Simulating Heterogeneous User Behaviors to Interact with Conversational Interfaces

David Griol and José Manuel Molina
Computer Science Department, Carlos III University of Madrid, Avda. de la Universidad, 30. Leganés (Spain), 28911
{david.griol, josemanuel.molina}@uc3m.es

KEYWORD

User modeling; Conversational Interfaces; Human-Machine Interaction.

ABSTRACT

Research in techniques to simulate users has a long history within the fields of language processing, speech technologies and conversational interfaces. In this paper, we describe a technique to develop heterogeneous user models that are able to interact with this kind of interfaces. By means of simulated users, it is possible not only to automatically evaluate the overall operation of a conversational interface, but also to assess the impact of the user responses on the decisions that are selected by the system. The selection of the user responses by the simulated user are based on a statistical model that considers the complete history of the interaction to carry out this selection. We describe this technique and its practical application to measure the influence of the most important user's features characteristics that affect the interaction of the simulated user with the a conversational interface.

1. Introduction

The evaluation of a conversational interface is a complex process that involves the assessment of a number of interaction parameters related to the overall operation of the system and the different modules in the architecture of these systems (McTear et al., 2016). In addition, a detailed evaluation requires the participation of a considerable number of users interacting with the system to test the different functionalities that it provides. To reduce the time and effort that is required for the evaluation, a technique that has currently attracted an increasing interest is based on the automatic generation of dialogs between the conversational interface and an additional system, called the user simulator, which represents simulated user interactions with the conversational system (Mï¿½ller et al., 2006; Schatzmann et al., 2006; Paek and Horvitz, 2000).

Simulated user models can also be used to evaluate different aspects of a conversational interface, 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 of errors or unpredicted answers in order to evaluate the capacity of the dialog manager to react to unexpected situations). Moreover, each time changes are made to the system it is necessary to complete a new evaluation to assess the impact of these changes. A second main use 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 acquired with simulated users are extremely valuable for this purpose.

In this paper, we describe a proposal for using simulated users to assess the different users' behaviors during the interaction with a dialog system. The proposed simulated users are based on a statistical user model, which is learned by means of a training dialog corpus. This model provides the probabilities of selecting each one of the user responses according to the previous dialog history and the objective of the dialog. This selection is carried out by means of a classification process that takes these information sources as input.

We have applied our proposal to evaluate the different users' behaviors interacting with a conversational interface that provides tourist information and services in Spanish. Our proposal has been used not only to evaluate the overall operation of the conversational interface, but also to detect the most important user's characteristics that have influence in the correct operation of the system and errors detected during the interaction.

The remainder of the paper is as follows. Section 2 describes the main approaches and proposals to develop user models for the interaction with conversational interfaces. Section 3 describes our proposal to develop a statistical user model providing heterogeneous behaviors. Section 4 shows the application of our proposal for the evaluation of a practical dialog system providing tourist information and services. In Section 5 we discuss the results obtained after the overall evaluation of the simulated user and the influence of different user's features in these results. Finally, Section 6 presents our conclusions and future work guidelines.

2. Related work

Two main approaches can be distinguished to the creation of simulated users: rule-based and data or corpus-based. In a rule-based simulated user, the defined rules determine the behavior of the user (Chung, 2004; Lin and Lee, 2001; Lï¿½pez-Cï¿½zar et al., 2003). This approach is particularly useful when the purpose of the research is to evaluate the effects of different dialog management strategies. In this way the researcher has complete control over the design of the evaluation study.

Corpus-based approaches are based on probabilistic methods to select the user responses, with the advantage that this uncertainty can better reflect the unexpected behaviors of users interacting with the system. Statistical models for modeling users' behavior have been suggested as the solution to the lack of the data that is required for training and evaluating conversational interfaces (Engelbrecht, 2012). Using this approach, the dialog system can explore the space of possible dialog situations and learn new potentially better strategies. A summary of corpus-based user modeling techniques for reinforcement learning of the dialog strategy can be found in (Schatzmann et al., 2006).

In (Eckert et al., 1997; Eckert et al., 1998), Eckert, Levin and Pieraccini introduced the use of statistical models to predict the next user action by means of a n-gram model. The proposed model has the advantage of being both statistical and task-independent. In (Levin et al., 2000), the bigram model is modified by considering only a set of possible user answers following a given system action. 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.

In (Scheffler and Young, 2001a; Scheffler and Young, 2001b), Scheffler and Young propose a graph-based model that requires in-depth knowledge of the task and great manual effort for the specification of all possible dialog paths. Pietquin, Beaufort 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). All model parameters are hand-selected.

Georgila, Henderson and Lemon propose the use of HMMs, defining a more detailed description of the states and considering an extended representation of the history of the dialog (Georgila et al., 2005). Cuayáhuitl et. al present a method for dialog simulation also 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 data-driven user intention simulation method that integrates diverse user discourse knowledge (cooperative, corrective, and self-directing) is presented in (Jung et al., 2011). User intention is modeled based on logistic regression and Markov logic framework.

In (Schatzmann et al., 2007a), a 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 EM-based algorithm is used to estimate optimal parameter values iteratively. In (Schatzmann et al., 2007c), the agenda-based simulator is used to train a statistical a Partially Observable MDPs (POMDPs)-based dialog manager (Williams and Young, 2007). The main drawback of this approach is due to 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, since the state space would be huge and exact POMDP optimization is intractable. As it is 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 responses each time a dialog is successfully simulated. A set of stop conditions are applied to automatically discover whether a simulated dialog has completed the predefined objectives or not.

3. Proposed user modeling technique

Usually, spoken dialog systems carry out five main tasks: Automatic Speech Recognition (ASR), Spoken Language Understanding (SLU), Dialog Management (DM), Natural Language Generation (NLG), and Text-To-Speech Synthesis (TTS). These tasks are typically implemented in different modules of the system's architecture.

The goal of speech recognition is to obtain the sequence of words uttered by a speaker (Tsilfidis et al., 2013). It is a very complex task, as there can be a great deal of variation in the input the recognizer must analyze, for example, in terms of the linguistics of the utterance, inter and intra speaker variation, the interaction context and the transmission channel. Once the speech recognizer has provided an output, the system must understand what the user said. The goal of spoken language understanding is to obtain the semantics from the recognized sentence. This process generally requires morphological, lexical, syntactical, semantic, discourse and pragmatical knowledge (Wu et al., 2010).

The dialog manager decides the next action of the dialog system (Williams and Young, 2007), interpreting the incoming semantic representation of the user input in the context of the dialog. In addition, it resolves ellipsis and anaphora, evaluates the relevance and completeness of user requests, identifies and recovers from recognition and understanding errors, retrieves information from data repositories, and decides about the next system's response. Natural language generation is the process of obtaining sentences in natural language from the non-linguistic, internal representation of information handled by the dialog system (Lemon, 2011). Finally, the TTS module transforms the generated sentences into synthesized speech (Dutoit, 1996).

The user modeling technique that we propose in this paper replaces real users in the interaction with the conversational interface. This technique simulates the user intention level, that is, the simulated user provides concepts and attributes that represent the intention of the user utterance. Therefore, the simulated user carries out the functions of the ASR and NLU modules, i.e., it generates the semantic interpretation of the user utterance in the same format defined for the output of the SLU module. Figure 1 shows the interaction of the real users and the simulated user with the described architecture of a spoken conversational interface.

The methodology that we have developed for user modeling extends our work for developing a statistical methodology for dialog management (Griol et al., 2014). The user responses are generated taking into account the information provided by the simulator throughout the history of the dialog, the last system turn, and the objective(s) predefined for the dialog.

In order to control the interaction, the simulated user uses the representation the dialogs as a sequence of pairs (A_i, U_i), where A_i is the output of the dialog system (the system answer) at time i, expressed in terms of dialog acts; and U_i is the semantic representation of the user turn (the result of the understanding process of the user input) at time i, also expressed in terms of dialog acts. This way, each dialog is represented by $(A_1, U_1), \cdots, (A_i, U_i), \cdots, (A_n, U_n)$, where A_1 is the greeting turn of the system (the first turn of the dialog), and U_n is the last user turn. We refer to a pair (A_i, U_i) as S_i, the state of the dialog sequence at time i.

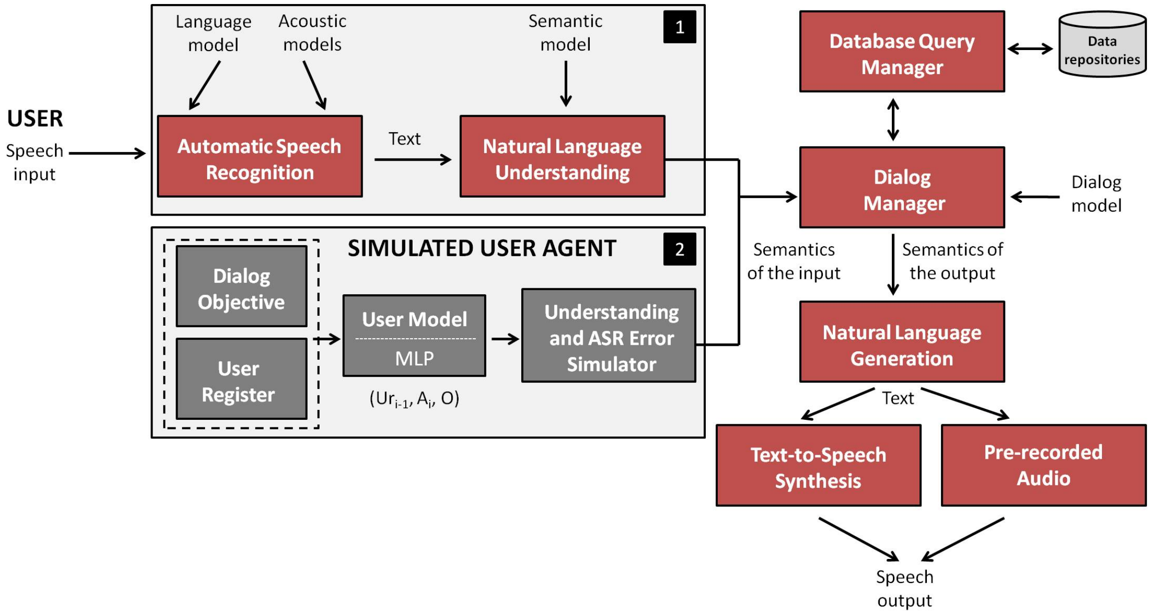

Figure 1: Interaction of a conversational interface (1) with real users. (2) with the proposed user modeling technique.

In this framework, we consider that, at time i, the objective of the simulated user is to find an appropriate user answer U_i. This selection is a local process for each time i and takes into account the sequence of dialog states that precede time i, the system answer at time i, and the objective of the dialog $\mathcal{O}$. If the most probable user answer U_i is selected at each time i, the selection is made using the maximization:

$$\hat{U}_i = \arg\max_{U_i \in \mathcal{U}} P(U_i | S_1, \cdots, S_{i-1}, A_i, \mathcal{O})$$

where set $\mathcal{U}$ contains all the possible user answers.

As the number of possible sequences of states is very large, we establish a partition in this space (i.e., in the history of the dialog preceding time i).This data structure, that we call *User Register* (UR), contains the information provided by the user throughout the previous history of the dialog. After applying the above considerations and establishing the equivalence relations in the histories of the dialogs, the selection of the best U_i is given by:

$$\hat{U}_i = \arg\max_{U_i \in \mathcal{U}} P(U_i | UR_{i-1}, A_i, \mathcal{O})$$

As in our previous work on dialog management (Griol et al., 2014), we propose the use of a multilayer perceptron (MLP) (Rumelhart et al., 1986; Bishop, 1995) to make the determination of the next user response. The input layer receives the current situation of the dialog, which is represented by the term $(UR_{i-1}, A_i, \mathcal{O})$ in the previous equation. The values of the output layer can be viewed as the a posteriori probability of selecting the different user responses defined for the user given the current situation of the dialog. The choice of the most probable user answer of this probability distribution leads to the previous equation. In this case, the simulated user will always generate the same response for the same situation of the dialog. Since we want to provide a richer variability of users behaviors, we base our choice on the probability distribution supplied by the MLP on all the feasible user responses, not only selecting the most probable user response for each dialog situation.

For the simulated user to select the next response, we have assumed that the exact values provided by the simulation model are not significant. They are important for accessing the data repositories and for constructing the output sentences of the dialog system. However, the only information necessary to determine the next action by the simulated user is the presence or absence of specific information. Therefore, the information we used from the UR is a codification of this data in terms of three values, $\{0, 1, 2\}$, for each field in the UR according to the following criteria:

- **0**: The value of the specific position of the UR has not been provided by the user.
- **1**: The value of the specific position of the UR has been provided with a confidence score that is higher than a given threshold. Confidence scores are provided by the Understanding and ASR Error Simulator, as it is explained in Section 3.1.
- **2**: The value of the specific position of the UR has been provided with a confidence score that is lower than the given threshold.

3.1 Simulating the SLU process and the communication channel

A real dialog corpus includes information about the errors that were introduced by the ASR and the SLU modules during the acquisition. This information also includes confidence measures, which are used by the conversational interface to evaluate the reliability of the concepts and attributes generated by the SLU module. This way, an error simulator has been designed to perform error generation. This module modifies the dialog acts generated by the user simulator once the UR is updated. In addition, the error simulator adds confidence scores to the semantic representation generated by the user simulator.

One of the main problems that must be considered during the interaction with a conversational interface is the propagation of errors through the different modules in the system. The ASR 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 SLU module could also add its own errors (which are mainly due to 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 about what parts of the user utterance have been clearly recognized and understood and what parts have not.

In our proposal, the simulated user provides the dialog system with the dialog act representation associated to the user input together with its confidence scores (García et al., 2003). To do this, an error simulation module has also been incorporated to include semantic errors in the generation of dialogs. This module modifies the dialog acts provided by the user model once it has selected the next user response. 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 paper, we have improved this module using a model for introducing errors based on the method presented in (Schatzmann et al., 2007b). 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 $P(c, a_u|\tilde{a}_u)$, 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 $P(\tilde{a}_u|a_u)$ is obtained by Maximum-Likelihood using the initial labeled corpus acquired with real users and 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 $P(c|\tilde{a}_u, a_u)$ 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.

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

During the automatic interaction of the simulated user and the conversational interface, the dialog manager of the dialog system considers that a dialog is not successful when one of the following conditions takes place: i) the dialog exceeds a maximum number of system turns, usually higher than the average number of turns of the dialogs acquired with real users; ii) the answer selected by the dialog manager corresponds to a query not made by the simulated user; iii) the database query module generates an error because the simulated user has not provided the mandatory data needed to carry out the query; iv) the answer generator generates an error when the selected answer involves the use of a data item not provided by the user. A user request for closing the dialog is selected once the system has provided the information defined in its objective(s). The dialogs that fulfill this condition before the maximum number of turns are considered successful.

4. Practical application

We have applied our proposal to develop and evaluate the *Enjoy Your City* spoken dialog system, which provides user-adapted tourist information in natural language in Spanish (Griol and Molina, 2015). The information provided by the system includes places of interest, weather forecast, hotel booking, restaurants and bars, shopping, street guide and "how to get there" functionalities, cultural activities (cinema, theater, music, exhibitions, literature and science), sport activities, festivities, and public transportation. The information offered to the user is extracted from different web pages and several databases are also used to store this information and automatically update the data that is provided.

We have defined ten concepts to represent the different queries that the user can perform (*Places-Interest*, *Weather-Forecast*, *Hotel-Booking*, *Restaurants-Bars*, *Shopping*, *Street-how-to-get*, *Cultural*, *Sport*, *Festivities*, and *Public-Transport*). Three task-independent concepts have also been defined for the task (*Affirmation*, *Negation*, and *Not-Understood*). A total of 115 system actions (dialog acts) were defined taking into account the information that is required by the system to provide the requested information.

An example of the semantic interpretation of a user utterance is shown in Figure 2.

The UR defined for the task is a sequence of 128 fields, corresponding to:

- The 10 concepts defined for the dialog act representation.
- The total of 115 possible attributes for the concepts.
- The 3 task-independent concepts that users can provide (*Acceptance*, *Rejection* and *Not-Understood*).

Input sentence:
[**SPANISH**] *Me gustaría conocer el horario de visita del Templo de Debod para maï¿½ana.*
[**ENGLISH**] *I would like to know the visit hours of the Temple of Debod for tomorrow.*

Semantic interpretation:
(*Places-Interest*)
Query_type: Timetables
Place: Temple of Debod
Date: Tomorrow

Figure 2: An example of the labeling of a user turn in the Enjoy Your City system.

5. Experiments and results

A corpus of 300 dialogs was acquired by means of real users interacting with the *Enjoy Your City* system. In the acquisition of this corpus participated 60 recruited users at the Technical University of Valencia (Valencia, Spain), University of Granada (Granada, Spain), and Carlos III University of Madrid (Leganés, Spain).

A 5-fold cross-validation process was used to carry out the evaluation of the proposal to develop simulated users. The corpus was randomly split into five subsets (20% of the corpus). Our experiment consisted of five trials. Each trial used a different subset taken from the five subsets as the test set, and the remaining 80% of the corpus was used as the training set. A validation subset (20%) was extracted from each training set.

In order to successfully use neural networks as classifiers, we firstly tested the influence of the topology of the MLP, by training different MLPs of increasing number of weights using the standard backpropagation algorithm (with a sigmoid activation function and a learning rate equal to 0.2), and selecting the best topology according to the mean square error (MSE) of the validation data. Different training algorithms were evaluated: the incremental version of the backpropagation algorithm (with and without momentum term) and the quickprop algorithm. The best result on the validation data was obtained using an MLP with one hidden layer of 32 units trained with the standard backpropagation algorithm and a value of LR equal to 0.3.

We defined three measures to compare the response automatically generated by the simulated user for each sample in the test partition with regard to the reference response annotated in the training corpus. This way, the evaluation is carried out turn by turn. These measures are:

- *Exact*: the percentage of responses provided by the simulated user that are exactly the same that the reference response annotated in the training corpus;
- *Coherent*: the percentage of responses provided by the simulated user that are coherent with the current state of the dialog although they are not exactly the same response annotated in the training corpus.
- *Error*: the percentage of responses provided by the user model that would cause the failure of the dialog;

Firstly, we evaluated the overall operation of the simulated user by carrying out a 5-fold cross validation process that considers only the semantic information provided by the SLU module for each user utterance, without considering any additional context information related to the user for the definition of the training and test partitions. The number of user turns in each partition considered users' location, gender, duration of the turns, and number of words provided in each turn. Table 1 shows the results of this evaluation.

These results show the satisfactory operation of the proposed user modeling technique. The codification of the state of the dialog and the correct operation of the MLP classifier allow the simulated user to generate

Exact	*Coherent*	*Error*
78.3%	94.7%	5.3%

Table 1: Results of the overall evaluation of the users modeling technique.

a response that is coherent with the current state of the dialog in a 94.7% percentage. The user response also coincides exactly with the reference response in the corpus in 78.3% of cases. Finally, the number of responses that can lead to system failure is only 5.3%.

Secondly, we completed an evaluation of the user model taking into account the size of the training corpus. The same partitions described in the overall evaluation were employed, discarding training samples randomly to reduce the size of this partition. Three experiments were completed, using a 75% of the training samples (3678 samples), 50% of the training set (2452 samples) and 25% (1226 samples). The test sets were the same described for the overall evaluation of the simulated user. Table 2 shows the results of this evaluation.

	Exact	*Coherent*	*Error*
100%	78.3%	94.7%	5.3%
75%	75.9%	91.4%	8.6%
50%	72.1%	88.3%	11.7%
25%	68.6%	82.6%	17.4%

Table 2: Results of the evaluation of the users model according to the size of the training corpus.

The results of this evaluation show the correct operation of the user model even if only used 50% of the training corpus is used to learn the user model. Thus, the results obtained for the coherent measure are very similar if more than the 50% of the training corpus is used. However, if only 25% of the training corpus is used, the percentage of responses that can cause the failure of the dialog increases to 17.4%. The good operation of the user modeling technique (even for a reduced size of the training corpus) can be explained because there are many dialog states that are very frequent in the corpus, or are similar to other states and then can be easily classified by the MLP.

Then, we evaluated our proposal considering the gender of the users as a parameter to be assessed. To do this, the corpus was divided into a set of partitions with equal number of samples of women and men. Table 3 shows the results of this evaluation, specifying the partitions used for training and test (Training/Test).

	Exact	*Coherent*	*Error*
Women / Both	70.6%	89.1%	10.9%
Men / Both	70.3%	93.1%	6.9%
Both / Women	71.8%	92.5%	7.5%
Both / Men	77.1%	93.4%	6.6%

Table 3: Results of the evaluation of the user model taking into account the influence of gender.

The results of this evaluation show that there are not remarkable differences if the learning of the user model for the simulated user is completed using only samples of men or women (first two columns of results in Table 3). Higher differences are observed in the evaluation of the model considering the gender of the users in the test partitions (third and fourth column of this table). The differences obtained in these cases show a greater similarity in the samples of men.

Following, we evaluated the influence of the users' expertise level in the operation of the user modeling technique. Users were classified into three groups: Group 1 (users that employed the system 5 or less times), Group 2 (users that employed the system between 5 and 10 times), and Group 3 (users that employed the system more than 10 times). Table 4 shows the results of this experimentation, specifying the partitions used for training and test (Training/Test).

	Exact	*Coherent*	*Error*
Group 1 / Group 2	61.4%	66.2%	33.8%
Group 1 / Group 3	53.8%	59.1%	40.9%
Group 2 / Group 1	67.6%	75.9%	24.1%
Group 2 / Group 3	68.3%	78.3%	21.7%
Group 3 / Group 1	60.7%	69.3%	30.7%
Group 3 / Group 2	72.5%	80.8%	19.2%

Table 4: Results of the evaluation of the user model taking into account the influence of age.

As the results of Table 4 show, the more significant differences were observed when novel users (Group 1) where employed for training the user model and the test partition included the rest of users. These differences are very important when users in Group 1 were used for training and users of Group 3 were employed to test the user model.

Finally, we have evaluated our proposal taking into account the influence of the origin of the dialogs. This evaluation starting with the same partitions defined for the overall evaluation of the proposal, training the simulated user with the samples coming from the specific location to be evaluated and using the same test partitions (samples from the three locations). Table 5 shows the results of this evaluation.

	Exact	*Coherent*	*Error*
Location 1	71.9%	84.8%	15.2%
Location 2	77.2%	91.3%	8.7%
Location 3	66.4%	82.6%	17.4%

Table 5: Results of the evaluation of the user model taking into account the origin of the dialogs.

The results of this evaluation show the better operation of the proposal when the user model was learned with the dialogs acquired at Location 2. When the user model was learned using only the dialogs of the Location 1, the percentage of responses that follow the strategy is equivalent to the one obtained for the Location 3. With regard the dialog corpus acquired at Location 3, in addition of obtaining a percentage of exact responses of only 66.4%, the number of user responses that can cause the failure of the dialog is also the highest. Therefore, considering the values obtained for the different measures, we can conclude a significant difference between the dialogs acquired at each location.

6. Conclusions and future work

In this paper, we have described a technique to generate simulated users to automatically evaluate spoken dialog systems. The simulated user is based on a statistical model which takes the complete history of the interaction into account to decide the next user response. This decision is modeled by a classification process in which a neural network is used. An additional statistical model has been introduced for errors introduction and confidence measures generation. This way, the dialog system can also be evaluated by considering different conditions in the communication channel.

The simulated user provides the user intention level in terms of the semantic representation that would be generated by the ASR and NLU modules in the architecture of a dialog system. This way, dialogs are automatically labeled during the simulation using the semantics defined for the task. Thus, the interaction of the simulated user and a dialog system allows the generation of new dialogs with little effort and the adaptation of a the system to a new task can also simplified.

We have described the application of our proposal to evaluate both the overall operation of the user model and the main characteristics of a dialog corpus acquired with real users. This evaluation has allowed us to measure the influence of the most important user's features characteristics that affect the interaction of the simulated user with the dialog system. As a future work, we are adapting the proposed user modeling technique for its application in more difficult domains. We also want to extend our proposal for user modeling by means of the incorporation of additional features related to the user's emotional state and their personality.

7. References

Bishop, C. M., 1995. *Neural networks for pattern recognition*. Oxford University Press.

Chung, G., 2004. Developing a flexible spoken dialog system using simulation. In *Proc. of 42nd Annual Meeting of the Association for Computational Linguistics (ACL'04)*, pages 63–70. Barcelona, Spain.

Cuayáhuitl, H., Renals, S., Lemon, O., and Shimodaira, H., 2005. Human-Computer Dialogue Simulation Using Hidden Markov Models. In *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU'05)*, pages 290–295. San Juan, Puerto Rico.

Dutoit, T., 1996. *An introduction to text-to-speech synthesis*. Kluwer Academic Publishers.

Eckert, W., Levin, E., and Pieraccini, R., 1997. User modeling for spoken dialogue system evaluation. In *Proc. of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU'97)*, pages 80–87. Santa Barbara, USA.

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

Engelbrecht, K., 2012. *Estimating Spoken Dialog System Quality with User Models*. Springer.

García, F., Hurtado, L., Sanchis, E., and Segarra, E., 2003. The incorporation of Confidence Measures to Language Understanding. *Lecture Notes in Computer Science*, 2807:165–172.

Georgila, K., Henderson, J., and Lemon, O., 2005. Learning user simulations for information state update dialogue systems. In *Proc. of European Conference on Speech Communications and Technology (Eurospeech'05)*,
pages 893–896. Lisbon, Portugal.

Griol, D., Callejas, Z., López-Cózar, R., and Riccardi, G., 2014. A domain-independent statistical methodology for dialog management in spoken dialog systems. *Computer, Speech and Language*, 28(3):743–768.

Griol, D. and Molina, J., 2015. Modeling Users Emotional State for an Enhanced Human-Machine Interaction. In *Proc. of 10th International Conference on Hybrid Artificial Intelligence Systems (HAIS'15)*, pages 357–368. Bilbao, Spain.

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

Lemon, O., 2011. Learning what to say and how to say it: Joint optimisation of spoken dialogue management and natural language generation. *Computer Speech & Language*, 25:210–221.

Levin, E., Pieraccini, R., and Eckert, W., 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 Lee, L., 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., de la Torre, A., Segura, J., and Rubio, A., 2003. Assessment of dialogue systems by means of a new simulation technique. *Speech Communication*, 40:387–407.

McTear, M. F., Callejas, Z., and Griol, D., 2016. *The Conversational Interface*. Springer.

Mïller, S., Englert, R., Engelbrecht, K., Hafner, V., Jameson, A., Oulasvirta, A., Raake, A., and Reithinger, N., 2006. MeMo: towards automatic usability evaluation of spoken dialogue services by user error simulations. In *Proc. of the 9th Int. Conference on Spoken Language Processing (Interspeech/ICSLP)*, pages 1786–1789. Pittsburgh, USA.

Paek, T. and Horvitz, E., 2000. Conversation as Action Under Uncertainty. In *Proc. of 16th Conference on Uncertainty in Artificial Intelligence (UAI'00)*, pages 455–464. San Francisco, USA.

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

Rumelhart, D. E., Hinton, G. E., and Williams, R. J., 1986. *PDP: Computational models of cognition and perception, I*, chapter Learning internal representations by error propagation, pages 319–362. MIT Press.

Schatzmann, J., Thomson, B., Weilhammer, K., Ye, H., and Young, S., 2007a. Agenda-Based User Simulation for Bootstrapping a POMDP Dialogue System. In *Proc. of Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 149–152. Rochester, USA.

Schatzmann, J., Thomson, B., and Young, S., 2007b. Error Simulation for Training Statistical Dialogue Systems. In *Proc. of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU'07)*, pages 273–282. Kyoto, Japan.

Schatzmann, J., Thomson, B., and Young, S., 2007c. Statistical User Simulation with a Hidden Agenda. In *Proc. of 8th SIGdial Workshop on Discourse and Dialogue*, pages 273–282. Antwerp, Belgium.

Schatzmann, J., Weilhammer, K., Stuttle, M., and Young, S., 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 Young, S., 2001a. Automatic learning of dialogue strategy using dialogue simulation and reinforcement learning. In *Proc. of Second International Conference on Human Language Technology Research (HLT'02)*, pages 12–18. San Diego, USA.

Scheffler, K. and Young, S., 2001b. Corpus-based Dialogue Simulation for Automatic Strategy Learning and Evaluation. In *Proc. of Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 64–70.

Tsilfidis, A., Mporas, I., Mourjopoulos, J., and Fakotakis, N., 2013. Automatic speech recognition performance in different room acoustic environments with and without dereverberation preprocessing. *Computer Speech & Language*, 27(1):380–395.

Williams, J. and Young, S., 2007. Partially Observable Markov Decision Processes for Spoken Dialog Systems. *Computer Speech and Language*, 21(2):393–422.

Wu, W.-L., Lu, R.-Z., Duan, J.-Y., Liu, H., Gao, F., and Chen, Y.-Q., 2010. Spoken language understanding using weakly supervised learning. *Computer Speech & Language*, 24(2):358–382.