Griol, D., Carbó, J. & Molina, J. M. (2013). An automatic dialog simulation technique to develop and evaluate interactive conversational agents. *Applied Artificial Intelligence*, 27(9), 759-780.
DOI: http://dx.doi.org/10.1080/08839514.2013.835230

An automatic dialog simulation technique to develop and evaluate interactive conversational agents

David Griol[1], Javier Carbó, and José M. Molina
Group of Applied Artificial Intelligence (GIAA), Computer Science Department, Carlos III University of Madrid, Leganés, Spain

During recent years, conversational agents have become a solution to provide straightforward and more natural ways of retrieving information in the digital domain. In this article, we present an agent based dialog simulation technique for learning new dialog strategies and evaluating con versational agents. Using this technique, the effort necessary to acquire data required to train the dialog model and then explore new dialog strategies is considerably reduced. A set of measures has also been defined to evaluate the dialog strategy that is automatically learned and to compare different dialog corpora. We have applied this technique to explore the space of possible dialog stra tegies and evaluate the dialogs acquired for a conversational agent that collects monitored data from patients suffering from diabetes. The results of the comparison of these measures for an initial corpus and a corpus acquired using the dialog simulation technique show that the conversational agent reduces the time needed to complete the dialogs and improve their quality, thereby allowing the conversational agent to tackle new situations and generate new coherent answers for the situations already present in an initial model.

INTRODUCTION

As we move toward a world where all the information is in the digital domain, it becomes necessary to provide straightforward ways of retrieving it. To achieve this goal, it is necessary to provide an effective, easy, safe, and transparent interaction between the user and the system. Thus, it is important to identify which modality or combination of modalities would be optimal to present the information and to interact with the user. To do so, in recent years there has been an increasing interest in simulating

This work was supported in part by Projects MINECO TEC2012 37832 C02 01, CICYT TEC2011 28626 C02 02, CAM CONTEXTS (S2009/TIC 1485).

[1] Address correspondence to David Griol, Applied Artificial Intelligence Group, Computer Science Department, Universidad Carlos III de Madrid, Av. de la Universidad, 30 Edificio Sabatini, Office 2.1 B06, Leganés, Spain. E mail: dgriol@inf.uc3m.es

human-to-human communication, including the so-called conversational agents in multiagents systems (McTear, 2004; López-Cózar and Araki, 2005).

Conversational agents have became a strong alternative for providing computers with intelligent and natural communicative capabilities. A conversational agent is a software that accepts natural language as input and generates natural language as output, engaging in a conversation with the user. To successfully manage the interaction with the users, conversational agents usually carry out five main tasks: automatic speech recognition (ASR), natural language understanding (NLU), dialog management (DM), natural language generation (NLG), and text-to-speech synthesis (TTS).

Spoken interaction can be the only way to access information in some cases, for example, when the screen is too small to display information (e.g., hand-held devices) or when the eyes of the user are busy with other tasks (e.g., driving; Weng et al., 2006). It is also useful for remote control of devices and robots, especially in smart environments (Menezes et al., 2007). One of the most wide-spread applications is information retrieval. Some sample applications are tourist-and-travel information (Glass et al., 1995), weather forecast over the phone (Zue et al., 2000), speech-controlled telephone banking systems (Melin, Sandell, and Ihse 2001), conference help (Bohus et al., 2007), and so forth. They have also been used for education and training, particularly in improving phonetic and linguistic skills; assistance and guidance to F18 aircraft personnel during maintenance tasks (Bohus and Rudnicky, 2005); and dialog applications for computer-aided speech therapy with different language pathologies (Vaquero et al., 2006). Finally, one of the most demanding applications for fully natural and understandable dialogs is embodied conversational agents and companions (Brahnam, 2009; Bailly, Raitt, and Elisei 2010).

The application of statistical approaches to the design of this kind of agents, especially regarding the dialog management process, has attracted increasing interest during the last decade (Young, 2002; Griol et al., 2008; Paek and Horvitz, 2000; Williams and Young, 2007). Statistical models can be trained from real dialogs, modeling the variability in user behaviors. The final objective is to develop conversational agents that have a more robust behavior and are easier to adapt to different user profiles or tasks.

The success of these approaches depends on the quality of the data used to develop the dialog model. Considerable effort is necessary to acquire and label a corpus with the data necessary to train a good model. A technique that has currently attracted an increasing interest is based on the automatic generation of dialogs between the dialog manager and an additional module called the user simulator, which represents user interactions with the conversational agent (Schatzmann et al., 2006; Paek and Horvitz, 2000). This way, a very important application of the simulated

dialogs is to support the automatic learning of optimal dialog strategies (Schatzmann et al., 2006).

In this article, we present an agent-based dialog simulation technique to automatically generate the data required to learn a new dialog model for a conversational agent. We have applied our technique to explore dialog strategies for the DI@L-log conversational agent (Black et al., 2005), designed to collect monitored data from patients suffering from diabetes. In addition, a set of specific measures has been defined to evaluate the main characteristics of the acquired data and the new dialog strategy that can be learned from them. The results of the comparison of these measures for an initial corpus and a corpus acquired using the dialog simulation technique show how the quality of the dialog model is improved once the simulated dialogs are incorporated.

The remainder of this article is organized as follows. Next section reviews different approaches related to the simulation of multiagent systems. This section focuses on the description of statistical techniques for user simulation and conversational agents. The third section of the paper describes the proposed agent-based dialog generation technique. The following section describes the measures defined for the evaluation of our proposal. The fifth section describes the practical application of our proposal for the DI@L-log conversational agent. "Evaluation Results" shows the results obtained for the different measures for an initial corpus and a corpus acquired using the proposed dialog simulation technique. Some conclusions and future work lines are described in the last section.

RELATED WORK

The study of multiagent systems (MAS) focuses on systems in which many intelligent agents interact with each other. Agents are considered to be autonomous entities characterized by a set of properties including social ability, reactivity, and proactiveness (Wooldridge and Jennings, 1995). The behavior of each of these agents determines the global operation and evolution of the system. The specification of agents' behaviors can be considered to be composed of two elements (Bandini, Manzoni, and Vizzari 2009), the representation of the set of agent actions and the environment in which they are situated.

The term computer simulation is defined in Bandini, Manzoni, and Vizzari (2009) as "the usage of a computational model to gain additional insight into a complex system's behavior (e.g., biological or social systems) by envisioning the implications of the modeling choices, but also to evaluate designs and plans without actually bringing them into existence in the real world."

Agent-based simulation (ABS) is a relatively recent modeling technique widely used to model these complex systems with applications in many disciplines ranging from logistics optimization (Weyns, Boucké, and Holvoet 2006), biological systems (Bandini et al., 2006), traffic conditions (Balmer and Nagel, 2006), pedestrian simulation (Ballinas-Hernández, Muñoz-Melendez, and Rangel-Huerta 2011), urban planning and simulation (Navarro, Flacher, and Corruble 2011), social sciences (Pavón et al., 2008), and economics (Windrum, Fagiolo, and Moneta 2007). Detailed studies can be found in Macal and North (2010); Bandini, Manzoni, and Vizzari (2009); Heath, Hill, and Ciarallo (2009).

The use of ABS models can be attributed to different causes, for instance, the system has still not fully completed, ethical reasons (e.g., the safety of humans would be involved), practical reasons (e.g., reduce the time and costs that are required to develop and evaluate the system), and so forth.

Despite this extreme heterogeneity of simulated realities and research areas, the different approaches usually share the common viewpoint on the modeled system based on the analytical unit as represented by an individual agent acting and interacting with other entities in a shared environment (i.e., the overall system dynamics are defined in terms of the result of individual agents' actions and interactions). This way, models of this kind are characterized by the presence of agents performing some kind of behavior in a shared environment. Thus, the main elements in the simulation model are agents with a possibly heterogeneous behavior, the environment that provides perceptions and enables their actions, and mechanisms of interaction among agents involving the exchange of information and the effects of the perceptions and corresponding actions decided on by the different agents.

Considering the growing interest of adapting agent-based approaches to modeling and simulation, the number of software frameworks specifically aimed at supporting the realization of agent-based simulation systems is not surprising. These kinds of frameworks often provide not only abstractions and mechanisms for the definition of agents and their environments, but also additional functionalities for the management of the simulation, its visualization, monitoring, and the acquisition of data about the simulated dynamics. A first category of these platforms provides general-purpose frameworks in which agents mainly represent passive abstractions interacting in an overall simulation process (e.g., NetLogo; Wilensky and Rand, 2012). A second category of platforms is based on general-purpose programming languages providing very similar support tools (e.g., Repast; North, Collier, and Vos 2006). A third category of platforms represents an attempt to provide a higher level linguistic support, trying to reduce the distance between agent-based models and their implementations (e.g., SimSesam; Klügl, Herrler, and Oechslein 2003).

User Modeling and Natural Language Processing

Research in techniques for user modeling has a long history within the fields of language processing and conversational agents. The main purpose of a simulated user in this field is to improve the usability of a conversational agent through the generation of corpora of interactions between the system and simulated users (Möller et al., 2006), reducing time and effort required for collecting large samples of interactions with real users. Moreover, each time changes are made to the system, it is necessary to collect more data in order to evaluate the changes. Thus, the availability of large corpora of simulated data should contribute positively to the development of the system.

Simulated data can be used to evaluate different aspects of a conversational agent, particularly at the earlier stages of development, or to determine the effects of changes to the system's functionalities (e.g., evaluate confirmation strategies or introduce errors or unpredicted answers in order to evaluate the capacity of the dialog manager to react to unexpected situations). A second usage is to support the automatic learning of optimal dialog strategies using statistical methodologies. Large amounts of data are required for a systematic exploration of the dialog state space, and corpora of simulated data are extremely valuable for this purpose.

Two main approaches can be distinguished in the creation of simulated users: rule based and data or corpus based. In a rule-based simulated user the investigator can create different rules that determine the behavior of the system (Chung, 2004; Lin and Lee, 2001; López-Cózar et al., 2003). This approach is particularly useful when the purpose of the investigation is to evaluate the effects of different dialog management strategies. In this way the investigator has complete control over the design of the evaluation study.

An alternative approach, often described as corpus based or data based, uses probabilistic methods to generate the user input, with the advantage that this uncertainty can better reflect the unexpected behaviors of users interacting with the system. Statistical models for modeling user behavior have been suggested as the solution to the lack of the data that is required for training and evaluating dialog strategies. Using this approach, the dialog manager can explore the space of possible dialog situations and learn new, potentially better strategies. Methodologies based on learning user intentions have the purpose of optimizing dialog strategies. A summary of user simulation techniques for reinforcement learning of the dialog strategy can be found in Schatzmann and colleagues (2006).

Studies done by Eckert, Levin, and Pieraccini (1997; 1998) introduced the use of statistical models to predict the next user action by means of an *n*-gram model. The proposed model has the advantage of being both statistical and task independent. Its weak point consists of approximating

the complete history of the dialog by a bigram model. In Levin, Pieraccinin and Eckert (2000), the bigram model is modified by considering only a set of possible user answers following a given system action (the Levin model). Both models have the drawback of considering that every user response depends only on the previous system turn. Therefore, the simulated user can change objectives continuously or repeat information previously provided.

Scheffler and Young (2001a; 2001b) propose a graph-based model. The arcs of the network symbolize actions, and each node represents user decisions (*choice points*). In-depth knowledge of the task and great manual effort are necessary for the specification of all possible dialog paths.

Pietquin and Dutoit combine characteristics of the Scheffler and Young model and Levin model. The main objective is to reduce the manual effort necessary for the construction of the networks (Pietquin and Dutoit 2005). A Bayesian network is suggested for user modeling. All model parameters are hand selected.

Georgila, Henderson, and Lemon propose the use of hidden Markov models (HMMs), to define a more detailed description of the states and consider an extended representation of the history of the dialog (Georgila, Henderson, and Lemon 2005). Dialog is described as a sequence of *Information States* (Bos et al., 2003). Two different methodologies are described for selecting the next user action, given a history of information states. The first method uses *n*-grams (Eckert, Levin, and Pieraccini 1997), but with values of n from 2 to 5 to consider a longer history of the dialog. The best results are obtained with 4-grams. The second methodology is based on the use of a linear combination of 290 characteristics in order to calculate the probability of every action for a specific state.

Cuayáhuitl and coauthors present a method for dialog simulation based on HMMs in which both user and system behaviors are simulated (Cuayáhuitl et al., 2005). Instead of training only a generic HMM model to simulate any type of dialog, the dialogs of an initial corpus are grouped according to the different objectives. A submodel is trained for each of the objectives, and a bigram model is used to predict the sequence of objectives.

A data-driven user intention simulation method that integrates diverse user discourse knowledge (cooperative, corrective, and self-directing) is presented in Jung and colleagues (2011). User intention modeling is based on logistic regression and Markov logic framework. Human dialog knowledge is designed into two layers, domain and discourse knowledge, and integrated with the data-driven model in generation time. A methodology of user simulation applied to the evaluation and refinement of stochastic dialog systems is presented in Torres, Sanchis, and Segarra (2008). The proposed user simulator incorporates several knowledge sources, combining statistical and heuristic information to enhance the dialog models by an automatic strategy learning.

In Schatzmann and coauthors (2007), a new technique for user simulation based on explicit representations of the user goal and the user agenda is presented. The user agenda is a structure that contains the pending user dialog acts that are needed to elicit the information specified in the goal. This model formalizes human–machine dialogs at a semantic level as a sequence of states and dialog acts. An Expectation Maximization (EM)-based algorithm is used to estimate optimal parameter values iteratively. In Schatzmann, Thomson, and Young (2007), the agenda-based simulator is used to train a statistical Partially Observable Markov Decision Processes (POMDPs)-based dialog manager. The main drawback of this approach is because of the large state space of practical spoken dialog systems, whose representation is intractable if represented directly. Although POMDPs outperform MDP-based dialog strategies, they are limited to small-scale problems, because the state space would be huge, and exact POMDP optimization is intractable (Williams and Young, 2007). As described in the following section, our proposed dialog simulation technique is based on iteratively building a statistical user and dialog model by modifying the probabilities associated to each user and system response each time a dialog is successfully simulated. A set of stop conditions is applied to automatically discover whether a simulated dialog has completed the predefined objectives.

OUR AGENT-BASED DIALOG SIMULATION TECHNIQUE

Our proposed architecture for providing context-aware services by means of conversational agents is described in Griol and coauthors (2010). It consists of five different types of agents that cooperate to provide an adapted service. *User Agents* are configured into mobile devices or PDAs. *Provider Agents* supply the different services in the system and are bound to *Conversational Agents* that provide the specific services. A *Facilitator Agent* links the different positions to the providers and services defined in the system. A *Positioning Agent* communicates with the Aruba positioning system to extract and transmit positioning information to other agents in the system (Sánchez-Pi et al., 2007). Finally, a *Log Analyzer Agent* generates user profiles that are used by Conversational Agents in order to adapt their behavior, taking into account the preferences detected in the users' previous dialogs.

The interaction with the different agents follows a process that consists of the following phases:

1. The Aruba positioning system is used to extract information about the positions of the different agents in the system. This way, it is possible to know the positions of the different User Agents and thus extract

information about the Conversational Agents that are available in the current location.
2. The Positioning Agent reads the information about position (coordinates *x* and *y*) and place (*Building* and *Floor*) provided by the Aruba Positioning Agent by reading it from a file, or by processing manually introduced data.
3. The Positioning Agent communicates the position and place information to the User Agent.
4. Once a User Agent is aware of its own location, it communicates this information to the Facilitator Agent in order to find out the different services available in that location.
5. The Facilitator Agent informs the User Agent about the services available in this position.
6. The User Agent selects the services in which it is interested.
7. Once the User Agent has selected a specific service, it communicates its decision to the Facilitator Agent and queries it about the service providers that are available.
8. The Facilitator Agent informs the User Agent about the identifier of the Conversational Agent that supplies the required service in the current location.
9. The User Agent asks the Conversational Agent for the required service.
10. Given that the different services are provided by context-aware Conversational Agents, they ask the User Agent about the context information that would be useful for the dialog. The User Agent is never forced to transmit its personal information and preferences. This is only a suggestion to customize the service provided by means of the Conversational Agent.
11. The User Agent provides the context information that has been required.
12. The conversational agent manages the dialog, providing an adapted service by means of the context information that it has received.
13. Once the interaction with the Conversational Agent has finished, the Conversational Agent reads the contents of the log file for the dialog and sends this information to the Log Analyzer Agent.
14. The Log Analyzer Agent stores this log file and generates a user profile to personalize future services. This profile is sent to the Conversational Agent.

In this article, we focus on the simulation of the User and Conversational Agents to acquire a dialog corpus. In our dialog generation technique, both agents use a random selection of one of the possible responses defined for the semantics of the task (expressed in terms of user and system dialog acts). At the beginning of the simulation, the set

of system responses is defined as equiprobable. When a successful dialog is simulated, the probabilities of the answers selected by the the conversational agent simulator during that dialog are incremented before beginning a new simulation.

One of the main problems that must be considered during the interaction with a Conversational Agent is the propagation of errors through the different modules in the system. The recognition module must deal with the effects of spontaneous speech and with noisy environments; consequently, the sentence provided by this module could incorporate some errors. The understanding module could also add its own errors (which are mainly a result of the lack of coverage of the semantic domain). Finally, the semantic representation provided to the dialog manager might also contain certain errors. Therefore, it is desirable to provide the dialog manager with information with regard to the parts of the user utterance that have been clearly recognized and understood and the parts that have not.

In our proposal, the user simulator provides the Conversational Agent with the semantic representation associated to the user input together with its confidence scores (García et al., 2003). To do this, an Error Simulation Agent has been implemented to include semantic errors in the generation of dialogs. This agent modifies the dialog acts provided by the user agent simulator once it has selected the information to be provided to the user. In addition, the error simulation module adds a confidence score to each concept and attribute in the semantic representation generated for each user turn.

For the study presented in this article, we have improved this agent using a model for introducing errors based on the method presented in Schatzmann and coauthors (2007). The generation of confidence scores is carried out separately from the model employed for error generation. This model is represented as a communication channel by means of a generative probabilistic model $P(c, a_u \mid \tilde{a}_u)$, where a_u is the true incoming user dialog act, $\tilde{a}_u$ is the recognized hypothesis, and c is the confidence score associated with this hypothesis.

The probability $P(\tilde{a}_u \mid a_u)$ is obtained by maximum-likelihood using the initial labeled corpus acquired with real users, and it considers the recognized sequence of words w_u and the actual sequence uttered by the user $\tilde{w}_u$. This probability is decomposed into a component that generates a word-level utterance from a given user dialog act, a model that simulates ASR confusions (learned from the reference transcriptions and the ASR outputs), and a component that models the semantic decoding process.

$$P(\tilde{a}_u|a_u) = \sum_{\tilde{w}_u} P(a_u|\tilde{w}_u) \sum_{w_u} P(\tilde{w}_u|w_u)P(w_u|a_u).$$

Confidence score generation is carried out by approximating $P(c \mid \tilde{a}_u, a_u)$, assuming that there are two distributions for c. These two

distributions are handcrafted, generating confidence scores for correct and incorrect hypotheses by sampling from the distributions found in the training data corresponding to our initial corpus.

$$P(c|a_w, \tilde{a}_u) = \begin{cases} P_{corr}(c) & if \quad \tilde{a}_u = a_u \\ P_{incorr}(c) & if \quad \tilde{a}_u \neq a_u \end{cases}.$$

The conversational agent simulator considers that the dialog is unsuccessful when one of the following conditions takes place:

- The dialog exceeds a maximum number of system turns slightly higher than the average number of turns of the dialogs acquired with real users.
- The answer selected by the dialog manager in the conversational agent simulator corresponds to a query not made by the user simulator.
- A query to the database generates an error because the user agent simulator has not provided the mandatory data needed to carry out the query.
- The answer generator generates an error when the answer selected by the conversational agent simulator involves the use of a data item not provided by the user agent simulator.

A user request for closing the dialog is selected once the conversational agent simulator has provided the information defined in its objective(s). The dialogs that fulfill this condition before the maximum number of turns are considered successful. Figure 1 shows the complete architecture for the proposed dialog simulation technique.

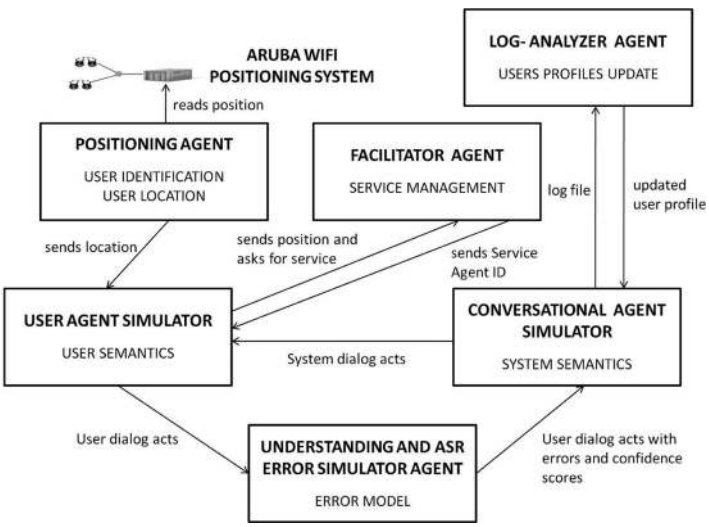

FIGURE 1 Graphical scheme of the proposed agent based dialog simulation technique.

MEASURES DEFINED FOR THE EVALUATION

For the evaluation of the quality of the dialogs provided by a conversational agent, we have defined a set of quantitative evaluation measures based on prior work in the dialog literature (Schatzmann, Georgilia, and Young 2005; Ai et al., 2007). This set of proposed measures can be divided into two types.

1. High-level dialog features: These features evaluate how long the dialogs last, how much information is transmitted in individual turns, and how active the dialog participants are.
2. Dialog style/cooperativeness measures: These measures analyze the frequency of different speech acts and study what proportion of actions is goal directed, what part is taken up by dialog formalities, and so on.

Six high-level dialog features have been defined for the evaluation of the dialogs: the average number of turns per dialog, the percentage of different dialogs without considering the attribute values, the number of repetitions of the most seen dialog, the number of turns of the most seen dialog, the number of turns of the shortest dialog, and the number of turns of the longest dialog. Using these measures, we tried to evaluate the success of the simulated dialogs as well as their efficiency and variability.

For dialog style features, we have defined and counted a set of system/user dialog acts. On the system side, we have measured the confirmation of concepts and attributes, questions to require information, and system answers generated after a database query. On the user side, we have measured the percentage of turns in which the user carries out a request to the system, provides information, confirms a concept or attribute, supplies Yes/No answers, and other answers not included in the previous categories.

Finally, we have evaluated the behavior of our system with real users considering the following measures for the evaluation:

1. Successful dialogs. This is the percentage of successfully completed tasks. In each dialog, the user has to obtain one or several items of information, and the dialog success depends on whether the system provides correct data or incorrect data to the user.
2. Average number of turns per dialog (nT).
3. Confirmation rate. It was computed as the ratio between the number of explicit confirmations turns (nCT) and the number of turns in the dialog (nCT/nT).
4. Average number of corrected errors per dialog (nCE). This is the average of errors detected and corrected by the dialog manager. We have considered only those errors that modify the values of the attributes and that could cause the failure of the dialog.

5. Average number of uncorrected errors per dialog (nNCE). This is the average number of errors not corrected by the dialog manager. Again, only errors that modify the values of the attributes are considered.
6. Error correction rate (%ECR). The percentage of corrected errors, computed as nCE/ (nCE + nNCE).

CASE APPLICATION: THE DI@L-LOG CONVERSATIONAL AGENT

DI@L-log is a spoken conversational agent that acts as a voice logbook to collect home-monitored data from patients suffering from Type-2 diabetes (Black et al., 2005). The data collected by the system are the patient's weight, blood pressure (systolic and diastolic values), and sugar levels. The system validates and analyzes the data, providing some immediate feedback to the patients regarding their current progress as well as communicating the results to doctors at the hospital who are able to review the patient's progress graphically and deal with any alerts generated by the system.

To apply our dialog simulation technique to this task, the definition of the semantics was carried out considering the information that is required to monitor the patients and inform them about their condition. Taking into account the requirements of the task, an initial strategy was designed for the DM. Figure 2 shows an example of a dialog acquired using this strategy.

As can be observed, three different phases are present in every dialog. First, there is an identification phase in which the system asks the user about his login and password and then waits until the user says that he is ready to provide the control data (S1 and S2 system turns). Second, the system analyzes which data are required for the current user, taking into account that the weight and sugar values are mandatory and the blood control is carried out only for specific patients (S3 to S10 system turns). In this phase, the system requires the user to provide this data. Every item is confirmed after the user has provided its value. The user can provide only one item at a time. In the last phase, the system consults the information that the patient has provided during the current dialog and compares it with the data that are present in a database that contains the values that were provided in previous dialogs. By means of this comparison, the system is able to inform the user about his/her condition and provide instructions that take this into account (S11 system turn).

A corpus of 100 dialogs was acquired using this strategy. In order to learn statistical models, the dialogs of the corpus were labeled in terms of dialog acts. In the case of user turns, the dialog acts correspond to the classical frame representation of the meaning of the utterance. For the DI@L-log task, we defined three task-independent concepts (*Affirmation*,

S1: Welcome to the DI@L-log System. Please say or enter your four-digit pin.

U1: 1356.

S2: Are you ready to give your measurements?

U2: Yes.

S3: Let's begin with your weight. How many stones?

U3: 12.

S4: I have understood 12 stones. Is it correct?

U4: Yes.

S5: What is your blood sugar?

U5: 11.

S6: I have understood 11. Is it correct?

U6: Yes.

S7: And finally, for your blood pressure you will be asked for the top number and then the lower number. What is the top number of your blood pressure?

U7: 160.

S8: I have understood 160. Is it correct?

U8: Yes.

S9: And what is the lower number?

U9: 86.

S10: I have understood 86. Is it correct?

U10: Yes.

S11: Your readings have been successfully updated. According to them, your weight is higher than last week, be very careful. Thanks for contacting DI@L-log. Goodbye.

FIGURE 2 An example of a dialog for the DI@L log task.

Negation, and *Not-Understood*) and four attributes (*Weight*, *Sugar*, *Systolic-Pressure*, and *Diastolic-Pressure*).

The labeling of the system turns is similar to the labeling defined for the user dialog acts. A total of twelve task-dependent concepts was defined, corresponding to the set of concepts used by the system to acquire each of the user variables (*Weight*, *Sugar*, *Systolic-Pressure*, and *Diastolic-Pressure*), concepts used to confirm the values provided by the user (*Confirmation-Weight*, *Confirmation-Sugar*, *Confirmation-Systolic*, and *Confirmation-Diastolic*), concepts used to inform the patient about his condition (*Inform*), and three task-independent concepts (*Not-Understood*, *Opening*, and *Closing*).

EVALUATION RESULTS

By employing the previously described agent-based architecture for dialog simulation, a set of 100,000 automatically labeled dialogs was acquired for the DI@-log task. Table 1 summarizes the statistics of the acquisition of this simulated corpus. A set of 11 different scenarios was defined to specify

TABLE 1 Statistics of the Corpus Acquisition for the DI Log System

Simulated dialogs	100,000
Successful dialogs	27,521
Different dialogs	1,573

the objectives of the simulation, taking into account whether the pressure values are necessary and different possibilities for the generation of errors and confidence measures. Given that the first and third phases of the dialog are mandatory and they always have the same structure, only the second phase in which the system collects the different values to monitor patients was taken into account for the simulation.

Figure 3 shows an example of a dialog from the acquired corpus. The objective defined for the dialog was to collect the weight, sugar, and pressure values. The values defined in the scenario are 12, 11, 160, and 80, respectively. Confidence scores generated by the error simulator agent are shown between brackets. A sentence in natural language, which is equivalent to the labeled system turn selected by the conversational agent simulator, has been incorporated to clarify the explanation.

In this dialog, the conversational agent simulator begins asking the user about his weight. Because a low confidence measure is introduced for the value provided by the user agent simulator in U1, the conversational agent simulator decides to confirm this value in S2. Then, this agent asks for the

————	
LOGIN PHASE	**U4:** (*Systolic*): 150 [0.1]
————	**S5:** (*Confirmation-Systolic*)
S1: (*Weight*)	I have understood 150. Is it correct?
Tell me your weight in stones?	**U5:** (*Rejection*) [0.9]
U1: *Weight*: 12 [0.1]	**S6:** (*Systolic*)
S2: (*Confirmation-Weight*)	Tell me your systolic pressure.
I have understood 12 stones. Is it correct?	**U6:** *Systolic*: 160 [0.9]
U2: (*Acceptance*) [0.9]	**S7:** (*Diastolic*)
S3: (*Sugar*)	And what is the lower number?
What is your blood sugar?	**U7:** *Diastolic*: 80 [0.9]
U3: *Sugar*: 11 [0.9]	————
S4: (*Systolic*)	*DATA ANALYSIS - FINAL ANSWER*
Tell me your blood systolic pressure.	————

FIGURE 3 A dialog extracted from the simulated corpus of the DI@L log task.

sugar value. The user agent simulator provides this value in U3 and a high confidence measure is assigned.

The conversational agent simulator asks for the systolic pressure in S4. An error is introduced in the value provided by the error simulator agent for this parameter (it changes 160 to 150) and a low confidence measure is assigned to this value. Then, the conversational agent simulator asks the user agent simulator to confirm this value. The user agent simulator rejects this value in U5 and the conversational agent simulator decides to ask for it again. Finally, the conversational agent simulator asks for the diastolic pressure. This value is correctly introduced by the user agent simulator and the error simulator agent also assigns a high confidence level. Then, the conversational agent simulator obtains the data required from the patient; next, the third phase of the dialog carries out the analysis of the condition of the patient and finally, it informs him.

High-Level Dialog Features

The first group of experiments covers the following statistical properties to evaluate the quality of the dialogs obtained using different dialog strategies: (1) dialog length, measured as the number of turns per task; number of turns of the shortest dialog; number of turns of the longest dialog; and number of turns of the most seen dialog; (2) different dialogs in each corpus, measured as the percentage of different dialogs (different labeling and/or order of dialog acts) and the number of repetitions of the most observed dialog; (3) turn length, measured as the number of actions per turn; (4) participant activity, measured as the ratio between system and user actions per dialog. Table 2 shows the comparison of the different high-level measures for the initial corpus and the corpus acquired incorporating the successfully simulated dialogs.

The first improvement that can be observed is the reduction in the number of turns. This reduction can also be observed in the number of turns of the longest, shortest, and most-seen dialogs. These results show

TABLE 2 Results of the High Level Dialog Features Defined for the Comparison of the Dialogs for the Initial and Final Strategy

	Initial strategy	Final strategy
Average number of turns per dialog	12.9 ± 2.3	7.4 ± 1.6
Percentage of different dialogs	62.9%	78.3%
Repetitions of the most seen dialog	18	3
User turns of the most seen dialog	9	7
User turns of the shortest dialog	7	5
User turns of the longest dialog	13	9

that improving the dialog strategy makes it possible to reduce the number of necessary system actions. The greater variability of the resulting dialogs can be observed in the higher percentage of different dialogs and less repetitions of the most-seen dialog obtained with the final dialog strategy.

We have observed that there is also a slight increment in the mean values of the turn length for the dialogs acquired with the final strategy. These dialogs are statistically longer; they showed 1.6 actions per user turn instead of the 1.3 actions observed in the initial dialogs. This is also a result of the better selection of the system actions. Regarding the dialog participant activity, Figure 4 shows the ratio of user-versus-system actions. Dialogs in the final corpus have a higher proportion of system actions because the systems needs to make a smaller number of confirmations.

Dialog Style and Cooperativeness

The experiments described in this section cover the following statistical properties: frequency of different user and system actions (dialog acts) and proportion of goal-directed actions (request and provide information) versus grounding actions (confirmations). We consider as well the remaining possible actions. The histograms in Figure 5 show the frequency of the most dominant user and system dialog acts in the initial and final strategy. In both cases, significant differences in the dialog acts distribution can be observed.

With regard to user actions, it can be observed that users need to employ fewer confirmation turns in the final strategy, which explains the higher proportion for the rest of user actions in this strategy. It also explains the lower proportion of Yes/No answers in the final strategy, which are mainly used to confirm that the system's service has been correctly provided. With regard to the system actions, it can be observed that there is a reduction in the number of system requests for data items. This explains a

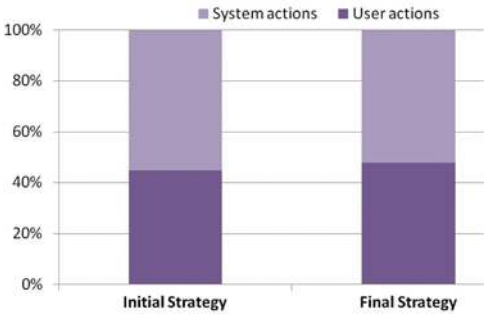

FIGURE 4 Ratio of user versus system actions (color figure available online).

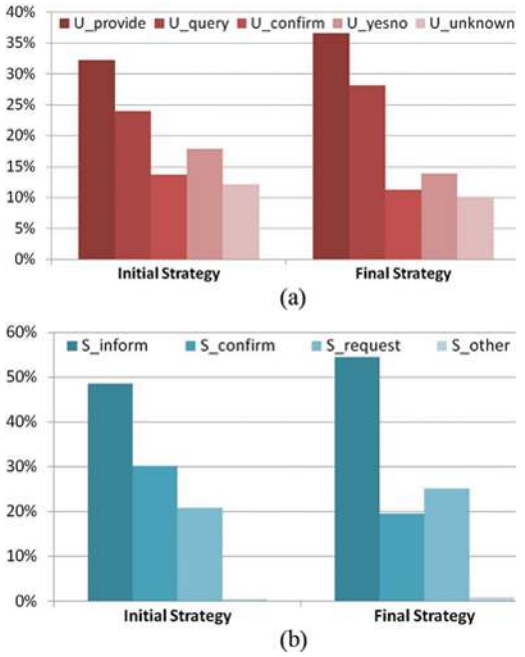

FIGURE 5 Histogram of (a) user dialog acts and (b) system dialog acts (color figure available online).

higher proportion of turns to inform and confirm data items in the dialogs of the final strategy.

Finally, we grouped user and system actions into categories in order to compare turns to request and provide information (goal-directed actions) versus turns to confirm data items and make other actions (grounding actions), as shown in Figure 6. This study also shows the better quality of the dialogs in the final strategy, given that the proportion of goal-directed actions is higher in these dialogs.

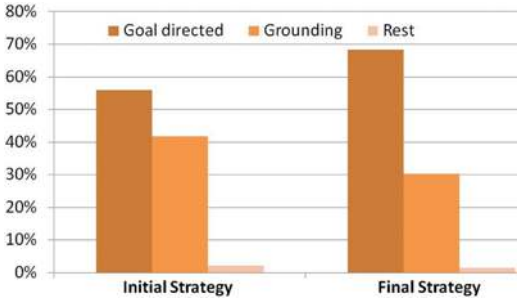

FIGURE 6 Proportions of dialog spent on goal directed actions, ground actions, and the rest of possible actions (color figure available online).

TABLE 3 Results of the Objective Evaluation of the Initial and Final Dialog Strategies with Real Users

	Successful dialogs	*n*T	Confirmation rate	ECR	*n*CE	*n*NCE
Initial strategy	91%	13.5	33%	82%	0.84	0.18
Final strategy	97%	9.3	25%	91%	0.88	0.09

Evaluation with Real Users

Finally, we evaluated the behavior of our system with real users using the same set of scenarios designed for the user simulation. A total of 150 dialogs were recorded from interactions of six users employing the initial and final dialog strategies. The evaluation was carried out by students and lecturers in our department. The results of the objective evaluation presented in Table 3 show that both systems could interact correctly with the users in most cases. However, the final system obtained a higher success rate, improving the results achieved with the initial strategy by 6% absolute. Using the final system, the average number of required turns is also reduced from 13.5 to 9.3. These values are slightly higher for both systems because in some dialogs the real users provided additional information, which was not mandatory for the corresponding scenario, or asked for additional information not included in the definition of the scenario once its objectives were achieved.

The confirmation and error correction rates were also improved by the final system, as the learned strategy makes possible to require less information from the user, reducing the probability of introducing ASR errors. The main problem detected was that when there was a user input misrecognized with a very high ASR confidence, this erroneous information was forwarded to the dialog manager. However, as the success rate shows, this fact did not have a considerable impact on the conversational agent operation.

CONCLUSIONS

In this article, we have described a technique for exploring dialog strategies in conversational agents. Our technique is based on an automatic dialog simulation technique to generate the data that is required to retrain a dialog model. Dialogs are automatically labeled during the simulation, using the semantics defined for the task. Successfully simulated dialogs are automatically detected by means of the definition of a set of stop conditions. The only requirements for applying our proposal are the definition of the semantics of the task and this set of stop conditions. Thus, the adaptation to a new task is simplified. In addition, conversational agents are

integrated into an architecture in which different agents cooperate to provide context-aware services by means of this kind of agents.

We have applied our proposal to the DI@L-log conversational agent, which acts as a voice logbook to collect home-monitored data from patients suffering from Type-2 diabetes. Different measures have been defined to evaluate high-level dialog features, dialog style and cooperativeness, and statistics of the acquired dialog corpora.

The results of the evaluation show that the proposed methodology can be used to automatically explore new enhanced dialog strategies. Carrying out these tasks with a nonautomatic approach would require a very high cost that sometimes is not affordable. By means of the simulated dialogs, the conversational agent reduces the time needed to completely fulfill the dialogs, thereby allowing the conversational agent to tackle new situations and generate new coherent answers for the situations already present in an initial model. This way, the conversational agent can ask for the required information using different orders, confirm these information items taking into account the confidence scores, reduce the number of system turns for the different kinds of dialogs, automatically detect different valid paths to achieve each of the required objectives, and so forth.

As a future work, we are adapting a previously developed statistical dialog management technique to learn a dialog manager for this task and evaluate the complete agent-based architecture with real users. We also want to evaluate the influence of taking into account different context information sources to improve the operation of an MAS developed with our proposed architecture. Finally, we want to apply our proposal to more difficult domains.

REFERENCES

Ai, H., A. Raux, D. Bohus, M. Eskenazi, and D. Litman. 2007. Comparing spoken dialog corpora collected with recruited subjects versus real users. In *Proceedings of the 8th SIGdial workshop on discourse and dialogue*, 124 131. Antwerp, Belgium.

Bailly, G., S. Raidt, and F. Elisei. 2010. Gaze, conversational agents and face to face communication. *Speech Communication* 52 (6): 598 612.

Ballinas Hernandez, A. L., A. Munoz Melendez, and A. Rangel Huerta. 2011. Multiagent system applied to the modeling and simulation of pedestrian traffic in counterflow. *Journal of Artificial Societies and Social Simulation* 14 (3): 2.

Balmer, M., and K. Nagel. 2006. Shape Morphing of Intersection Layouts Using Curb Side Oriented Driver Simulation. In *Innovations in design & decision support systems in architecture and urban planning*, Chapter 10, 167 183. Netherlands: Springer Verlag.

Bandini, S., F. Celada, S. Manzoni, R. Puzone, and G. Vizzari. 2006. Modelling the immune system with situated agents. In *Neuralriets*, Lecture Notes in Computer Science 3931: 231 243. Heidelberg: Springer.

Bandini, S., S. Manzoni, and G. Vizzari. 2009. Agent based modeling and simulation: An informatics perspective. *Journal of Artificial Societies and Social Simulation* 12 (4): 97 126.

Black, L., M. F. McTear, N. D. Black, R. Harper, and M. Lemon. 2005. Appraisal of a conversational arte fact and its utility in remote patient monitoring. In *Proceedings of the 18th IEEE symposium CBMS 05*, 506 508. Dublin, Ireland.

Bohus, D., S. Grau, D. Huggins Daines, V. Keri, G. Krishna, R. Kumar, A. Raux, and S. Tomko. 2007. Conquest An open source dialog system for conferences. In Proceedings of 7th meeting of the North American chapter of the association for computational linguistics (HLT/NAACL 07), 9 12. Rochester, NY, USA.

Bohus, D., and A. Rudnicky. 2005. LARRI: A language based maintenance and repair assistant. Spoken Multimodal Human Computer Dialogue in Mobile Environments. *Text, Speech and Language Tech nology* 28: 203 218.

Bos, J., E. Klein, O. Lemon, and T. Oka. 2003. DIPPER: Description and formalisation of an information state update dialogue system architecture. In *Proceedings of the 4th SIGdial workshop on discourse and dialogue*, 115 124. Sapporo, Japan.

Brahnam, S. 2009. Building character for artificial conversational agents: Ethos, ethics, believability, and credibility. *PsychNology Journal* 7 (1): 9 47.

Chung, G. 2004. Developing a flexible spoken dialog system using simulation. In *Proceedings of the 42nd annual meeting of the association for computational linguistics (ACL 04)*, 63 70. Barcelona, Spain.

Cuayáhuitl, H., S. Renals, O. Lemon, and H. Shimodaira. 2005. Human computer Dialogue simulation using hidden Markov models. In *Proceedings of the IEEE workshop on automatic speech recognition and understanding (ASRU'05)*, 290 295. San Juan, Puerto Rico.

Eckert, W., E. Levin, and R. Pieraccini. 1997. User modeling for spoken dialogue system evaluation. In *Proceedings of IEEE automatic speech recognition and understanding workshop (ASRU 97)*, 80 87. Santa Barbara, USA.

Eckert, W., E. Levin, and R. Pieraccini. 1998. *Automatic evaluation of spoken dialogue systems* (Technical report, TR98.9.1, ATT Labs Research).

García, F., L. Hurtado, E. Sanchis, and E. Segarra. 2003. The incorporation of confidence measures to language understanding. In *Text speech and dialogue*, *Lecture Notes in Computer Science* 2807: 165 172. Berlin, Heidelberg: Springer.

Georgila, K., J. Henderson, and O. Lemon. 2005. Learning user simulations for information state update dialogue systems. In *Proceedings of the 9th European conference on speech communication and technology (Eurospeech 05)*, 893 896. Lisbon, Portugal.

Glass, J., G. Flammia, D. Goodine, M. Phillips, J. Polifroni, S. Sakai, S. Seneff, and V. Zue. 1995. Multi lingual spoken language understanding in the MIT Voyager system. *Speech Communication* 17:1 18.

Griol, D., L. Hurtado, E. Segarra, and E. Sanchis. 2008. A statistical approach to spoken dialog systems design and evaluation. *Speech Communication* 50 (8 9): 666 682.

Griol, D., N. Sánchez Pi, J. Carbó, and J. Molina. 2010. An architecture to provide 31 context aware services by means of conversational agents. *Advances in Intelligent and Soft Computing* 79:275 282.

Heath, B., R. Hill, and F. Ciarallo. 2009. A survey of agent based modeling practices (January 1998 to July 2008). *Journal of Artificial Societies and Social Simulation* 12 (4): 9.

Jung, S., C. Lee, K. Kim, D. Lee, and G. Lee. 2011. Hybrid user intention modeling to diversify dialog simulations. *Computer Speech and Language* 25 (2): 307 326.

Klügl, F., R. Herrler, and C. Oechslein. 2003. From simulated to real environments: How to use sesam for software development. In *Multiagent System Technologies*, *Lecture Notes in Computer Science* 2831: 13 24. Berlin, Heidelberg: Springer.

Levin, E., R. Pieraccini, and W. Eckert. 2000. A stochastic model of human machine interaction for learning dialog strategies. *IEEE Transactions on Speech and Audio Processing* 8 (1): 11 23.

Lin, B., and L. Lee. 2001. Computer aided analysis and design for spoken dialogue systems based on quantitative simulations. *IEEE Transactions on Speech and Audio Processing* 9 (5): 534 548.

López Cózar, R., and M. Araki. 2005. *Spoken, multilingual and multimodal dialogue systems.* Chichester: John Wiley & Sons Publishers.

López Cózar, R., A. D. la Torre, J. Segura, A. Rubio, and V. Sánchez. 2003. Assessment of dialogue systems by means of a new simulation technique. *Speech Communication* 40 (3): 387 407.

Macal, C., and M. North. 2010. Tutorial on agent based modelling and simulation. *Journal of Simulation* 4:151 162.

McTear, M. F. 2004. *Spoken dialogue technology: Towards the conversational user interface*. London, UK: Springer Verlag.

Melin, H., A. Sandell, and M. Ihse. 2001. CTT bank: A speech controlled telephone banking system an initial evaluation. In *TMH Quarterly Progress and Status Report (TMH QPSR)* 1:1 27.

Menezes, P., F. Lerasle, J. Dias, and T. Germa. 2007. *Humanoid robots, humanlike machines. Towards an interactive humanoid companion with visual tracking modalities*, 367 398. Advanced Robotic Systems Int. and I Tech Education and Publishing.

Möller, S., R. Englert, K. Engelbrecht, V. Hafner, A. Jameson, A. Oulasvirta, A. Raake, and N. Reithinger. 2006. Memo: Towards automatic usability evaluation of spoken dialogue services by user error simulations. In *Proceedings of the 9th international conference on spoken language processing (Interspeech/ICSLP)*, 1786 1789. Pittsburgh, PA, USA.

Navarro, L., F. Flacher, and V. Corruble. 2011. Dynamic level of detail for large scale agent based urban simulations. In *Proceedings of the 10th international conference on autonomous agents and multiagent systems (AAMAS'11)*, 701 708. Taipei, Taiwan.

North, M., N. Collier, and J. Vos. 2006. Experiences creating three implementations of the repast agent modeling toolkit. *ACM Transactions on Modeling and Computer Simulation* 16:1 25.

Paek, T., and E. Horvitz. 2000. Conversation as action under uncertainty. In *Proceedings of the 16th confer ence on uncertainty in artificial intelligence*, 455 464. San Francisco, CA, USA.

Pavón, J., C. Sansores, J. Gómez, and F. Wang. 2008. Modelling and simulation of social systems with INGENIAS. *International Journal of Agent Oriented Software Engineering* 2 (2):196 221.

Pietquin, O., and T. Dutoit. 2005. A probabilistic framework for dialog simulation and optimal strategy learning. *IEEE Transactions on Speech and Audio Processing, Special Issue on Data Mining of Speech, Audio and Dialog* 14:589 599.

Sánchez Pi, N., V. Fuentes, J. Carbó, and J. Molina. 2007. Knowledge based system to define context in commercial applications. In *Proceedings of the 8th ACIS international conference on software engineer ing, artificial intelligence, networking and parallel/distributed computing (SNPD 07)*, 694 699. Tsingtao, China.

Schatzmann, J., K. Georgila, and S. Young. 2005. Quantitative evaluation of user simulation techniques for spoken dialogue systems. In *Proceedings of the 6th SIGdial workshop on discourse and dialogue*, 45 54. Lisbon, Portugal.

Schatzmann, J., B. Thomson, K. Weilhammer, H. Ye, and S. Young. 2007. Agenda based user simulation for bootstrapping a POMDP dialogue system. In *Proceedings of human language technologies 2007: The conference of the North American chapter of the association for computational linguistics (HLT/NAACL)*, 149 152. Rochester, NY, USA.

Schatzmann, J., B. Thomson, and S. Young. 2007. Statistical user simulation with a hidden agenda. In *Proceedings of the 8th SIGdial workshop on discourse and dialogue*, 273 282. Antwerp, Belgium.

Schatzmann, J., K. Weilhammer, M. Stuttle, and S. Young. 2006. A survey of statistical user simulation techniques for reinforcement learning of dialogue management strategies. *Knowledge Engineering Review* 21 (2): 97 126.

Scheffler, K., and S. Young. 2001a. Automatic learning of dialogue strategy using dialogue simulation and reinforcement learning. In *Proceedings of the human language technology conference (HLT 02)*, 12 18. San Diego, CA, USA.

Scheffler, K., and S. Young. 2001b. Corpus based dialogue simulation for automatic strategy learning and evaluation. In *Proceedings of the 2nd meeting of the North American chapter of the association for computational linguistics (NAACL 2001)*. Workshop on adaptation in dialogue systems, 64 70. Pittsburgh, PA, USA.

Torres, F., E. Sanchis, and E. Segarra. 2008. User simulation in a stochastic dialog system. *Computer, Speech and Language* 22 (3): 230 255.

Vaquero, C., O. Saz, E. Lleida, J. Marcos, and C. Canalís. 2006. VOCALIZA: An application for computer aided speech therapy in Spanish language. In *Proceedings IV Jornadas en Tecnología del Habla*, 321 326. Zaragoza, Spain.

Weng, F., S. Varges, B. Raghunathan, F. Ratiu, H. Pon Barry, B. Lathrop, Q. Zhang, T. Scheideck, H. Bratt, K. Xu, M. Purver, R. Mishra, M. Raya, S. Peters, Y. Meng, L. Cavedon, and L. Shriberg. 2006. CHAT: A conversational helper for automotive tasks. In *Proceedings of the 9th international conference on spoken language processing (Interspeech/ICSLP)*, 1061 1064. Pittsburgh, PA, USA.

Weyns, D., N. Boucké, and T. Holvoet. 2006. Gradient field based task assignment in an Agv transportation system. In *Proceedings of the 5th international conference on autonomous agents and multiagent systems (AAMAS'06)*, 842 849. Hakodate, Japan.

Wilensky, U., and W. Rand. 2012. *An introduction to agent based modeling: Modeling natural, social and engineered complex systems with NetLogo*. Cambridge, MA: MIT Press.

Williams, J., and S. Young. 2007. Partially observable markov decision processes for spoken dialog systems. *Computer, Speech and Language* 21 (2): 393 422.

Windrum, P., G. Fagiolo, and A. Moneta. 2007. Empirical validation of agent based models: Alternatives and prospects. *Journal of Artificial Societies and Social Simulation* 10 (2): 8.

Wooldridge, M., and N. Jennings. 1995. Intelligent agents: Theory and practice. *The Knowledge Engineering Review* 10:115 152.

Young, S. 2002. The statistical approach to the design of spoken dialogue systems (Technical Report, CUED/F INFENG/TR.433, Cambridge University Engineering Department, Cambridge, UK).

Zue, V., S. Seneff, J. Glass, J. Polifroni, C. Pao, T. Hazen, and L. Hetherington. 2000. JUPITER: A telephone based conversational interface for weather information. *IEEE Transactions on Speech and Audio Processing* 8 (1): 85 96.