Chapter 14

EVALUATING DIALOGUE STRATEGIES IN MULTIMODAL DIALOGUE SYSTEMS

Steve Whittaker, Marilyn Walker
University of Sheffield, Sheffield, United Kingdom
{s.whittaker,m.a.walker}@sheffield.ac.uk

Abstract Previous research suggests that multimodal dialogue systems providing both speech and pen input, and outputting a combination of spoken language and graphics, are more robust than unimodal systems based on speech or graphics alone (Andr´e, 2002; Oviatt, 1999). Such systems are complex to build and signifi cant research and evaluation effort must typically be expended to generate well-tuned modules for each system component. This chapter describes experiments utilising two complementary evaluation methods that can expedite the design process: (1) a *Wizard-of-Oz* data collection and evaluation using a novel Wizard tool we developed; and (2) an *Overhearer* evaluation experiment utilising logged interactions with the real system. We discuss the advantages and disadvantages of both methods and summarise how these two experiments have informed our research on dialogue management and response generation for the multimodal dialogue system MATCH.

Keywords: User modelling; Natural language generation; Wizard-of-Oz experiments; Overhearer method; User-adaptive generation; Multiattribute decision theory.

1. Introduction

Multimodal dialogue systems promise users mobile access to a complex and constantly changing body of information. However, mobile information access devices such as PDAs, tablet PCs, and next-generation phones offer limited screen real-estate and no keyboard or mouse. Previous research suggests that spoken language interaction is highly desirable for such systems, and that systems that provide *both* speech and pen input, and that output a combination of spoken language and graphics, are more robust than unimodal systems (André, 2002; Oviatt, 1999). However, such systems are complex to build and typically significant research and evaluation effort must be expended

to generate well-tuned modules for each system component. Furthermore, during the development process, it is necessary to evaluate individual components to inform the design process before the whole system is robust enough for data collection with real users. This chapter describes experiments utilising two complementary evaluation methods that can be applied to collect information useful for design during the design process itself. We summarise how we have used these methods to inform our research on improved algorithms for (a) dialogue management and (b) generation for information presentation in multimodal dialogue.

Our testbed application is MATCH (Multimodal Access To City Help), a dialogue system providing information for New York City (Johnston and Bangalore, 2000; Bangalore and Johnston, 2000; Johnston and Bangalore, 2001; Johnston et al., 2002). MATCH runs standalone on a Fujitsu PDA, as shown in Figure 1, yet can also run in client-server mode across a wireless network. MATCH provides users with mobile access to restaurant, entertainment and transportation information for New York City (NYC). Figure 2 depicts the multimodal architecture supporting MATCH, which consists of a series of agents which communicate through a facilitator MCUBE. Figure 2 shows modules that support users in specifying inputs via speech, gesture, handwriting or by a combination of these. Other modules support output generated in speech, using a graphical display, or a combination of both these modes. Automatic Speech Recognition (ASR) is provided by AT&T's Watson engine (Sharp et al., 1997), and the Text-To-Speech (TTS) is based on AT&T's Natural Voices (Beutnagel et al., 1999). MATCH uses a finite-state approach (MMFST) to parse, integrate, and understand multimodal and unimodal inputs (Johnston and Bangalore, 2000). See (Johnston et al., 2002) for more architectural detail.

Our primary research focus has been to provide MATCH with improved capabilities for dialogue management and generation during the information presentation portion of the dialogue, i.e., research on the Multimodal Dialogue Manager (MDM), Text Planner and Multimodal Generator in Figure 2. During this dialogue phase, the system retrieves from its database a set of options that match the user's constraints. The user must then evaluate the various options before selecting one. Even in a multimodal system such as MATCH, that displays some information graphically, this is a complex and time-consuming process: the user must browse a list or graphical representation of the options and access information about each one.

For example, consider a user's request to *Show Italian Restaurants in the West Village*. Figure 3 shows the large number of highlighted options generated as a graphical response. To make an informed choice, the user has to access more detailed information about each individual restaurant either with speech or by graphical browsing. In addition to the tedium of sequentially

Figure 1. MATCH running on a Fujitsu PDA.

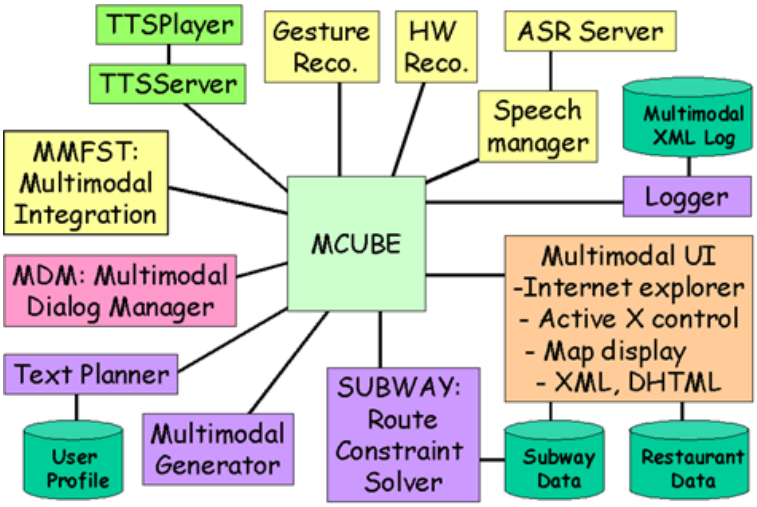

Figure 2. Multimodal architecture.

accessing the set of retrieved options, it may also be hard for users to remember information relevant to making a decision. Thus our research focus has been this critical information presentation problem. Specifically we attempt to devise improved algorithms for: (1) **Option Selection**: selecting the most relevant subset of options to mention or highlight, and (2) **Content Selection**:

choosing what to say about them (Walker et al., 2002; Stent et al., 2002; Whittaker et al., 2002).

Figure 3. MATCH's graphical system response to "Show me Italian Restaurants in the West Village."

Our solution to this problem is to try to reduce the amount and complexity of the information, by focusing on identifying: (a) the options that are relevant to the user; (b) ensuring that we restrict the information we present about the options to attributes that are of direct interest to the user. Our hypothesis is that a solution to these problems can be provided by quantitative user models of user preferences based on multi-attribute decision theory (Keeney and Raiffa, 1976; Edwards and Barron, 1994). We base our dialogue strategies on the user model when providing information (Carenini and Moore, 2000; Carenini and Moore, 2001). The model of user preferences addresses the content selection problem by providing a means to rank the options returned by the database, predicting that the most relevant subset of options are those that are most highly ranked. The same model also addresses the content selection problem by identifying the attributes that should be mentioned when describing the option, namely those attributes that are predicted by the model to be most convincing to a particular user.

We have followed an iterative design approach. It has been necessary to evaluate different potential designs for response generation and dialogue management before the whole system was robust enough for data collection with real users. This chapter describes experiments utilising two complementary evaluation methods that can be applied to the design process that allow evaluation of system components before the whole system is built: (1) a *Wizard-of-Oz* data collection and evaluation using a novel Wizard tool we developed; and (2) an *Overhearer* evaluation experiment utilising logged interactions with the MATCH system. We describe in detail our Wizard Interaction Tool (WIT) and the required functionality of such a tool, as well as the design of the Overhearer evaluation. We discuss the advantages and limitations of both methods. We then summarise how these two experiments have informed our research on dialogue management and response generation for the MATCH system.

The structure of the chapter is as follows. Section 2 provides details about requirements for, and design of the Wizard Interface Tool, describes how we used the tool to collect data during the early design phase of the system, and summarises what we learned from the Wizard experiment. Section 3 describes the Overhearer evaluation paradigm that we developed to support evaluation experiments on response generation in spoken dialogue systems and summarises the findings from this experiment. Section 4 compares what can be learned from these two evaluation methodologies and summarises the chapter.

2. Wizard-of-Oz Experiment

One problem in developing information presentation algorithms for complex information gathering dialogues is that we lack detailed information about (a) the types of tasks and the strategies employed by users when seeking information and (b) the techniques employed by human experts in providing information about those complex domains. We also wanted to explore the effects of user models on users' information seeking behaviour, and various strategies for the presentation of complex information. The Wizard-of-Oz technique is well-suited to generating corpora from which information about tasks and strategies can be derived. It also allows us to collect early data about the effectiveness of various information presentation strategies without actually implementing these in a working system. The goals of the Wizard-of-Oz experiment were to: (a) acquire user models (b) generate representative domain tasks (c) understand the requirements for designing and building a Wizard interface; (d) evaluate dialogue strategies that the Wizard could perform in real time that exploit the user models; and (e) collect sample dialogues to explore the utility of the user model/dialogue strategy combination, which could also be used as training material for the spoken language understanding components of the system.

We recruited a set of 18 users via email who frequently go out to dinner and had some familiarity with the neighbourhoods and restaurants in NYC. We told the users that they would be interacting with a Wizard (referred to as "Monsieur Croque"), who would simulate the functionality and strategies of the real dialogue system. We told them that the Wizard had access to information about thousands of restaurants in the New York area, derived from Zagat's reviews. Zagat's is a popular US food guide. It is based on feedback and reviews from ordinary people who have eaten at restaurants and who volunteer their views. A Zagat's entry for a restaurant includes the following types of information: food type, food ratings, locations, prices, service, decor, along with restaurant reviews that are composites of comments made by different Zagat survey participants.

2.1 Generating Representative Tasks and Acquiring User Models

We gave 18 users an illustrative example of the types of information available for all restaurants in our database and asked them to generate two sample task scenarios, according to the following description: *A scenario should be a description of a set of characteristics that would help Mr. Croque find a small set of restaurants that would match your description. Our goal is to examine the process by which you and Mr. Croque jointly identify a restaurant that you want to eat at, so please do not select a particular restaurant in advance.* The initial instructions and scenario generation were carried out in email. Fifteen users responded with sample task scenarios. Two such tasks (MS and CK) are shown in Figure 4; a dialogue generated for the CK task is shown in Figure 5. There are several points to note about the tasks. First, task scenarios often specify particular situational constraints, e.g., meal cost of $50 in MS, which override more general dispositional preferences, (for example that MS would generally only pay $30 for a meal). Second, scenarios often mention information which is not generally available from our database, e.g., wine selection in CK. One well documented problem with this type of system lies in specifying to the user exactly what data the system knows about.

Our next step was to determine a user model for each person. In order to define a multi-attribute decision model for the restaurant domain, we first had to determine the attributes and their relative weighting for particular users. Edwards and Barron describe a procedure called SMARTER for eliciting multi-attribute decision models for particular users or user groups (Edwards and Barron, 1994). This method requires users to rank attributes. It takes only a few minutes and has been shown to result in high accuracy user models (Edwards and Barron, 1994).

USER	TASK
MS	We want to go to the Indian restaurant with the best cuisine and the best service in walking distance of the Broadway theater district. We can't eat before 6, and we need to be able to leave the restaurant by 7:30 to make an 8 p.m. show near Times Square. Don and I will both arrive separately via subway, so transportation isn't an issue. We're willing to pay up to $50 each for the meal, including drinks.
CK	I'm going to see the play Chicago on May 19. It is at the Shubert Theatre. I'm going to the matinee. Since this is a birthday celebration, we want to go out for an early dinner afterwards. I think a French restaurant in walking distance from there would be nice. My friends are wine experts, so it would be good if there was an impressive wine selection. I'm not too worried about the price, but I don't want to have to mortgage my house for this meal.

Figure 4. Two sample tasks from users MS and CK.

We elicited models for 18 users; the models are stored in a database that is accessed by the Wizard programme. Figure 6 shows three of the user models. The columns show the weightings associated with continuous variables and particular likes/dislikes for categorical variables. For all of the users, food quality is important, being the highest or second highest ranked attribute for users overall. Cost is also relatively important for each of these users, with both decor and service being of lesser importance. Overall in the 18 user models, food quality and cost were generally among the top three ranked attributes, while the ranking of other attributes such as decor, service, neighbourhood and food type varied widely.

2.2 Designing a Wizard Interface

The Wizard's task of generating dialogue responses in real time based on a specified set of strategies and a user model is extremely demanding. By carrying out pilot studies using pen and paper, and accessing restaurant information web sites, we identified several requirements for a Wizard Interaction Tool:

1 It had to allow the Wizard to specify user constraints on restaurant options in a flexible way (e.g., West Village, costing $50-$80);

2 It had to respond to Wizard queries in real time;

3 It had to provide a ranking of options consistent with a user model;

4 It had to support the use of many different user models;

5 It had to allow the Wizard to easily see, for different user models, the reason why a particular option was highly ranked;

Wiz:	So there are approximately 20 restaurants that satisfy the criteria that you specifi ed. The fi rst one is called Lespinasse and it's a French restaurant in midtown east which has excellent food decor and service and the cost is $86.
CK:	do you have any sample menus or anything?
Wiz:	I am sorry, I have no sample menus.
CK:	are they known for their wine list?
Wiz:	again I am sorry I haven't got that information.
CK:	ok can you tell me about another restaurant then?
Wiz:	I do have, I do have reviews.
CK:	for that Lespinasse restaurant.
Wiz:	yes.
CK:	can you read me the review or part of it?
Wiz:	The much-celebrated Gray Kunz has departed, leaving the more traditional Christian Louvrier at his place behind the stove.
Wiz:	that's it.
CK:	(laughs) ok. Tell me about another restaurant
Wiz:	ok the next restaurant is called La Grenouille, it's again a French restaurant. Again it's in midtown east, the food quality decor and service are all excellent the cost of this one is $80.
CK:	ok do you have menus or reviews for this one.
Wiz:	the review says: Gorgeous fbwers, fi ne service, rich people and a menu written entirely in French.
CK:	I think that you need better better reviews. They don't. ok the food was excellent in both those places. Can you tell me about can you tell me how far those are from the Shubert theatre?
Wiz:	That's 8 blocks.
CK:	8 blocks so that's great. Ok. Do you have anything that's tell me about the best restaurant that you have.
Wiz:	the best restaurant along what dimension?
CK:	erm
Wiz:	in terms of food quality, cost..
CK:	food quality
Wiz:	ok, erm the best quality restaurant is er Lespinasse
CK:	ok Lespinasse, that's the fi rst one that you told me about.
Wiz:	yes
CK:	ok, erm ,then I'm happy with that one.

Figure 5. Excerpt form the Wizard dialogue with user CK, default strategy, CK task from Figure 4.

6 It had to allow the user to modify the task constraints over the course of the dialogue.

It was obviously impossible to simply use the Zagat's web site to support a Wizard-of-Oz data collection since:(1) networking delays mean that it often did not respond in real time; (2) it did not support the use of different user models;

Usr	FQ	Serv	Dec.	Cost	Nbh.	FT	Nbh Likes	Nbh Dislikes	FT Likes	FT Dislikes
CK	0.41	0.10	0.03	0.16	0.06	0.24	Midtown, Chinatown, TriBeCa	Harlem, Bronx	Indian , Mexican, Chinese, Japanese, Seafood	Vegetarian, Vietnamese, Korean, Hungarian, German
HA	0.41	0.10	0.03	0.16	0.06	0.24	Upper W. Side, Chelsea, Chinatown, E. Village, TriBeCa	Bronx, Uptown, Harlem, Upper E. Side, Lower Manhattan	Indian, Mexican, Chinese, Japanese, Thai	no-dislike
OR	0.24	0.06	0.16	0.41	0.10	0.03	W. Village, Chelsea, Chinatown, TriBeCa, E. Village	Upper E. Side, Upper W. Side, Uptown, Bronx, Lower Manhattan	French, Japanese, Portugese, Thai, Middle Eastern	no-dislike
SD	0.41	0.10	0.03	0.16	0.06	0.24	Chelsea, E. Village, TriBeCa	Harlem, Bronx	Seafood, Belgian, Japanese	Pizza, Vietnamese

Figure 6. Sample user models: FQ = Food Quality, Serv = Service, Nbh = Neighbourhood; FT = Food Type.

and (3) new constraints added while executing the task entailed respecifying the whole query from the top level page.

We therefore built a Wizard Interaction Tool (WIT) to aid in data collection. WIT was built using dynamic HTML and the display was driven using XSLT transformations of the underlying database of New York restaurant information. This allowed the web browser display to be extremely fast. The database was populated with the database of New York restaurants used in MATCH and augmented with additional information that was downloadable from the Zagat's web site. WIT allows the Wizard to specify a set of restaurant selection criteria, and returns a list of options that match the users' request. The tool also supports the selection of any user model, so that the Wizard can identify restaurant options and attributes that are important to the particular user, as explained in more detail below. The Wizard used this information, along with a written schema of dialogue strategies to guide his interaction with the user in each dialogue.

Figure 7 illustrates the Wizard interaction tool (WIT). The main function of the interface is to provide relevant information to allow the Wizard to quickly identify sets of restaurants satisfying the user's query, along with reasons for choosing them, while respecting the particular preferences of that specific user. The tool contains three main panels. The right hand panel supports query specification, allowing the Wizard to specify constraints corresponding to the user's query. as a Boolean combination of constraints and database attribute values. In this example the specified query is *Japanese, Korean, Malaysian or Thai restaurants, costing between 30-40 dollars, with food quality greater than 20 and service and decor greater than 15 anywhere in Manhattan*. The right hand panel contains radio buttons allowing the Wizard to specify: cost range (using one button for upper, and one for lower limits), food quality, service, decor, cuisine and neighbourhood. Note that neighbourhood is not depicted as the user has scrolled to the top of the relevant panel. Omitting a selection (e.g., neighbourhood) means that this attribute is unconstrained, corresponding in this case to the statement *anywhere in Manhattan*.

The left hand panel shows the specific restaurants satisfying the query along with information for each restaurant including its overall utility calculated using the relevant user model, and the absolute values for food quality, service, decor and cost. For each attribute we also supply corresponding weighted attribute values, shown in brackets after each absolute attribute value. The overall utility and weighted attributes are all specific to a given user model. In this example, for the restaurant Garden Cafe (the first option in the left hand panel), the overall utility is 80, absolute food quality is 25 (weighted value 35), service is 25 (weighted value 8), decor is 15 (weighted value 2) and cost is 38 dollars (weighted value 10). So, according to the user model, the main reason why CK should like Garden Cafe is that the food quality is excellent, as indicated by the fact that this attribute contributes almost half of the overall weighted utility (35 out of 80 units).

The centre panel of WIT provides specific information about the restaurant selected in the left hand panel, including its address, neighbourhood, a review and telephone number.

Overall the tool provides a method for the Wizard to quickly identify candidate restaurants satisfying a particular user's preferences, along with reasons (the weighted attribute values) why the user should choose that restaurant. The UI also allows the Wizard to see at a glance the trade-offs between the different restaurants, by comparing the different weighted utilities. For example, the main reason for preferring the Garden Cafe over Taka (second in the list, highlighted) is that it has better service and decor (as shown by the different weighted values of these attributes).

We demonstrate the effects of the user model by showing the results for the same query for the OR user model from Figure 6. The different user model for

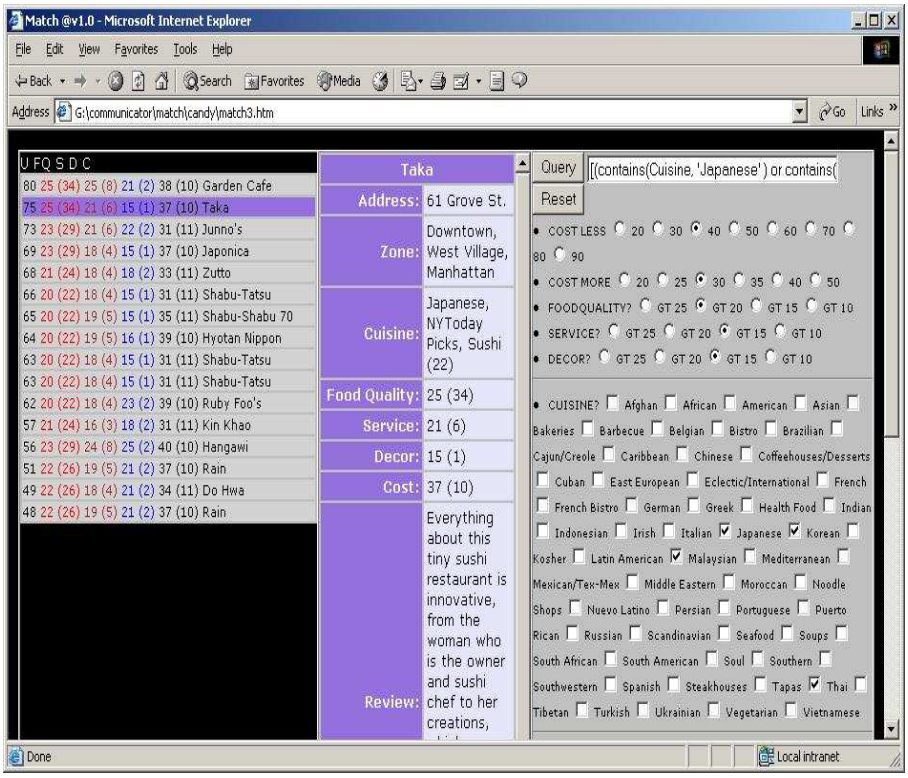

Figure 7. Wizard interface for user CK after Wizard enters query for: "Japanese, Korean, Malaysian or Thai restaurants, costing between 30-40 dollars, with food quality greater than 20 and service and decor greater than 15 anywhere in Manhattan."

OR leads all weighted utilities to change, causing a change in the ordering of the overall set of options. In Figure 7, the highest ranked restaurant was Garden Cafe, mainly because of its good food quality (the attribute most highly valued by user CK). In contrast in Figure 8, the highest ranked restaurant is Junnos because of its reasonable cost, cost being the most highly valued attribute for user OR.

It is also possible for the Wizard to easily override preferences expressed in the user model by specifying additional constraints on the query. For example, Figure 9 illustrates a dialogue with the Wizard with user HA. The user model for user HA is given in Figure 6. As the user model indicates, HA has expressed a preference for Japanese food (see column FT Likes). However during the dialogue in Figure 9, in turn HA10, HA overrides this preference in the current situation. The Wizard implements this override by clicking the

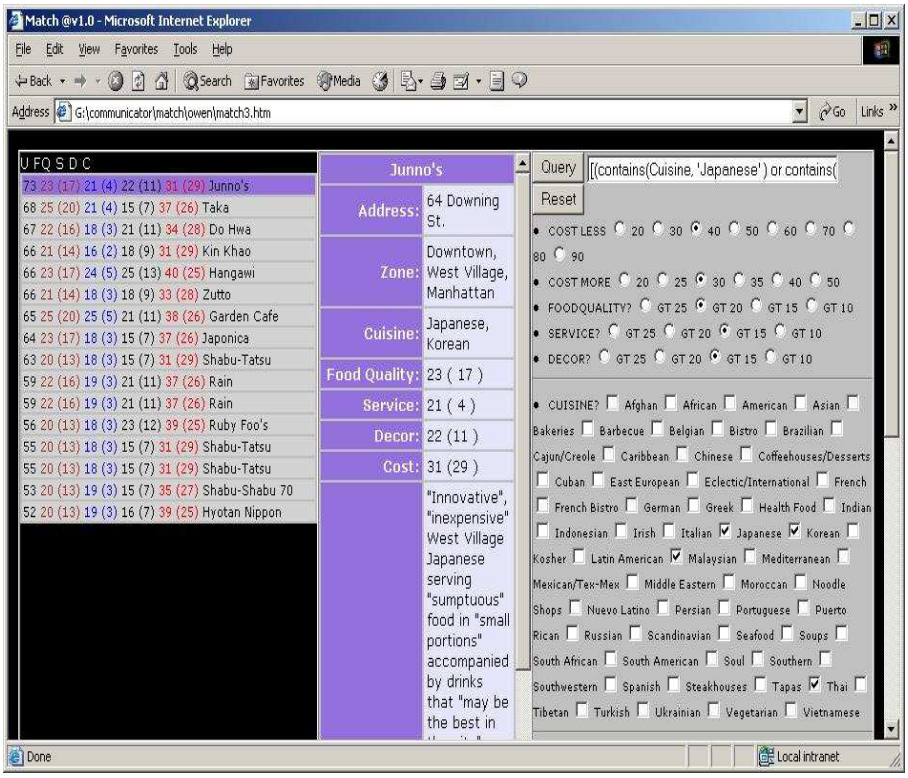

Figure 8. Wizard interface for user OR after Wizard enters query for: "Japanese, Korean, Malaysian or Thai restaurants, costing between 30-40 dollars, with food quality greater than 20 and service and decor greater than 15 anywhere in Manhattan".

"not Japanese" button in the query specification window. Similarly, when HA specifies in turn HA18, that he is also not interested in Italian food, the Wizard can simply add to the set of constraints on the query by clicking "not Italian" (selected by scrolling down the screen).

2.3 Wizard Dialogue Strategy

In addition to identifying tasks and user models, we were also interested in testing specific hypotheses about information presentation strategies. Our pilot studies with pen and paper indicated that a major problem for users was to remember and compare complex sets of options and their attributes. To reduce this complexity we therefore devised a combined SUMMARY/RECOMMEND

Wiz1:	I've actually got, erm, a large number of restaurants, again this time about, 50 I'd say, erm
HA2:	do you know the location of the Lucille Lortel theatre? I should know I've just been at a play there.
Wiz3:	its on, erm, its between 6 and 7th avenues on, is it Nicholas? Christopher.
HA4:	oh its, I see yes. Right yes, I know where it is.
Wiz5:	erm, so its west village, right, west of west village.
HA6:	right
Wiz7:	so I have erm, the top three restaurants are erm again very similar in decor, they have different food quality service and cost. And the first one I have is er a Japanese in the west village, erm, very good food and service, erm, the cost is $37.
HA8:	ok, since I just had Japanese.
Wiz9:	you don't want Japanese?
HA10:	I think I'll want something different today.
Wiz11:	ok. erm, so I've now got about 40 erm, the top three on this occasion, erm, all have very similar decor, but they have different food quality, service and cost, erm, the top one is erm, Mexicana Mama, erm, which is west village Mexican/Tex/mex which has er very good food, er poor decor , sorry very good service, and erm the cost is $26. The next one is the Pearl Oyster Bar which is again west village, it's a seafood restaurant excellent quality food, erm, decent service, cost is $34.
HA12:	can you read the review for that - does it say anything about how fresh the seafood is?
Wiz13:	its just a marble counter with a few small tables, but Pearl has won over its neighbourhood with its casual charm, and Maine inspired seafood.
HA14:	hmm that's Maine as in the state?
Wiz15:	yes. erm the next one is erm, called the Blue Ribbon, erm this is again west village, it's a new American west with er very good food and service, the cost is $45 a head. Do you want to hear some more?
HA16:	er yes
Wiz17:	erm the next three I've got er again have very similar decor, they differ in terms of food quality service and cost. Er the first one is Il Mulino, which is, erm west village Italian, it has excellent food, erm, very good service, erm, the cost is $65 a head.
HA18:	ok, er you can probably skip Italian as well as Japanese.
Wiz19:	ok.
HA20:	I've been eating too much Italian food. And er do you have a couple more?
Wiz21:	yes, erm, there's Natali West, which is, erm, which is west village Indian, which has, erm, decent food, erm and service, the cost is $27. do you want to hear more?
HA22:	yeah
Wiz23:	erm, the the next three I have erm again have very similar er decor, they differ in terms of food quality service and cost. Erm, the first one is Little Basil which is a Thai restaurant in the West Village which has, erm, decent food and service, the cost is $27 a head.
HA24:	and can you read the review for that?
Wiz25:	this is not typical Thai home cooking yet the food remains true to the essence of Thai cuisine with salty sour hot and sweet flavours. do you want to
HA26:	well, ok, I actually I think that I like the erm, the Oyster,
Wiz27:	ok, the Pearl Oyster
HA28:	Pearl Oyster bar
Wiz29:	the Pearl Oyster bar
HA30:	so I think, I think, that I'll go for trying to reserve that.

Figure 9. Wizard dialogue with user HA, tailored strategy, for the West Village task.

strategy, which we compared with a SERIAL presentation strategy similar to that used by current speech dialogue systems (Levin et al., 2000).

The SUMMARY provides an overview of the range of overall utility of the option set, along with the dimensions along which that set differ with respect to their attribute values. The aim is to inform users about both the set of choices, along with the range of reasons for making those choices. After entering a query corresponding to the user's choice in WIT, the Wizard examines the user selected set of restaurants and determines which attributes have the *same* values and which attributes have *different* values. Then he states for the chosen

restaurants which attributes are similar and which are different. The RECOMMENDATION then describes the first restaurant including all attributes that have not been mentioned so far. The tailored strategies are applied with the relevant user model. The RECOMMENDATION strategy that the Wizard uses in the dialogues is motivated by user tailored strategies for the real estate domain described in (Carenini and Moore, 2001).

The Zagat attributes are on a scale of 1-30. The Wizard's dialogue strategy lexicalises the absolute values, as follows, in order to increase comprehensibility: 26-30 excellent; 21-25 very good; 16-20 decent; 11-15 poor. We did not lexicalise price instead using absolute value, as there was little agreement about how to describe cost among pilot subjects. Two restaurants were judged to have the same value for a given attribute if the attribute had the same lexicalisation. We did not make similarity judgements about price.

Here is an example of a SUMMARY/RECOMMEND STRATEGY: *There are 20 restaurants that satisfy your criteria. The first three have decent decor, but differ in food quality, service and cost. The first one I have is the Garden Cafe, which is in midtown east. It's Japanese, it has very good food and service and the cost is 38 dollars.*

Note that although this strategy states the total number of restaurants satisfying a query, we provide details about just three restaurants to avoid overloading the user. If there are fewer than three that satisfy the query then we obviously just provide information about these.

We contrasted this with the SERIAL strategy applied with a default user model derived by combining the average weights for the 18 user models we collected. The SERIAL strategy specified the number of restaurants satisfying the query, and then stated the attributes in sequence, stating positive before negative values and aggregating across these where possible: *There are 18 restaurants that satisfy your criteria, the first one is Nyona, which is in Chinatown, it's southeast Asian, the food quality is very good, although the decor and service are poor. The cost is 21 dollars.*

2.4 Collecting Sample Dialogues

Six subjects participated in the Wizard dialogue collection experiment resulting in a corpus of 24 dialogues. All of the subjects were familiar with Manhattan restaurants. We first examined the 30 typical tasks generated by our users. By identifying the common characteristics of these user-generated tasks, we generated two further control tasks for the domain. Each user participated in four tasks, two that they had generated themselves and two control tasks. We used this combination of user-generated and control tasks to combine ecological validity while controlling for task variability. User-generated tasks have the advantage of being both real and motivating, i.e., they are prob-

lems that the user genuinely wants to solve. At the same time, however there was a great deal of variability in the complexity and number of solutions to these user-generated tasks, and we wanted to be able to reduce this using the control tasks.

The underlying model and Wizard strategy were also varied; each user carried out two tasks with their own user model, and the tailored SUMMARY/RECOMMENDATION dialogue strategies. Each user also carried out two tasks with the default user model and the SERIAL strategy. Model/strategy and task provenance were crossed so that each user overall received four tasks: self-task/own model/tailored strategy, self task/default model/serial strategy, control-task/own model/tailored strategy, control-task/default model/serial strategy. Users carried out the four tasks in two separate sessions. Task order was randomised but each session included one user-generated and one control-task, one own model/tailored strategy and one default model/serial strategy.

A sample dialogue illustrating a control task of *Find a restaurant in the West Village* using a strategy tailored to user HA is shown in Figure 9. A dialogue illustrating the CK task in Figure 4 with the default user model and the SERIAL dialogue strategy is in Figure 5. This dialogue illustrates issues concerning what information the Wizard has available and the user's understanding of the system's capabilities. The user is trying to find a French restaurant for her friends who are food snobs. She would like to hear about the menu and wine list but this information is not available. The Wizard offers that he does have reviews, but a little later, she says that better reviews are needed. The dialogue also illustrates how the Wizard needed access to distance information. The real MATCH system can do such calculations, but this was not implemented in WIT. The Wizard kept a map of New York City next to him during the dialogue interactions, and tried to quickly make such calculations.

For each dialogue we also collected both quantitative and qualitative data. We collected quantitative information about the number of turns, words and duration of each dialogue. After each dialogue was completed, the users were asked to complete a survey. The survey first requested the users to give permission for their dialogues to become part of a public corpus so they can be distributed as part of the ISLE project. Then they were required to state their degree of agreement on a 5 point Likert-scale with three specific statements designed to probe their perception of their interaction with the Wizard (Mr. Croque): (1) I feel confident that I selected a good restaurant in this conversation; (2) Mr. Croque made it easy to find a restaurant that I wanted to go to; and (3) I'd like to call Mr. Croque regularly for restaurant information.

2.5 Results of the Wizard Study

The Wizard study allowed us to collect 30 representative tasks and 24 dialogues, along with user models for 18 users for a complex information seeking domain. We also devised a useful tool for supporting Wizard-of-Oz style data collection, that embodies user's specific preferences. This should support the collection of further data in this, and with suitable modifications, other domains.

Users' qualitative comments were useful for the further development of both the MATCH system and the strategies implemented in it. Our main experimental manipulation was not completely successful: several of our predicted effects were not supported by our data because our combined SUMMARY/RECOMMEND strategy did not improve efficiency of information access more than our control strategy. This may have been because we ran too few subjects. Nevertheless we did observe some interesting findings that confirmed our predictions: Users were more proactive with the Own Model/Tailored combination in actively supplying task constraints, and we also found correlations between task length and perceived task complexity, as well as between verbosity and likelihood of future use. Overall these findings provide some evidence for the benefits of strategies that are tailored to user requirements and that reduce the overall length of the dialogue. We next set out to define and test such strategies.

One problem with the Wizard-of-Oz method is that the cognitive demands on the Wizard mean that it is hard to test multiple strategies simultaneously. Furthermore, the fact that strategies occur in different dialogue contexts for different users makes it hard to draw definitive conclusions. We attempted to address these problems in our next study, using the Overhearer method.

3. Overhearer Experiment

The idea behind the *Overhearer* method is that the subject is an "overhearer" of a series of exchanges from several dialogues, that have previously been logged as successful interactions with MATCH. The experimental subject is asked, for each exchange, to provide feedback assessing some aspect of the quality of the system's output. Even though the Overhearer method requires our information presentation strategies to be fully implemented and integrated into the system, the use of the method: (1) allows us to get feedback during the course of the dialogue, rather than only at the end of the dialogue; (2) allows us to finesse problems with speech recognition and understanding, which may not be robust enough to support dialogue interaction with real users at the point at which we are trying to refine dialogue management and response strategies; and (3) allows us to ask the user to directly compare and contrast the use of two (or more) alternative dialogue strategies in the same dialogue

context. The remainder of this section first describes the experimental setup and then summarises our results.

3.1 Experimental Method

Each dialogue involves one restaurant-selection task, but requires several exchanges to complete. Users' judgements are elicited using a series of web-pages. Each web page sets up the task by showing the MATCH system's graphical response for an initial user query, e.g., *Show Italian restaurants in the West Village*. Then the page shows the user circling some subset of the restaurants and asking the system to *summarise*, *compare* or *recommend* options from the circled subset. Figure 10 shows an example of an initial web page with the systems graphical response, followed by a screen dump showing the user circling a set in order to ask for a comparison. Each user completed 4 tasks with the system.

The subject sees one page each for SUMMARY and RECOMMEND, and two for COMPARE, for each task. On each page, the subject sees one system response tailored to her user model, and a different one tailored to the user model of another randomly selected subject. The order of the four tasks, and the order of appearance of strategies within the task is consistent across subjects. However, the order of presentation of subject-tailored and other-tailored responses is randomised from page to page.

For each instance of a RECOMMEND, SUMMARY, or COMPARE, the subject is asked to state her degree of agreement (on a 5-point Likert-scale) with the following statement, intended to determine the *informativeness*, or *information quality*, of the response: *"The system's utterance is easy to understand and it provides exactly the information I am interested in when choosing a restaurant."*.

This entire sequence of web pages is presented twice, as we wanted to compare spoken and textual presentation of information. We wanted to test the hypothesis that outputs tailored by User Models would be especially helpful for speech because the additional memory load of remembering complex information. In this study we did not test the utility of multimodal outputs. The first time the subject can only read (not hear) the system responses. The second time, she can only hear them. We used this read-then-hear approach after careful piloting because we wanted to make comparisons between text and speech that were not biased by the performance of TTS. By presenting subjects with already familiar material, we hoped that their judgements would not be prejudiced by an inability to understand poor TTS.

To summarise, each subject "overhears" a sequence of four dialogues about different restaurant-selection tasks. The entire sequence is presented twice (once for text, once for speech). The subject makes eight information qual-

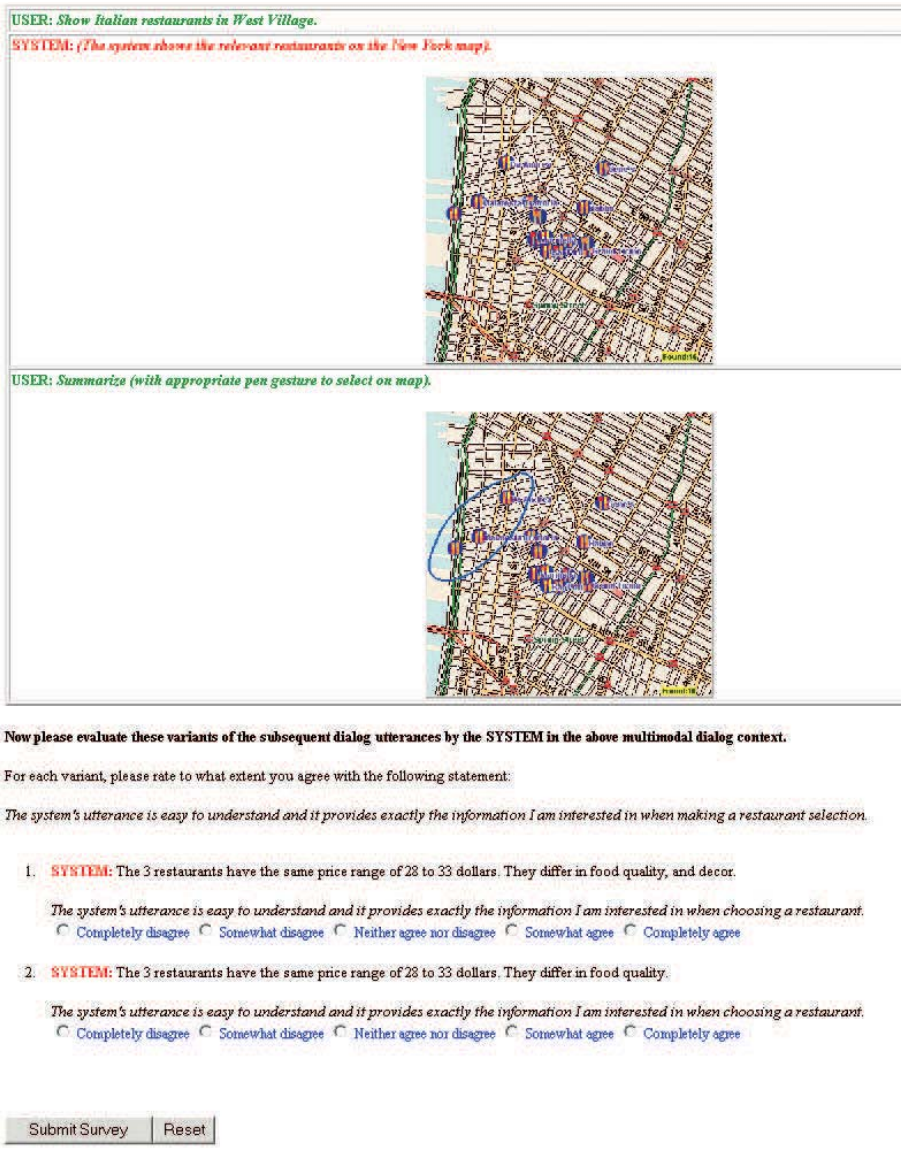

Figure 10. Sample web page for Overhearer experiment.

ity judgements for each dialogue each time. The total number of information quality judgements per subject is sixty-four. The total time required to complete the experiment is approximately half an hour per subject.

3.2 Overhearer Experimental Results

Sixteen subjects completed the experiment. All were fluent English speakers. We also had them provide demographic information, about the frequency they ate out, and their familiarity with Manhattan, as we felt that these might affect their judgements. Most eat out moderately often (seven eat out 3-5 times per month, six 6-10 times). All sixteen currently live in northern New Jersey. Eleven described themselves as somewhat or quite familiar with Manhattan, while five thought they were not very familiar with it. After the experiment, ten subjects identified themselves as very interested in using a system like MATCH in the future.

To analyse the results, we first tested whether the type of user model affected subjects' rankings of the information quality of the system's responses. A one-way Analysis of Variance (ANOVA) for information quality by strategy and model indicates no overall effect of model (F = 1.0, p = 0.30 n.s.).

However, the Random model condition includes cases where the randomly assigned model is close to the User's Own model. We therefore filtered the original set of judgements to exclude cases where the distance between the Random Model and the User's Own Model was less than 0.2. This removed 9% of judgements from the original data set. To test the hypotheses, we conducted an analysis of variance with Model Type (Own,Random) * Mode (Speech,Text) * Strategy (Recommend, Compare2, Compare3,Summary) as independent variables and Judgements of Response Quality as dependent variable.

As predicted, there were main effects for Model Type (F=6.00, df = 1,906, $p < 0.02$) showing that using the User's Own Model significantly improved judgements. Again as predicted, Mode was significant, (F=7.57, df = 1,906, $p < 0.01$), with Text responses being rated more highly than Speech. There were also differences between the different Strategies (F=117.59, df = 3,906, $p < 0.0001$), with post hoc tests showing the Summary Strategy being judged as much worse than others. Finally, and contrary to our predictions, there was no interaction between Model Type and Mode (F=0.06, df = 1,906, $p > 0.05$). We expected that the additional difficulty of remembering complex spoken information would lead users to especially prefer responses generated using their Own Model in the Speech condition, given that these are explicitly tailored to their needs. The absence of the interaction term means that this prediction was not confirmed.

Thus, we were able to use the Overhearer method to test the efficacy of user tailored dialogue strategies based on the decision-theoretic user models, even though the complete system was not robust enough to support a data collection with real users interacting live with the system. Our main prediction was confirmed, namely that the user models are effective at addressing the problem of information presentation for complex problem-solving tasks.

4. Discussion

This chapter makes both technical and methodological contributions. At the technical level, we have addressed a critical problem for multimodal systems, namely presentation of information about multiple options each with complex attributes, in a way that enables users to make informed comparisons between these options. We summarise our research on information presentation techniques based on user models that are motivated by multi-attribute decision theory (Stent et al., 2002; Walker et al., 2002; Walker et al., 2004; Whittaker et al., 2002). These address the information presentation problem by allowing us to identify options that are relevant to the specific user, as well as the attributes of those options that are most important to that particular user. These are promising techniques which may generalise to speech-only interfaces, where again there are problems of complex information presentation.

Our methodological contribution is to describe how different methods can be used in iterative development of multimodal systems. We have shown how a combination of Wizard-of-Oz and Overhearer techniques can be used to derive system requirements and develop information presentation algorithms without needing to develop complete working systems at the outset. The Wizard-of-Oz technique is more open-ended: allowing us to determine user tasks and strategies as well as to pilot various expert presentation strategies. Although this data can be gathered using tools that are fairly straightforward to develop, there are nevertheless important design constraints for such tools, in that they need to support the Wizard in producing complex but plausible system behaviours in real-time. While the technique allows rich data to be collected, one limitation of this data is that it does not carefully control the dialogue context in which various strategies are generated. In contrast the Overhearer technique allows more control: once we have clear ideas about potential presentation strategies we can systematically compare and evaluate these in situations where we can control the specific dialogue context. However one weakness of the Overhearer technique is that it measures *perception* rather than observed behaviour and in the long term Overhearer data should be supplemented with this.

In conclusion, we have presented and evaluated novel algorithms for information presentation in multimodal systems, and described methods by which such algorithms can be iteratively developed. Future work will explore other information presentation techniques and evaluate other potential uses of information derived from these types of user model.

Acknowledgements

The work reported in this chapter was partially funded by DARPA contract MDA972-99-3-0003 and by the National Science Foundation under Grant No. 9910603 (ISLE) to the University of Pennsylvania. Thanks also to our col-

leagues on the MATCH project, Johanna Moore, Michael Johnston, Patrick Ehlen, Guna Vasireddy, Srini Bangalore, Preetam Maloor and Amanda Stent.

References

André, E. (2002). Natural language in multimedia/multimodal systems. In Mitkov, R., editor, *Handbook of Computational Linguistics*, pages 715–734. Oxford University Press.

Bangalore, S. and Johnston, M. (2000). Tight coupling of multimodal language processing with speech recognition. In *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, pages 126–129, Beijing, China.

Beutnagel, M., Conkie, A., Schroeter, J., Stylianou, Y., and Syrdal, A. (1999). The AT&T next-generation text-to-speech system. In *Proceedings of Meeting of ASA/EAA/DAGA*, pages 20–24, Berlin, Germany.

Carenini, G. and Moore, J. D. (2000). An empirical study of the influence of argument conciseness on argument effectiveness. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 150–157, Hong Kong, China.

Carenini, G. and Moore, J. D. (2001). An empirical study of the influence of user tailoring on evaluative argument effectiveness. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1307–1314, Seattle, Washington, USA.

Edwards, W. and Barron, F. H. (1994). SMART and SMARTER: Improved simple methods for multiattribute utility measurement. *Organizational Behavior and Human Decision Processes*, 60:306–325.

Johnston, M. and Bangalore, S. (2000). Finite-state multimodal parsing and understanding. In *Proceedings of International Conference on Computational Linguistics*, pages 1200–1208, Saarbrücken, Germany.

Johnston, M. and Bangalore, S. (2001). Finite-state methods for multimodal parsing and integration. In *Proceedings of ESSLLI Workshop on Finite-state Methods, European Summer School in Logic, Language and Information*, pages 74–80, Helsinki, Finland.

Johnston, M., Bangalore, S., Vasireddy, G., Stent, A., Ehlen, P., Walker, M., Whittaker, S., and Maloor, P. (2002). MATCH: An architecture for multimodal dialogue systems. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 376–383, Philadelphia, Pennsylvania, USA.

Keeney, R. and Raiffa, H. (1976). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. John Wiley and Sons, Chichester, United Kingdom.

Levin, E., Narayanan, S., Pieraccini, R., Biatov, K., Bocchieri, E., Fabbrizio, G. D., Eckert, W., Lee, S., Pokrovsky, A., Rahim, M., Ruscitti, P., and Walker, M. (2000). The AT&T DARPA Communicator mixed-initiative spoken dialog system. In *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, pages 122–125, Beijing, China.

Oviatt, S. (1999). Ten myths of multimodal interaction. *Communications of the ACM*, 42(11):74–81.

Sharp, R., Bocchieri, E., Castillo, C., Parthasarathy, S., Rath, C., Riley, M., and Rowland, J. (1997). The Watson speech recognition engine. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4065–4068, Munich, Germany.

Stent, A., Walker, M., Whittaker, S., and Maloor, P. (2002). User-tailored generation for spoken dialogue: An experiment. In *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, pages 1281–1284, Denver, Colorado, USA.

Walker, M., Whittaker, S., Stent, A., Maloor, P., Moore, J., Johnston, M., and Vasireddy, G. (2002). Speech-plans: Generating evaluative responses in spoken dialogue. In *Proceedings of International Conference on Natural Language Generation (INLG)*, pages 73–80, New York, New York, USA.

Walker, M., Whittaker, S., Stent, A., Maloor, P., Moore, J., Johnston, M., and Vasireddy, G. (2004). Generation and evaluation of user tailored responses in dialogue. *Cognitive Science*, In press.

Whittaker, S., Walker, M., and Moore, J. (2002). Fish or fowl: A Wizard of Oz evaluation of dialogue strategies in the restaurant domain. In *Proceedings of International Conference on Language Resources and Evaluation (LREC)*, pages 1074–1078, Las Palmas, Gran Canaria, Spain.