Evaluating Spoken Dialogue Models under the Interactive Pattern Recognition Framework

Fabrizio Ghigi[1], *M. Inés Torres*[1], *Raquel Justo*[1], *José-Miguel Benedí*[2]

[1]Dpto. Electricidad y Electrónica. Universidad del País Vasco, Spain
[2]Instituto Tecnológico de Informática. Universitat Politècnica de València, Spain
fabrizio.ghigi@ehu.es, manes.torres@ehu.es, raquel.justo@ehu.es, jbenedi@dsic.upv.es

Abstract

The new Interactive Pattern Recognition (IPR) framework has been proposed to deal with human-machine interaction. In this context a new formulation has been recently defined to represent a Spoken Dialogue System as an IPR problem. In this work this formulation is applied to define graphical models that deal with Spoken Dialogue Systems. The definition of both a Dialogue Manager and a User Model are shown and the estimation of the parameters and smoothing techniques are presented in the paper. These models were evaluated in a dialogue generation task on two very different corpora: Dihana corpus consisting of Spanish spoken dialogues acquired with the Wizard of Oz technique and Let's Go corpus consisting of spoken dialogues in English between real users and the Ravenclaw dialogue manager developed by CMU. The results obtained show that original and simulated dialogues exhibited very similar behaviours, thus demonstrating the learning capacity of the proposed models in both a controlled Wizard of Oz task and a spoken dialogue system that interacts with real users. This formulation can then be considered as a promising framework to deal with Spoken Dialogue Systems.

Index Terms: spoken dialogue, pattern recognition

1. Introduction

Spoken Dialogue Systems (SDS) aim to enable people to interact with computers, using spoken language in a natural way [1, 2]. However, the management of a SDS is a very complex task that involves many other problems to be solved like the Automatic Speech Recognition (ASR), semantic representation and understanding, answer generation, etc. The Dialogue Manager (DM) is the main component of a SDS. According to the information provided by the user and the history of previous dialogues the DM must decide the next action to be taken. Due to its complexity the design of DM has been traditionally related to rules based methodologies [3], sometimes combined with some statistical knowledge [4]. These kind of methodologies have been successfully used for specific tasks. However, a big effort has to be spent in rewriting the rules that govern the dialogue. Thus, this kind of DMs are hard to adapt to different tasks and they lack sensitivity to changes present in real world tasks. As an alternative to these methodologies, a plan-based and task-independent Ravenclaw DM has been developed [5] over the Olympus framework created at Carnegie Melon University. Ravenclaw has been successfully used to develop the Let's Go bus information system for the general public [6]. In addition, statistical models based on Markov Decision Processes can be found in the literature for DMs [7, 8]. Partially Observable Markov Decision Process (POMDP) provided an excellent statistical framework that admits global optimization and deal with uncertainty of user goals [9]. This formulation is now considered as the state of the art of statistical SDSs. However, its put in practice entails intractable problems that need efficient and suboptimal approaches such as factorization of the states space and partition of the dialogue state distributions [9, 10, 11, 12].

In the human-machine interaction context, the new Interactive Pattern Recognition (IPR) framework has also been proposed [13]. According to this paradigm, Pattern Recognition (PR) systems design is shifting from the concept of full-automation to systems where the decision process is conditioned by human feedback. IPR has been successfully applied to some classical PR problems such as interactive transcription of handwritten and spoken documents, computer assisted translation, interactive text generation and parsing, among others [13]. Recently, a new formulation to model SDS within IPR framework has also been proposed in a previous work [14]. To this end some extensions to the IPR approach have been proposed to deal with both speech and text-based dialogue systems. Additionally, a user model based on the IPR paradigm has also been defined [14].

The aim of this work is to put in practice and evaluate this formulation, which is summarized in Section 2. To this end we deal with the parameter estimation of proposed graphical models as well as model smoothing to manage unforeseen situations in Section 3. These models were then evaluated in a dialogue generation task on two very different corpora: Dihana corpus consisting of Spanish spoken dialogues acquired with the Wizard of Oz technique and Let's Go corpus consisting of spoken dialogues between general public users and the Ravenclaw DM. These experiments are presented in Section 4. The results obtained for the two tasks are equally promising and they allow to consider these models as an alternative to deal with SDSs.

2. Spoken Dialogue Systems and IPR

Let h be an hypothesis or output derived by a classical PR system from some input stimulus. Let $\mathcal{M}$ be a model used by the system to derive its hypotheses, which is obtained through a batch learning procedure. Under the IPR framework [13] the user provides some feedback signals, f, which may iteratively help the system to improve its hypothesis. In SDS we assumed that the systems interacts with the user providing a first hypothesis h through a greeting turn that acts as unique stimulus. Ignoring the user feedback except for the last interaction and hypothesis h and assuming a classical PR *minimum-error* criterion the Bayes decision rule is simplified to maximize the posterior $Pr(h|h\ , f)$ [13]. In a SDS the interpretation or decoding d

of the user feedback f cannot be considered as a deterministic process. In fact the space of decoded feedback signals is the output of an ASR system. Thus, a best hypothesis $\hat{h}$ is obtained as follows [14]:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} P(h|h\ , f) = \arg\max_{h \in \mathcal{H}} \sum_{d} P(h, d|h\ , f)$$
$$\arg\max_{h \in \mathcal{H}} \max_{d} P(h|d, h\ )P(f|d)P(d|h\ ) \quad (1)$$

A suboptimal approach can be considered through a two step decoding: find first an optimal user feedback $\hat{d}$ and then, use $\hat{d}$ to decode system hypothesis $\hat{h}$ as follows:

$$\hat{d} = \arg\max_{d} P(f|d)P(d|h\ ) \quad (2)$$

$$\hat{h} \quad \arg\max_{h \in \mathcal{H}} P(h|\hat{d}, h\ ) \quad (3)$$

In this work the action to be taken by the DM at each interaction step is based on Equation (3), thus assuming a classical minimum-error criterion. However, different criteria could also be considered. In fact, in SDS the DM strategy is usually based on maximizing the probability of achieving the unknown user goals while minimizing the cost of getting them [9, 15].

2.1. Simulated User

The development of a complete SDS requires an online learning to train the DM. Therefore, a large amount of dialogues are needed together with real users with different goals, expectations and behaviours. Thus, statistical DMs are usually trained by Simulated Users (SU) [16]. A SU must provide the feedback f to the system at each interaction step. The user feedback f depends on its previous feedback f according to some unknown distribution $P(f|f\ , h)$, which represents the user's response to the history of system hypotheses and users feedbacks. This distribution considers the user behaviour and stands for the user model $\mathcal{M}_\mathcal{U}$ and can also be defined under the IPR framework considering now the user point of view. However, feedback f produced by the user in the previous interaction is not corrupted by any noisy channel, such as an ASR system, before arriving to the user again. Thus, a deterministic decoding $d : \mathcal{F} \quad \mathcal{D}$ maps each user turn signal into its corresponding unique decoding $d = d(f\ )$ before arriving to the *user model*. Consequently the *best* decoded user feedback $\hat{d}$ is the one that maximizes the posterior $P_{\mathcal{M}_\mathcal{U}}(d|d\ , h)$

$$\hat{d} = \arg\max_{d \in \mathcal{D}} P(d|d\ , h) \quad \arg\max_{d \in \mathcal{D}} P_{\mathcal{M}_\mathcal{U}}(d|d\ , h) \quad (4)$$

where $\hat{d}$ is estimated using only the hypothesis produced by the system and the feedback produced by the user in the previous interaction step according to its *user model*. Equation (4) represents the way the user model decides the feedback to be produced at each interaction step. As in the case of the DM, alternative criteria could be also considered to simulate the user behaviour. In fact, many simulated user models can be found in the bibliography related to SDSs [17, 18, 19, 20].

3. DM and SU: Cooperative models

In this work the probability distributions associated to both, the DM and the SU are modeled through a graphical model consisting of sets of states representing (h, d) pairs. Some of the states of this model correspond to the DM and are labelled by $(d, h\ )$, being d the output of the Speech Understanding system given the user feedback f and h the system hypothesis at the previous interaction. The states corresponding to the user, are labelled by pairs $(h, d\ )$ where h is the system hypothesis and d is the deterministic decoding of the previous user feedback f . An example of such a model is illustrated in Figure1. The edges that connect a DM node (d_i, h_i) with different SU nodes $((h_{i+1}, d_i), (h_{i+1}, d_i), \ldots)$ represent the different possible hypothesis that can be chosen in a system turn and each of them has associated a probability $P(h|d, h\ )$. Additionally, each state need to be labelled by the values of all the relevant internal variables and the uncertainty rate when speech is considered, thus leading to an attributed model. This information is typically associated to domain-specific frame-and-slot templates used in dialogue systems. In the same way edges connecting a SU node with different DM nodes represent possible user feedbacks according to the SU probability distribution. In this work we evaluate the proposed models in a dialogue generation task on two very different corpora. We first describe the parameter estimation procedure for these models in Section 3.1. Then the procedure to generate new dialogues is presented in Section 3.2.

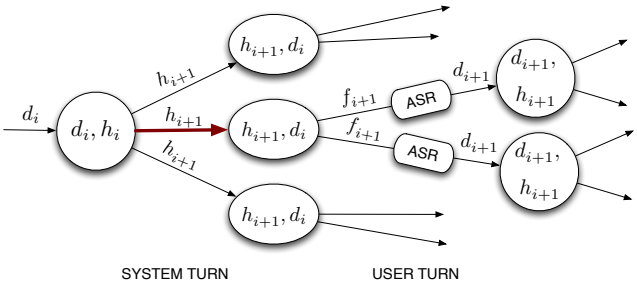

Figure 1: DM graphical model obtained from a training set.

3.1. Learning the models

The parameters of the model can be estimated in a three step learning procedure as follows:

- Get a dialogue corpus consisting of pairs of user and system turns. Then get an initial maximum likelihood estimation of the parameters of both DM and SU models.
- Define a DM strategy and several SU behaviours. Define also error recovery strategies. Run the system until desired dialogue goals are successfully achieved for different simulated user behaviours.
- Run the SDS with real users while using adaptive learning to obtain a DM adapted to real interaction feedbacks.

In this paper we are focussing on step one. Different DM strategies as well as SU behaviours will be compared in future works. Thus, a training corpus consisting of spoken dialogues need to be used. This corpus is used to get both the model topology, i.e. nodes and edges, and the probability distributions estimated through a maximum likelihood criterion.

The generalization issue is tackled by adopting a smoothing strategy for unseen events. Note that when the DM model is considered an alternative is needed for the cases in which the user provides a feedback that does not lead to one of the existing nodes in the DM model. In the same way, an alternative is also needed for the cases in which the system provides a hypothesis that does not lead to an existing node in the user model.

We have identified 3 different possible situations with the corresponding recovery strategies when considering the DM.

To an existing node: when the feedback provided by a new user turn is not one of the possible edges from our current state, but the destination node can be found in the graph, although it is not connected to our current state, it is going to be selected as the destination node. Thus, the new edge is added to the graph.

Non existing node but a node with the same user feedback: if the destination node is not present in our graph, we search for a node labelled with the same user feedback d in the graph. If we find more than one state we choose the nearest state to a closing state.

Non existing state and different user feedback: the most similar state to the one we are searching for is selected. To this end a similarity metric between two nodes need to be defined. The user perception about the system is strongly related to the similarity metric between the two nodes. When it is not very accurate the user may notice an incongruence in the system response.

The smoothing strategy for the user model was defined in an analogous way but considering the system hypothesis instead of the user feedback and exchanging user and system turns. Additionally, let us note that the user model has the ability to avoid loops in the graph. If the user model detects that the next selected action puts the dialogue in an already visited node, it applies the smoothing methods in order to change the action and exit from the loop.

3.2. Dialogue generation process

In order to generate new dialogues, the parameters of the models have to be estimated, as described in sec. 3.1, and a dialogue strategy and a user behaviour have to be defined. The DM will provide a hypothesis $\hat{h}$ according to eq. (3) in each system turn. In the same way the SU should provide a feedback $\hat{d}$ according to eq. (4) in each user turn. However, a different approach has been adopted in this work, in order to obtain a higher number of different dialogues and a less predictable behaviour. Specifically the user feedback is randomly chosen among all the possible edges that outs from the current user node. Thus, eq (4) is rewritten as eq (5), where $\mathcal{D}$ is the space of decoded feedback signals for which $P_{\mathcal{M}_\mathcal{U}}(d|d\ , h) = 0$.

$$\hat{d} \quad \text{random}_{d\ \in\ \mathcal{D}}\ P_{\mathcal{M}_\mathcal{U}}(d|d\ , h) \tag{5}$$

In this way, new dialogues are obtained using DM and SU models in a cooperative way. In a system turn ((d_i, h_i) state in Fig. 1) a hypothesis is chosen among all the possibilities, h_{i+1} (red arrow in Fig. 1) according to the dialogue strategy and a (h_{i+1}, d_i) state is reached. Then, in the user turn, the user should provide a feedback f_{i+1}. This feedback is decoded as d_{i+1} by an ASR system and then provided to the DM leading to a new system turn.

4. Experiments

Two series of experiments have been carried out by training the proposed models from two very different corpora: Dihana [21] and Let's Go [2].

Dihana is a set of spoken dialogues in Spanish, providing information related to the Spanish railway system. The corpus has been acquired by using the Wizard of Oz (WoZ) technique. In order to obtain more realistic dialogues, the speakers had to reach a certain goal in each dialogue of the 3 predetermined scenarios: get one way schedule, get one way price and schedule, get roundtrip price and schedule.

This corpus has been annotated in terms of Dialogue Acts (DAs), according to an adapted version of the Interchange Format defined in the C-STAR project [22]. Three hierarchical levels are present in the Dihana dialogue acts labelling scheme [23]. Next example shows a segment of the corpus labelled with this scheme:

```
I would like to know the fare on next Monday.
(U:Question:Price:Day)
```

Where "U" indicates that it is a user turn, "Question" is the first level that represents the action performed in the segment, "Price" the second level as the user is asking about the fare and "Day" is the third level because it is the additional information provided by the user.

Let's Go is a set of spoken dialogues in English in the bus information domain. Let's Go system has been developed by Carnegie Mellon University (CMU) over the Olympus-Ravenclaw framework [5]. It provides schedules and route information about the city of Pittsburgh's bus service to the general public. In this work we use a set of dialogues collected by the Let's Go Bus Information system in about two months, from March 2005 to April 2005.

In order to have the Let's Go corpus labelled in terms of Dialogue Acts, we have collected the information associated to each system turn from the log files of the Ravenclaw DM, whereas the information associated to the user turn was collected from the output of the Phoenix semantic decoder. The following example shows a dialogue labelled in this way:

```
(S:request:query.arrival_place.name)
Where are you going?
(U:PlaceInformation[SinglePlace][...])
SQUIRREL HILL
```

The main features of Dihana and Let's Go corpora used to carry out the experimental layout are summarized in Table 1: the size of both corpora in terms of number of dialogues, speakers and turns; the number of Dialogues Acts as a result of the labeling procedure; and the number of attributes representing the values of the variables that define the task.

Then, a separated maximum likelihood estimation of both models has been performed: the system model probability distribution $P_{\mathcal{M}_\mathcal{S}}(h|h\ , d)$ defined in Section 2 and the user model probability distribution $P_{\mathcal{M}_\mathcal{U}}(d|d\ , h)$ defined in Subsection 2.1. To this end both corpora were split into two subsets to train the DM and the SU respectively. Each node associated to a system turn is determined by the following variables: a DA label associated to the previous system action (h_i in Fig. 1), a DA label associated to the previous user feedback (d_i in Fig. 1) and finally, the list of attributes that has been already provided, with their values. On the other hand, the nodes associated to the user turns are determined by the following variables: a DA label associated to the previous user feedback (d_i in Fig. 1), a

	Dihana	**Let's Go**
Dialogues	900	1840
Speakers	225	1840
System Turns	9133	28141
User Turns	6280	28071
System Dialogue Acts	27	49
User Dialogue Acts	45	138
Attributes	13	14

Table 1: Main features of Dihana and Let's Go corpora.

	Dihana		Let's Go	
	Nodes	**Edges**	**Nodes**	**Edges**
System	7,466	5,049	13,634	9,906
User	7,387	4,289	13,648	10,199

Table 2: Sizes of the DM and SU models which have been trained with two subsets of Dihana and Let's Go corpora.

DA label associated to the system action (h_{i+1} in Fig. 1) and finally, the list of attributes that has been already provided, with their values. As a consequence, state labels are strings made up of the DAs and the attributes seen up to this point in the dialogue. Recovery strategies defined in Section 3.1 as smoothing techniques consider the most similar node to a given one. Thus the string edit distance has been used as a metric of similarity between two nodes.

The size of system and user models obtained for both tasks are summarized in Table 2. This table shows that the graph has an affordable size for vocabularies associated to restricted domain tasks, which are the most frequent in SDS applications. A more advanced model should also consider the confidence measure of each attribute, specifying if it has been seen with high or low confidence; in such a case, the state space would grow in direct proportion to the number of attributes seen in the corpus with their confidence measure. Then, using the interaction of the DM and SU models new dialogues were obtained. To this end the DM strategy and SU behaviour defined in Sec. 3.2 as well as the smoothing described in Sec. 3.1 were considered. A total of 100 new dialogues were obtained for each task. These dialogues were compared to a set of 100 dialogues randomly chosen from each corpus. The evaluation was carried out in terms of the metrics described below:

Task Completion (TC). Measures the success of the system in providing the user with the information requested [24].

Appropriate Utterance (AU). An utterance is considered appropriate when it provides the user the required information, when it asks for additional information which is essential to respond to the user's request or when it is dealing with a repair strategy. AU evaluates whether the DM provides a coherent response at each turn according to its input (output of the ASR).

Average Dialogue Length (ADL). The average number of turns in a dialogue.

TC, AU and ADL were computed for both the set of dialogues extracted from each corpus and the set of dialogues obtained through the DM and SU interaction. Table 3 shows the results of this evaluation. The number between parenthesis is the value computed when only the dialogues ending in a final state were considered for Let's Go task.

Table 3 shows that AU values are really similar for both the generated dialogues and the dialogues extracted from Dihana and Let's Go corpora. Thus, the proposed models have demonstrated their ability to replicate the original WoZ strategy as well as the Ravenclaw one.

	Dihana		Let's Go	
	WoZ	**Generated**	**Ravenclaw**	**Generated**
TC(%)	100	93	51.96	92 (77)
AU(%)	99	97	95.53	95.79
ADL	15.2	15	40.91	54 (46.5)

Table 3: TC, AU and DL over the generated dialogues and over a random subset of the original ones obtained through the WoZ and Ravenclaw DM. For Let's Go corpus, the number between parenthesis is the value of the metric computed when only the dialogues ending in a final state were considered.

	Dihana			Let's Go		
	Total	**User**	**System**	**Total**	**User**	**System**
NT	1496	698	798	5,506	2,703	2,803
NTS	515	262	253	1,221	686	535
SR	34.4	37.5	31.7	22.2	25.4	19.1

Table 4: (NT) Number of turns, (NTS) Number of Turns generated through a Smoothed edge and Smoothing Rate (SR) representing the percentage of turns obtained through smoothing techniques for system and user turns.

The TC value is very high and similar to the one obtained when the dialogue is guided by the WoZ for the Dihana task. Table 3 also shows a TC improvement of 40% for Let's Go dialogues generated by the proposed models. This is due to the fact that generated dialogues do not take into account the recognition error whereas corpus dialogues included ASR errors. We have computed the number of turns in Let's Go corpus where the recognized string is strongly different from the user utterance, not allowing the system to respond accurately because the meaning of the sentence was misunderstood. We have observed that the 45.56% of the turns in the corpus were wrongly recognized and interpreted by the system.

Finally, Table 4 shows the percentage of turns generated through smoothed edges of the DM and SU models. This table reveals a high use of smoothed edges that underlines the importance of considering an appropriate smoothing strategy, i.e. the recovery strategy. Notice that the smoothing percentage is lower in Let's Go showing a higher model coverage probably due to the higher number of training dialogues available. These experiments show that the proposed DM and SU models interact to get coherent dialogues that provides the user with the requested information.

5. Conclusions & Future Work

We have proposed an innovative approach to model a SDS under the IPR framework. In this work this formulation has been applied to define graphical models that deal with both a DM and a SU, and then put in practice to generate dialogues. To this end the estimation of the model parameters as well as the smoothing technique have been presented. These models were evaluated in a dialogue generation task on two very different corpora: Dihana corpus consisting of Spanish spoken dialogues acquired using the Wizard of Oz technique and Let's Go corpus consisting of spoken dialogues in English between real users and the Ravenclaw DM developed by CMU. The results obtained show that original and simulated dialogues exhibited very similar behaviours, thus demonstrating the learning capacity of the proposed models in both a controlled Wizard of Oz task and with a SDS that interacts with real users. The results obtained for the two tasks are equally promising and they allow to consider these models as an alternative to deal with SDSs. Future work includes the definition of more sophisticated DM strategies and SU behaviours and their evaluation in a complete SDS.

Acknowledgements

Work supported by Spanish Ministry of Science under TIN2011-28169-C05-04 and by Basque Government under grant IT685-13.

6. References

[1] S. Seneff and J. Polifroni, "Dialogue management in the mercury flight reservation system," in *Proceedings of the 2000 ANLP/NAACL Workshop on Conversational systems - Volume 3*, Stroudsburg, PA, USA, 2000, pp. 11–16.

[2] A. Raux, B. Langner, D. Bohus, A. W. Black, and M. Eskenazi, "Lets go public! taking a spoken dialog system to the real world," in *Proc. of Interspeech*, 2005.

[3] C. Lee, S. Jung, J. Eun, M. Jeong, and G. G. Lee, "A situation-based dialogue management using dialogue examples," in *Proceedings of ICASSP*, 2006, pp. 69–72.

[4] S. Varges, S. Quarteroni, G. Riccardi, A. V. Ivanov, and P. Roberti, "Combining pomdps trained with user simulations and rule-based dialogue management in a spoken dialogue system," in *Proceedings of the ACL-IJCNLP 2009 Software Demonstrations*, ser. ACLDemos '09, 2009, pp. 41–44.

[5] D. Bohus and A. I. Rudnicky, "The ravenclaw dialog management framework: Architecture and systems," *Computer Speech and Language*, vol. 23, pp. 332–361, 2009.

[6] A. Raux, D. Bohus, B. Langner, A. W. Black, and M. Eskenazi, "Doing research on a deployed spoken dialogue system: One year of lets go! experience," in *Proc. Interspeech*, 2006, pp. 65–68.

[7] E. Levin and R. Pieraccini, "A stochastic model of computer-human interaction for learning dialogue strategies," in *Proceedings of EUROSPEECH*, Rhodes, Greece, 1997, pp. 1883–1886.

[8] S. Young, "Probabilistic methods in spoken dialogue systems," *Philosophical Trans of the Royal Society*, vol. 1769(358), pp. 1389–1402, 2000.

[9] J. D. Williams and S. Young, "Partially observable markov decision processes for spoken dialog systems," *Computer Speech and Language*, vol. 21, pp. 393–422, 2007.

[10] J. Williams, "Incremental partiten recombination for efficient tracking of multiple dialog states," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Dallas, USA, 2010.

[11] S. Lee and M. Eskenazi, "Pomdp-based let's go system for spoken dialog challenge," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*, Dec., pp. 61–66.

[12] B. T. S. Young, M. Gasic and J. Williams, "Pomdp-based statistical spoken dialogue systems: a review," *Proc IEEE*, vol. to appear, 2013.

[13] A. H. Toselli, E. Vidal, and F. Casacuberta, Eds., *Multimodal Interactive Pattern Recognition and Applications*. Springer-Verlag, 2011.

[14] M. I. Torres, J. M. Benedí, R. Justo, and F. Ghigi, "Modeling spoken dialog systems under the interactive pattern recognition framework," in *SSPR&SPR. Lecture Notes on Computer Science*, 2012, pp. 519–528.

[15] M. Hajdinjak and france Mihleič, "The paradise evaluation framework: Issues and findings," *Computational Linguistics*, vol. 32, no. 2, pp. 263–272, 2006.

[16] E. Levin, R. Pieraccini, and W. Eckert, "A stochastic model of human-machine interaction for learning dialog strategies," *IEEE Transaction on Speech and Audio Processing*, vol. 8(1), pp. 11–23, 2000.

[17] D. Griol, Z. Callejas, and R. López-Cózar, "A comparison between dialog corpora acquired with real and simulated users," in *Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, 2009, pp. 326–332.

[18] K. Scheffler and S. Young, "Corpus-based dialogue simulation for automatic strategy learning and evaluation," 2001.

[19] S. Quarteroni, M. González, G. Riccardi, and S. Varges, "Combining user intention and error modeling for statistical dialog simulators," in *Proc. INTERSPEECH*, 2010.

[20] S. Lee and M. Eskenazi, "An unsupervised approach to user simulation: toward self-improving dialog systems," in *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, Seoul, Korea, July 2012, pp. 50–59.

[21] J. M. Benedí, E. Lleida, A. Varona, M. J. Castro, I. Galiano, R. Justo, I. López, and A. Miguel, "Design and acquisition of a telephone spontaneous speech dialogue corpus in spanish: DIHANA," in *Fifth LREC*, Genova, Italy, 2006, pp. 1636–1639.

[22] A. Lavie, L. Levin, P. Zhan, M. Taboada, D. Gates, M. Lapata, C. Clark, M. Broadhead, and A. Waibel, "Expanding the domain of a multi-lingual speech-to-speech translation system," in *Proceedings of the Workshop on Spoken Language Translation. ACL/EACL*, 1997.

[23] N. Alcácer, J. M. Benedí, F. Blat, R. Granell, C. D. Martnez, and F. Torres, "Acquisition and labelling of a spontaneous speech dialogue corpus," in *SPECOM 2005. Patras, Greece*, 2005, pp. 583–586.

[24] M. Danieli and E. Gerbino, "Metrics for evaluating dialogue strategies in a spoken language system," in *Proceedings of the 1995 AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation*, vol. 16, 1995, pp. 34–39.