CONSISTENT GOAL-DIRECTED USER MODEL FOR REALISITC MAN-MACHINE TASK-ORIENTED SPOKEN DIALOGUE SIMULATION

Olivier Pietquin[†]

École Supérieure d'Électricité – SUPELEC
Metz Campus – STS Team
2 rue Édouard Belin – F-57070 Metz – FRANCE
email: olivier.pietquin@supelec.fr

ABSTRACT

Because of the great variability of factors to take into account, designing a spoken dialogue system is still a tailoring task. Rapid design and reusability of previous work is made very difficult. For these reasons, the application of machine learning methods to dialogue strategy optimization has become a leading subject of researches this last decade. Yet, techniques such as reinforcement learning are very demanding in training data while obtaining a substantial amount of data in the particular case of spoken dialogues is time-consuming and therefore expansive. In order to expand existing data sets, dialogue simulation techniques are becoming a standard solution.

In this paper we describe a user modeling technique for realistic simulation of man-machine goal-directed spoken dialogues. This model, based on a stochastic description of man-machine communication, unlike previously proposed models, is consistent along the interaction according to its history and a predefined user goal.

1. INTRODUCTION

The design of an efficient spoken dialogue system (SDS) does not simply consist in combining speech processing systems. Indeed, it requires the development of a management strategy taking into account the performances of these systems, the nature of the task (form filling, database querying etc.) and the user's behavior. The great variability of these factors makes rapid design of dialogue strategies and reusability of previous work very difficult. For these reasons, automatic learning of optimal strategies is currently a leading domain of researches [1][2][3][4]. Yet, the lack of data for learning and testing dialogue strategies led to a new field of researches: man-machine spoken dialogue stochastic modeling and simulation [2][3][4][5][6][7].

Among simulation methods developed so far, one can distinguish between state-transition or global methods like proposed in [2] and methods based on modular simulation environments as described in [4][5][6][7]. The first type of methods is very task-dependent as well as the mixed method proposed in [3]. Moreover, using this type of methods for strategy learning can only lead to the learning of the best strategy used in the data corpus which is not always optimal. The second type of methods intends to be more task-independent by integrating models of each component of a SDS including the speech processing systems but also the user like depicted on Figure 1. Although it makes use of a complex modular simulation environment, the method presented in [5] stays very task-dependent and even system-dependent since it requires recordings of spoken utterances collected during real interactions and real implementations of speech processing systems such as an Automatic Speech Recognition (ASR) system. It is therefore out of the scope of this paper and we will focus on generic simulation methods [4][6][7] considering the dialogue at the intention level (see section 2.1) and not at the acoustic level like in [5].

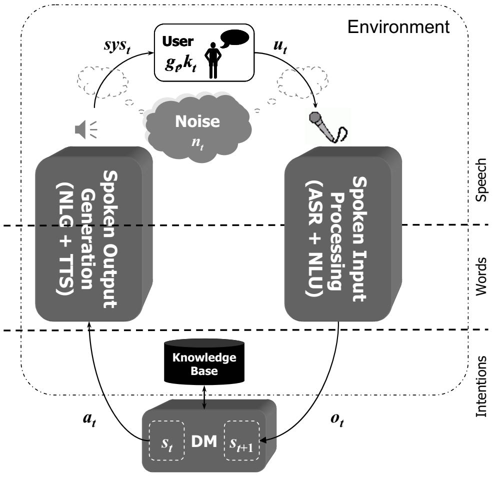

Figure 1: SDS environment

Although there is a complexity difference between the two simulation environments described in [6] and in [4][7], they both include a user model. User modeling for man-machine spoken dialogue simulation is a challenging issue and of a crucial importance for the efficiency of machine learning methods. It is currently an important domain of investigations within the broad field of research on SDS.

In this paper we describe a new user model based on a probabilistic description of man-machine spoken communication. We

[†] This work was realized when the author was with the Faculty of Engineering, Mons (FPMs, Belgium) and was sponsored by the 'Direction Générale des Technologies, de la Recherche et de l'Énergie' (DGTRE) of the Walloon Region (Belgium, First Europe Convention n° 991/4351).

1-4244-0367-7/06/$20.00 ©2006 IEEE

also provide examples of generated dialogues and compare them with dialogues obtained with other models.

2. PRELIMINARY CONSIDERATIONS

2.1. Intention-based modeling

In our vision of a simulated environment, communication between modules takes place at the intention level rather than at the word sequence or speech signal level, as it would be in real-world applications and like proposed in [5]. We regard an intention as the minimal unit of information that a dialogue participant can express independently. Intentions are closely related to concepts, speech acts or dialogue acts.

In the aim of learning strategies, there is no point to model dialogue at a lower level because a dialogue strategy is a high level concept. Furthermore, concept-based communication allows error modeling of all the parts of the system, including natural language understanding [8]. More pragmatically, it is easier to automatically generate concepts compared with word sequences (and certainly speech signals), as a large number of utterances can express the same intention.

2.2. State of the art

Statistical user modeling for spoken dialogue simulation is quite a recent field of investigations. Indeed, user modeling is commonly used for system adaptation to users' preferences or goal inference [9]. Yet, in [10] the authors proposed a first bigram model based on the following development. Naming ***sys*$_t$** the system utterance at time ***t*** and ***u*$_t$** the user's utterance at time t, the probability of the user saying u_t considering the interaction history is given by:

$$P(u_t \mid sys_t, u_{t-1}, sys_{t-1} ..., u_0, sys_0) \approx P(u_t \mid sys_t) \qquad (1)$$

This conditional probability distribution has to be learned from a data corpus but it turned out that no existing corpus could be used to learn such a distribution accurately, and certainly not the ATIS corpus which was the basis of this work. Therefore in [6] and [10], same authors proposed to learn a restricted set of parameters describing the behavior of a user answering to predefined types of system utterances such as the greeting, constraining questions, confirmations or relaxation requests. However, these two models don't ensure consistency of the user's behavior over the course of the dialogue with regard to a goal or even to the history of the interaction.

sys_0	*Hello! How may I help you?*
u_0	I'd like to go to Paris.
sys_1	*What is your departure City?*
u_1	I'd like to leave from Brussels.
sys_2	*Ok, please confirm the following request: you want a business class ticket to go from Brussels to Paris?*
u_2	No, I want to leave from Paris.
sys_3	*You want go from Paris to Paris?*
u_3	Yes.
sys_4	*It is impossible to go from Paris to Paris. Do you want to choose another destination?*
…	…

Table 1: Typical dialogue with the bigram model

Therefore typical problematic dialogues like shown in Table 1 can occur. In this example, from a train ticket booking application log, intentions have been expanded to real utterances for clarity purposes.

Moreover, depending on the number of possible answers to a single question (e.g. departure city), the average duration in turns of a simulated dialogue can vary significantly which is of course an undesired effect.

3. PROPOSED USER MODEL

3.1. Statistical model of man-machine spoken dialogue

The user model proposed in this paper is based on a statistical description of man-machine spoken communication previously described in [4] and [7]. A simplified version of this description, omitting details related to speech processing modules, will be used here since we focus on the user behavior. Using notations of Figure 1, a dialogue can be seen as a turn-taking process in which a user and a Dialogue Manager (DM) communicate through speech processing modules. At each turn t, according to its internal state $\boldsymbol{s_t}$ (usually describing the history and the context of the current interaction), the DM produces a set of intentions or dialogue acts $\boldsymbol{a_t}$. This set a_t is then transformed into a system spoken utterance sys_t by spoken output generation systems such as Natural Language Generation (NLG) and Text-To-Speech (TTS) synthesis systems. According to his/her goal $\boldsymbol{g_t}$ and knowledge $\boldsymbol{k_t}$, the user produces an utterance u_t answering to the system solicitation sys_t. Both u_t and sys_t can be mixed with environmental noise $\boldsymbol{n_t}$. The utterance u_t is in turn processed by spoken input processing systems such as ASR and Natural Language Understanding (NLU) systems to produce an observation $\boldsymbol{o_t}$ composed of concepts extracted from u_t and of confidence measures. This observation is finally used by the DM to update its internal state. From this description, the interaction can be described thanks to the following joint probability:

$$P(s_{t+1}, o_t, a_t \mid s_t, n_t) = \underbrace{P(s_{t+1} \mid o_t, a_t, s_t, n_t)}_{\text{Task Model}} \cdot \underbrace{P(o_t \mid a_t, s_t, n_t)}_{\text{Environment}} \cdot \underbrace{P(a_t \mid s_t, n_t)}_{\text{DM}} \qquad (2)$$

The factorization of this joint probability includes a term related to the environment processing of the DM intention set (second term). Omitting the t indices, this term can in turn be factored as follow:

$$\begin{aligned} P(o \mid a, s, n) &= \sum_{sys,k,g,u} P(o, sys, k, g, u \mid a, s, n) \\ &= \sum_{sys,k,g,u} \underbrace{P(sys \mid a, s, n)}_{\text{Output Generation}} \cdot \underbrace{P(o \mid u, g, sys, a, s, n)}_{\text{Input Processing}} \cdot \qquad (3) \\ &\qquad \underbrace{P(u, k, g \mid sys, a, s, n)}_{\text{User Behavior}} \end{aligned}$$

The last term corresponds to the user's behavior that we want to model.

3.2. User Model

From the previous section, the user behavior can be probabilistically described by the following joint probability:

$$
\begin{aligned}
&P(u,g,k|sys,a,s,n)\\
&\quad = P(k \mid sys,a,s,n)\cdot P(g \mid k,sys,a,s,n)\cdot\ P(u \mid g,k,sys,a,s,n) \quad (4)\\
&\quad = \underbrace{P(k \mid sys,s,n)}_{\text{Knowledge Update}}\cdot \underbrace{P(g \mid k)}_{\text{Goal Modification}}\cdot \underbrace{P(u \mid g,k,sys,n)}_{\text{User Output}}
\end{aligned}
$$

To obtain the last equality, the following assumptions were made:

- the user is only informed of the DM intentions a_t through the system utterance sys_t,
- if a goal modification occurs it is because the user's knowledge has been updated by the last system utterance.

Equation (4) emphasizes on the tight relation existing between the user's utterance production process and his/her goal and knowledge, themselves linked together. The user's knowledge can be modified during the interaction according to the speech outputs produced by the system. Yet, such a modification of the knowledge is incremental (it is an update to compare with the system state update) and it takes into account the last system utterance (which might be misunderstood, and especially in presence of noise) and the previous user's knowledge state. This can be written as follow with $\boldsymbol{k}^{-}$ standing for k_{t-1}:

$$
\begin{aligned}
P(k \mid sys,s,n) &= \sum_{k^-} P(k \mid k^-,sys,s,n)\cdot P(k^- \mid sys,s,n)\\
&= \sum_{k^-} P(k \mid k^-,sys,n)\cdot P(k^- \mid s)
\end{aligned} \quad (5)
$$

Although the user's knowledge k^- is not directly dependent of the system state s, we kept this dependency in our description so as to be able to introduce a mechanism for user knowledge inference from system state because it is supposed to contain information about the history of the dialogue. Such a mechanism can be helpful if one wants to introduce grounding subdialogues [11] in the interaction so as to obtain a good connection between the user's understanding of the interaction and the system view of the same interaction.

4. PRACTICAL EXAMPLE

In this section, we will use a simple train ticket booking application as already mentioned in section 2.2 so as to provide a practical example of use of the proposed user model. The task will consist in filling a 5-slot form which slots are: departure city, arrival city, desired departure time, desired arrival time, class.

4.1. Variable representation

In practice, the use of the proposed framework is difficult without a suitable representation of variables such as *u*, *sys*, *g* or *k*. According to the intention-based communication paradigm, these variables can be regarded as finite sets of abstract concepts, related to the specific task, that have to be manipulated along the interactions by the SDS and the user. Consequently, we opted for an Attribute-Value (AV) pair variable representation based on the Attribute-Value Matrix (AVM) representation of the task proposed in [12].

For the application we consider here, the AVM representation of the task is obtained by associating an attribute to each of the 5 slots. Than, to each intention or dialogue act corresponds a set of AV pairs. For instance, {"departure city", "arrival city" …} are attributes and possible values are {Namur, Brussels, Paris …}. The utterance "I want to go from Namur to Brussels" can therefore be represented by the following set of AV pairs:

$$
\begin{aligned}
u_t &= [\{\text{dep\_city} = \text{'Bruxelles'}\},\\
&\quad \{\text{arr\_city} = \text{'Paris'}\}]
\end{aligned} \quad (6)
$$

In Practice, we used 50 possible values for the cities, 48 values for the times (every half of an hour) and 2 values for the class (economy and business). So as to model the system and user's utterances, we added attributes and values to this description. The first attribute is the type of system utterance (S_A in the following), which can for instance take the following values: {'Greeting', 'Constraining Question', 'Open Question', 'Confirmation', 'Relaxation request', 'Closing'}. The second is a binary attribute corresponding to the user's will of closing the dialogue ($U_C \in$ {*true*, *false*}). Finally we also added attributes associated to user's answers to confirmation and relaxation prompts taking Boolean values.

The system utterances are therefore of the form:

sys = {[S_A = 'const_q'], [s^1 = 'dep_city']} or
sys = { const_q (dep_city)} in the following

The user's utterances are of the form:

u = {[U_C = *false*], [dep_city = 'Bruxelles']}

Beside the simplicity of use, this representation has an additional advantage: it allows the modeling of ASR and NLU errors when simulating the complete dialogue process as described in [4] [8][13].

4.2. Model Initialization

To apply the proposed user model for task-oriented dialogue simulation, it is mandatory to initialize the model with a goal and a knowledge structure. For the simple task we consider in this example, the proposed goal and knowledge structure is show in Table 2.

In this structure, the goal includes values for each of the task attributes (we consider in this example that the user has a preference for each of the attributes but it is not mandatory since an empty value could be considered) but also a priority value associated to each attribute. This value indicates how important it is to the user to transmit the associated attribute to the system (e.g. when asked to relax one attribute, the user is more likely to accept the relaxation of constrains linked to *low* priorities).

Goal			Knowledge
Attribute	Value	Priority	Count
Dep. City	Bruxelles	*high*	$k^1 = 0$
Arr. City	Paris	*high*	$k^2 = 0$
Dep. Time	8.30 AM	*low*	$k^3 = 0$
Arr. Time	1.00 PM	*high*	$k^4 = 0$
Class	Business	*low*	$k^5 = 0$

Table 2: Initial goal and knowledge

The knowledge is very simply modeled as a set of counters, each of them is associated to one attribute and incremented each time the user is asked the corresponding value by the system.

4.3. Model Parameters

It is obvious from eq. 4 and 5 that the parameters of the model are conditional probabilities describing the behavior of the model according to system utterances and the user's goal and knowledge. With the AV variable description these probabilities can be factored and a set of discrete conditional probability distributions have to be assessed so as to obtain a complete model.

For instance, the probability

P(*u* = {[dep_city = ‘Bruxelles’]} | *sys* = {[S_A = ‘const_q’], [s^1 = ‘dep_city’}, *g* = {[dep_city = ‘Bruxelles’], …}, *k* = {[k^1 = 0], [k^2 =0], …})

is the probability that the user’s utterance contains the value ‘Bruxelles’ for the departure city when the s/he is asked by the system to provide the value of the departure city and knowing that Bruxelles is in the goal and that the departure city has not been provided yet (k^1 = 0). These parameters can either be learned from suitable annotated data or handcrafted by experts.

4.4. Results

Table 3 shows an example of a simulated dialogue using the proposed user model. To generate this example and show the robustness of the user model, we used a random dialogue policy in which the system always starts with a greeting prompt and subsequently randomly chose any type and combination of intentions.

	Intentions	k	Expanded Intentions
sys_0	[S_A = greeting]		*Hello! How may I help you?*
u_0	[arr_city = ‘Paris’]	k^2 = 1	I’d like to go to Paris.
sys_1	S_A = const(arr_time)		*When do you prefer to arrive?*
u_1	[arr_time = ‘1.00 PM’]	k^1 = 1	I want to arrive around 1 PM.
sys_2	[S_A = rel(arr_time)}		*Don’t you prefer to arrive later?*
u_2	[rel = *false*] (p = *high*)		No.
sys_3	[S_A=conf(arr_city)]		*Can you confirm you want to go to Bruxelles?*
u_3	[conf = *true*]	k^2 = 2	Yes !
…	…	…	…
…	…	…	…
sys_i	[S_A = const(arr_city)]		*Where do you want to go ?*
u_i	[U_C = *true*]	k^2 = 3	Ok, bye! (*hang off*)

Table 3: Example of simulated dialogue

The second column of Table 3 shows the counters increments standing for the knowledge update. This example illustrates the consistency of the model which provides and confirms correct values according to its goal. It also shows two main points of the goal-directed model. First, the user closes the dialogue because the system behaved very badly which is detected by the user’s knowledge update. Second, a relaxation query is denied according to the goal priorities. The resulting dialogue seems more realistic and this model was used to generate learning data for optimal dialogue strategy search and provided promising results [4].

5. CONCLUSIONS AND PERSPECTIVES

In this paper, we proposed a probabilistic user model for simulating consistent goal-directed behavior in task-oriented dialogues. This user model takes into account a goal but also a knowledge representation allowing consistency all along the interaction according to the goal and the dialogue history. It is based on a probabilistic representation of man-machine spoken communication as an intention exchange process. So as to use practically this probabilistic model, we had to reduce the complexity of parameters by representing the task and the intentions as attribute-value sets. The application on a simple task shows promising results and the model could be inserted with success in a machine learning system (this was a first naïve evaluation method).

Although we tried to reduce the complexity of the model by using AV pairs, there is still a large number of parameters in the model. The low amount of suitable data to train this kind of models doesn’t ensure that the results statistically sound and some of the parameters had to be handcrafted. A side product of this research is thus a better understanding of what should be annotated in terms of dialogue context in data corpora.

Finally, in order to analyze more formally the results provided by our model, some objective and quantitative metrics should be used. Independent researches already started on this topic and showed that this model performs better than previous ones [14].

6. REFERENCES

[1] E. Levin, R. Pieraccini, W. Eckert, ‘Learning Dialogue Strategies within the Markov Decision Process Framework.’ *Proc. ASRU’97*, Santa Barbara, California, 1997.

[2] S. Singh, M. Kearns, D. Litman, M. Walker, ‘Reinforcement Learning for Spoken Dialogue Systems.’ *Proc. NIPS’99*, Denver, USA, 1999.

[3] K. Scheffler and S. Young, ‘Corpus-Based Dialogue Simulation for Automatic Strategy Learning and Evaluation.’ *Proc. NAACL Workshop on Adaptation in Dialogue Systems*, 2001.

[4] O. Pietquin and T. Dutoit, ‘A Probabilistic Framework for Dialog Simulation and Optimal Strategy Learning.’ *IEEE Transactions on Audio, Speech and Language Processing, Volume 14, Issue 2,* March 2006, pp 589-599.

[5] R. López-Cózar, A. de la Torre, J. Segura, A. Rubio ‘Assesment of Dialogue Systems by Means of a New Simulation Technique.’ *Speech Communication*, vol. 40, 2003.

[6] E. Levin, R. Pieraccini, W. Eckert, “A Stochastic Model of Human-Machine Interaction for learning dialog Strategies”, *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 1, pp. 11-23, 2000.

[7] O. Pietquin, ‘A Probabilistic Description of Man-Machine Spoken Communication’ *Proc. ICME 2005*, Amsterdam, The Netherlands, 2005.

[8] O. Pietquin and T. Dutoit, ‘Dynamic Bayesian Networks for NLU Simulation with Applications to Dialog Optimal Strategy Learning’, *Proc. ICASSP’06*, Toulouse, France, 2006

[9] I. Zukerman and D. Albrecht, ‘Predictive Statistical Models for User Modeling.’ *User Modeling and User-Adapted Interaction*, vol. 11(1-2), 2001, pp. 5-18.

[10] W. Eckert, E. Levin, R. Pieraccini, ‘User Modeling for Spoken Dialogue System Evaluation.’ *Proc. ASRU’97*, 1997.

[11] H. Clark, E. Schaefer, ‘Contributing to Discourse.’ *Cognitive Science*, 13, 1989, pp.259-294.

[12] M. Walker, D. Litman, C. Kamm, A. Abella, ‘PARADISE: A Framework for Evaluating Spoken Dialogue Agents.’ *Proc. 35th Annual Meeting of the ACL*, Madrid, Spain, 1997.

[13] O. Pietquin, R. Beaufort, ‘Comparing ASR Modeling Methods for Spoken Dialogue Simulation and Optimal Strategy Learning.’ *Proc. of Eurospeech’05*, Lisbon, Portugal, 2005.

[14] J. Schatzmann, K. Georgila, and S. Young. ‘Quantitative Evaluation of User Simulation Techniques for Spoken Dialogue Systems’. *Proc. SIGdial’05*, Lisbon, Portugal, 2005