C and DALE R (1998). Evaluation in the context of natural language generation. *Computer Speech and Language*, 12(4):349–373. doi:10.1006/csla.1998.0106.

MELLISH C, KNOTT A, OBERLANDER J, and O'DONNELL M (1998). Experiments using stochastic search for text planning. In *Proceedings of the 9th International Workshop on Natural Language Generation (INLG 1998)*. ACL Anthology W98-1411.

MELLISH C, SCOTT D, CAHILL L, PAIVA D, EVANS R, and REAPE M (2006). A reference architecture for natural language generation systems. *Natural Language Engineering*, 12(1):1–34. doi:10.1017/S1351324906004104.

MILOSAVLJEVIC M (1999). *Maximising the Coherence of Descriptions via Comparison*. Ph.D. thesis, Macquarie University, Sydney, Australia.

MILOSAVLJEVIC M and OBERLANDER J (1998). Dynamic hypertext catalogues: Helping users to help themselves. In *Proceedings of the 9th ACM Conference on Hypertext and Hypermedia*, pp. 123–131. doi:10.1145/276627.276641.

MITTAL V O, CARENINI G, and MOORE J D (1994). Generating patient specific explanations in MIGRAINE. In *Proceedings of the eighteenth annual symposium on computer applications in medical care*.

MOORE J, FOSTER M E, LEMON O, and WHITE M (2004). Generating tailored, comparative descriptions in spoken dialogue. In *Proceedings of the 17th International FLAIRS Conference (FLAIRS 2004)*.

MOORE J D and PARIS C L (1993). Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational Linguistics*, 19(4):651–694. ACL Anthology J93-4004.

NIELSEN J and LEVY J (1994). Measuring usability: Preference vs. performance. *Communications of the ACM*, 37(4):66–75. doi:10.1145/175276.175282.

OBERLANDER J and BREW C (2000). Stochastic text generation. *Philosophical Transactions of the Royal Society: Mathematics, Physical and Engineering Sciences*, 358(1769):1373–1387. doi:10.1098/rsta.2000.0592.

O'DONNELL M, KNOTT A, OBERLANDER J, and MELLISH C (2000). Optimising text quality in generation from relational databases. In *Proceedings of the 1st International Conference on Natural Language Generation (INLG 2000)*. ACL Anthology W00-1418.

OVIATT S (1999). Ten myths of multimodal interaction. *Communications of the ACM*, 42(11):74–81. doi:10.1145/319382.319398.

PAPINENI K, ROUKOS S, WARD T, and ZHU W J (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*. ACL Anthology P02-1040.

PARIS C (1988). Tailoring object descriptions to a user's level of expertise. *Computational Linguistics*, 14(3):64–78. ACL Anthology J88-3006.

PASSONNEAU R J (2004). Computing reliability for coreference annotation. In *Proceedings, Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, volume 4, pp. 1503–1506. Lisbon.

PELACHAUD C and RUTTKAY Z, editors (2004). *From Brows to Trust: Evaluating Embodied Conversational Agents*. Kluwer. doi:10.1007/1-4020-2730-3.

PFLEGER N (2004). Context based multimodal fusion. In *Proceedings of the 6th International Conference on Multimodal Interfaces (ICMI 2004)*, pp. 265–272. doi:10.1145/1027933.1027977.

PICKERING M and GARROD S (2004). Toward a mechanistic psychology of dialogue. *Behavioral and Brain Sciences*, 27(2):169–225. doi:10.1017/S0140525X04000056.

PIWEK P (2007). Modality choice for generation of referring acts: Pointing versus describing. In Theune *et al.* (2007).

PIWEK P, POWER R, SCOTT D, and VAN DEEMTER K (2005). Generating multimedia presentations from plain text to screen play. In Stock and Zancanaro (2005), pp. 203–225. doi:10.1007/1-4020-3051-7_10.

POGGI I and PELACHAUD C (2000). Performative facial expressions in animated faces. In Cassell *et al.* (2000), pp. 154–188.

POWER R, SCOTT D, and BOUAYAD-AGHA N (2003a). Document structure. *Computational Linguistics*, 29(2):211–260. ACL Anthology J03-2003.

— (2003b). Generating texts with style. In *Proceedings of the 4th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2003)*, pp. 444–452.

PRENDINGER H, MA C, YINGZI J, NAKASONE A, and ISHIZUKA M (2005a). Understanding the effect of life-like interface agents through users' eye movements. In *Proceedings of the 7th international conference on Multimodal interfaces (ICMI 2005)*, pp. 108–115. doi:10.1145/1088463.1088484.

PRENDINGER H, MORI J, and ISHIZUKA M (2005b). Using human physiology to evaluate subtle expressivity of a virtual quizmaster in a mathematical game. *International Journal of Human-Computer Studies*, 62(2):231–245. doi:10.1016/j.ijhcs.2004.11.009.

REHM M and ANDRÉ E (2005). Catch me if you can – exploring lying agents in social settings. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, pp. 937–944. doi:10.1145/1082473.1082615.

REITER E (1995). NLG vs. templates. In *Proceedings of the 4th European Workshop on Natural Language Generation (EWNLG 1995)*.

REITER E and BELZ A (2006). GENEVAL: A proposal for shared-task evaluation in NLG. In *Proceedings of INLG 2006, the International Natural Language Generation Conference*. ACL Anthology W06-1422.

REITER E and DALE R (2000). *Building Natural Language Generation Systems*. Cambridge University Press. doi:10.2277/052102451X.

REITER E, MELLISH C, and LEVINE J (1995). Automatic generation of technical documentation. *Applied Artificial Intelligence*, 9(3):259–287.

REITER E, ROBERTSON R, and OSMAN L M (2003). Lessons from a failure: Generating tailored smoking cessation letters. *Artificial Intelligence*, 144(1–2):41–58. doi:10.1016/S0004-3702(02)00370-3.

REITER E and SRIPADA S (2002). Should corpora texts be gold standards for NLG? In *Proceedings of the 2nd International Language Generation Conference (INLG 2002)*.

RICH E (1979). User modeling via stereotypes. *Cognitive Science*, 3:329–354.

RIST T (2005). Supporting mobile users through adaptive information presentation. In Stock and Zancanaro (2005), pp. 113–139. doi:10.1007/1-4020-3051-7_6.

RITCHIE G (2005). Computational mechanisms for pun generation. In *Proceedings of the 10th European Workshop on Natural Language Generation (EWNLG 2005)*.

ROCHA N F (2004). *Evaluating automatic assignment of prosodic structure for speech synthesis*. Master's thesis, Department of Theoretical and Applied Linguistics, University of Edinburgh.

DE ROSIS F, PELACHAUD C, POGGI I, CAROFIGLIO V, and CAROLIS B D (2003). From Greta's mind to her face: Modelling the dynamics of affective states in a conversational embodied agent. *International Journal of Human-Computer Studies*, 59(1–2):81–118. doi:10.1016/S1071-5819(03)00020-X.

ROSSIGNOL S and VUURPIJL L (2005). Functionalities of the pen interpreter module – T36. Internal deliverable 3.2, COMIC project.

ROTH S F and MATTIS J (1990). Data characterization for intelligent graphics presentation. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI 1990)*, pp. 193–200. doi:10.1145/97243.97273. Reprinted in Maybury and Wahlster (1998), pp. 194–203.

DE RUITER J P (2007). Some multimodal signals in humans. In Theune *et al.* (2007).

DE RUYTER B, SAINI P, MARKOPOULOS P, and VAN BREEMEN A (2005). Assessing the effects of building social intelligence in a robotic interface for the home. *Interacting with Computers*, 17(5):522–541. doi:10.1016/j.intcom.2005.03.003.

SCOTT D and MOORE J (2006). An NLG evaluation competition? Eight reasons to be cautious. Technical Report 2006/09, Computing Department, The Open University.

SHIMODAIRA H, UEMATSU K, KAWAMOTO S, HOFER G, and NAKAI M (2005). Analysis and synthesis of head motion for lifelike conversational agents. In *Proceedings of the 2nd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI 2005)*.

SIDNER C L, LEE C, KIDD C D, LESH N, and RICH C (2005). Explorations in engagement for humans and robots. *Artificial Intelligence*, 166(1–2):140–164. doi:10.1016/j.artint.2005.03.005.

VAN DER SLUIS I F (2005). *Multimodal Reference: Studies in Automatic Generation of Multimodal Referring Expressions*. Ph.D. thesis, University of Tilburg.

SRIPADA S and GAO F (2007). Summarising dive computer data: A case study in integrating textual and graphical presentations of numerical data. In Theune *et al.* (2007).

SRIPADA S G, REITER E, and HAWIZY L (2005). Evaluation of an NLG system using post-edit data: Lessons learnt. In *Proceedings of the 10th European Workshop on Natural Language Generation (EWNLG 2005)*.

STEEDMAN M (2000a). Information structure and the syntax-phonology interface. *Linguistic Inquiry*, 31(4):649–689. doi:doi:10.1162/002438900554505.

— (2000b). *The Syntactic Process*. MIT Press.

STENT A, MARGE M, and SINGHAI M (2005). Evaluating evaluation methods for generation in the presence of variation. In *Computational Linguistics and Intelligent Text Processing*, *Lecture Notes in Computer Science*, volume 3406/2005, pp. 341–351. Springer. doi:10.1007/b105772.

STOCK O and ZANCANARO M, editors (2005). *Multimodal Intelligent Information Presentation*. Kluwer. doi:10.1007/1-4020-3051-7.

STONE M, DECARLO D, OH I, RODRIGUEZ C, LEES A, STERE A, and BREGLER C (2004). Speaking with hands: Creating animated conversational characters from recordings of human performance. *ACM Transactions on Graphics (TOG)*, 23(3):506–513. doi:10.1145/1015706.1015753. Special Issue: Proceedings of the 2004 SIGGRAPH Conference.

SWERTS M and KRAHMER E (2004). Congruent and incongruent audiovisual cues to prominence. In *Proceedings of Speech Prosody 2004*.

— (2007). Facial expressions and prosodic prominence: Comparing modalities and facial areas. *Journal of Phonetics*. In press.

THEUNE M, KRAHMER E, VAN DER SLUIS I, and REITER E, editors (2007). *Workshop on Multimodal Output Generation (MOG 2007)*. Aberdeen, Scotland. http://www.csd.abdn.ac.uk/mog2007/.

THOMPSON W F and RUSSO F A (2006). Facial expressions of pitch structure. In *Proceedings of the 9th International Conference on Music Perception and Cognition*. Bologna, Italy.

VARGES S and MELLISH C (2001). Instance-based natural language generation. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL 2001)*. ACL Anthology N01-1001.

W3C (1999). XSL transformations (XSLT) version 1.0. Recommendation, 16 November 1999, World Wide Web Consortium. http://www.w3.org/TR/1999/REC-xslt-19991116.

— (2001). DAML+OIL (March 2001) reference description. Note, 18 December 2001, World Wide Web Consortium. http://www.w3.org/TR/daml+oil-reference.

WAHLSTER W, editor (2000). *Verbmobil: Foundations of Speech-to-Speech Translation*. Springer.

— (2006). *SmartKom: Foundations of Multimodal Dialogue Systems*. Springer. doi:10.1007/3-540-36678-4.

WALKER M (2007). Share and share alike: Resources for language generation. In White and Dale (2007). To appear.

WALKER M, RAMBOW O, and ROGATI M (2002). Training a sentence planner for spoken dialogue using boosting. *Computer Speech and Language*, 16(3–4):409–433. doi:10.1016/S0885-2308(02)00027-X.

WALKER M, WHITTAKER S, STENT A, MALOOR P, MOORE J, JOHNSTON M, and VASIREDDY G (2004). Generation and evaluation of user tailored responses in multimodal dialogue. *Cognitive Science*, 28(5):811–840. doi:10.1016/j.cogsci.2004.06.002.

WALKER M A (2005). Can we talk? Methods for evaluation and training of spoken dialogue systems. *Language Resources and Evaluation*, 39(1):65–75. doi:10.1007/s10579-005-2696-1.

WALKER M A, CAHN J E, and WHITTAKER S J (1997a). Improvising linguistic style: Social and affective bases for agent personality. In *Proceedings of the Conference on Autonomous Agents (AGENTS97)*.

WALKER M A, LITMAN D, KAMM C A, and ABELLA A (1997b). PARADISE: A general framework for evaluating spoken dialogue agents. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL/EACL 1997)*. ACL Anthology P97-1035.

WAN S, DALE R, DRAS M, and PARIS C (2005). Statistically generated summary sentences: A preliminary evaluation using a dependency relation precision metric. In Belz and Varges (2005).

WEGENER KNUDSEN M, MARTIN J C, DYBKJÆR L, MACHUCA AYUSO M J, BERNSEN N O, CARLETTA J, HEID U, KITA S, LLISTERRI J, PELACHAUD C, POGGI I, REITHINGER N, VAN ELSWIJK G, and WITTENBURG P (2002). Survey of multimodal annotation schemes and best practice. Deliverable 9.1, ISLE Project.

WHITE M (2004). Reining in CCG chart realization. In *Proceedings of the 3rd International Conference on Natural Language Generation (INLG 2004)*.

— (2005). Designing an extensible API for integrating language modeling and realization. In *Proceedings of the ACL 2005 Workshop on Software*.

— (2006a). CCG chart realization from disjunctive inputs. In *Proceedings of the 4th International Conference on Natural Language Generation (INLG-06)*. ACL Anthology W06-1403.

— (2006b). Efficient realization of coordinate structures in Combinatory Categorial Grammar. *Research on Language and Computation*, 4(1):39–75. doi:10.1007/s11168-006-9010-2.

WHITE M and DALE R, editors (2007). *NAACL 2007 Workshop on Shared Tasks and Comparative Evaluation in Natural Language Generation*. http://www.ling.ohio-state.edu/~mwhite/nlgeval07/.

WHITE M, FOSTER M E, OBERLANDER J, and BROWN A (2005). Using facial feedback to enhance turn-taking in a multimodal dialogue system. In *Proceedings of HCI International 2005 Thematic Session on Universal Access in Human-Computer Interaction*.

WHITTAKER S and WALKER M (2005). Evaluating dialogue strategies in multimodal dialogue systems. In W Minker, D Bühler, and L Dybkjær, editors, *Spoken Multimodal Human-Computer Dialogue in Mobile Environments*. Kluwer Academic Publishers. doi:10.1007/1-4020-3075-4_14.

WILCOCK G (2001). Pipelines, templates and transformations: XML for natural language generation. In *Proceedings of the 1st Workshop on NLP and XML (NLPXML 2001)*.

WILKINS D P (2003). Why pointing with the index finger is not a universal (in sociocultural and semiotic terms). In S Kita, editor, *Pointing: Where Language, Culture and Cognition Meet*, pp. 171–215. Lawrence Erlbaum.

WILKS Y (1990). Where am I coming from: The reversability of analysis and generation in natural language processing. In M Pütz, editor, *Thirty Years of Linguistic Evolution*. John Benjamins Publishing Company.

WILLEMS D J M, ROSSIGNOL S, and VUURPIJL L G (2005). Features for mode detection in natural online pen input. In *Proceedings of the 12th Conference of the International Graphonomics Society (IGS 2005)*.

WILLIAMS S and REITER E (2005). Generating readable texts for readers with low basic skills. In *Proceedings of the 10th European Workshop on Natural Language Generation (EWNLG 2005)*.

YEH C L and MELLISH C (1997). An empirical study on the generation of anaphora in chinese. *Computational Lingustics*, 23(1). ACL Anthology J97-1007.

Young S, Kershaw D, Odell J, Ollason D, Valtchev V, and Woodland P (2000). *The HTK Book Version 3.0*. Cambridge University Press.

Zinn C, Moore J D, and Core M G (2005). Intelligent information presentation for tutoring systems. In Stock and Zancanaro (2005), pp. 227–252. doi:10.1007/1-4020-3051-7_11.

Zukerman I and Litman D (2001). Natural language processing and user modeling: Synergies and limitations. *User Modeling and User-Adapted Interaction*, 11(1–2):129–159. doi:10.1023/A:1011174108613.

Zukerman I and McConachy R (1993). Consulting a user model to address a user's inferences during content planning. *User Modeling and User-Adapted Interaction*, 3(2):155–185. doi:10.1007/BF01099728.