HUMAN-COMPUTER DIALOGUE SIMULATION USING HIDDEN MARKOV MODELS

Heriberto Cuayáhuitl[1]*, Steve Renals*[1]*, Oliver Lemon*[2] *and Hiroshi Shimodaira*[1]

CSTR[1], HCRC[2], School of Informatics, University of Edinburgh
2 Buccleuch Place, Edinburgh, UK, EH8 9LW
{h.cuayahuitl, s.renals, olemon, h.shimodaira}@ed.ac.uk

ABSTRACT

This paper presents a probabilistic method to simulate task-oriented human-computer dialogues at the intention level, that may be used to improve or to evaluate the performance of spoken dialogue systems. Our method uses a network of Hidden Markov Models (HMMs) to predict system and user intentions, where a "language model" predicts sequences of goals and the component HMMs predict sequences of intentions. We compare standard HMMs, Input HMMs and Input-Output HMMs in an effort to better predict sequences of intentions. In addition, we propose a dialogue similarity measure to evaluate the realism of the simulated dialogues. We performed experiments using the DARPA Communicator corpora and report results with three different metrics: dialogue length, dialogue similarity and precision-recall.

1. INTRODUCTION

The task of human-computer dialogue simulation consists of generating artificial conversations between a spoken dialogue system and a user. The communication in real spoken dialogue systems is achieved at several levels: speech, words and intentions (analogous to dialogue acts). Training optimal dialogue strategies usually requires many dialogues to derive an optimal policy and online learning from real conversations may be impractical. An alternative is to use simulated dialogues. For dialogue modelling, simulation at the intention level is the most convenient, since the effects of recognition and understanding errors can be modelled and the intricacies of natural language generation can be avoided [1].

Several research efforts have been undertaken in this area for human-computer conversations, including rule-based [2]-[4] and corpus-based approaches [5]-[9][12]. Most of the investigations are intention based [5]-[10], and some use the speech and word levels [2][4], depending on the purposes of the simulated dialogues. All the investigations simulate user behaviour and some of them model speech recognition errors in order to corrupt users responses [2][3][8]-[10]. Domains vary from restaurants [2], air travel information [5]-[7][11][12], banking [8], cinema [9], computer purchasing [10], and fast food [4]. Finally, a few investigations attempt to evaluate the simulated user behaviour [8][9][11][12] using simple statistical metrics. These investigations mostly simulate user behaviour in order to interact with an existing spoken dialogue system. However, no corpus-based efforts have been undertaken to simulate both system and user behaviour.

This paper presents a method that addresses the following question: *How to expand a small corpus of dialogue data with more varied simulated conversations?* Our method learns system and user behaviour based on a network of HMMs, where each HMM represents a goal in the conversation. In an effort to better predict real dialogues we compare three models with different dependencies in their structures. In addition, this paper presents a measure to evaluate the realism of the simulated dialogues through the comparison of HMMs trained with real and simulated dialogues. Some potential uses of the expanded corpus may be to learn optimal dialogue strategies and to evaluate spoken dialogue systems in early stages of development.

2. PROBABILISTIC DIALOGUE SIMULATION

This section describes a probabilistic human-computer dialogue simulation method that models both system and user behaviour at the intention level (see figure 1). A set of real dialogues (the training set) is required in order to acquire knowledge and train the system and user models, which are used to make them interact together using intentions in order to generate simulated dialogues. The system model is a probabilistic dialogue manager that controls the flow of the conversation, and the user model is a set of conditional probabilities that describe user behaviour. Finally, the simulated dialogues and another set of real dialogues (the test set) are used to evaluate the realism of such simulated dialogues.

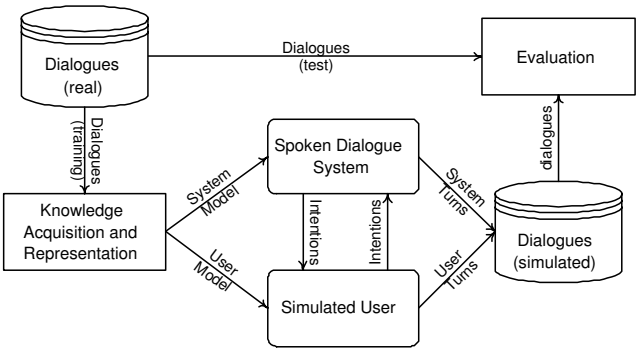

Fig. 1. A high-level diagram of the proposed human-computer dialogue simulator.

This research was mainly sponsored by PROMEP, part of the Mexican Ministry of Education (http://promep.sep.gob.mx); and partially sponsored by the Autonomous University of Tlaxcala (www.uatx.mx) and the European Community FP6 project "TALK" (www.talk-project.org).

2.1. The System Model

The task of the system model is to generate a sequence of system turns including system intentions, allowing user responses between turns. Due to the fact that conversations may have many system turns and that some turns are reused during the conversations, we decided to divide the conversation into goals, which are subsequences of system turns within the same topic. Therefore, our system model consists of multiple Hidden Markov Models (HMMs) connected by a bigram language model, where each HMM in the network represents a dialogue goal (see figure 2a). The task of the bigram language model is to predict the goal sequence within a dialogue by the conditional probability of the preceding goal $P(g_n|g_{n-1})$ given the set of goals $G = \{g_1, g_2, ..., g_N\}$. The language model is parameterized as $\Lambda = (\sigma, \delta)$, where σ is the initial distribution and δ the transition distribution. The conversation within a goal is modelled by an ergodic HMM with visible states. The notation $\lambda = (A, B, \pi)$ is used to indicate the complete parameter set of a standard HMM and its characterization is as follows [13]:

- N, the number of states within a goal plus a final state. We assume that any goal can be modelled as a set of visible states $S = \{S_1, S_2, ..., S_N\}$ representing system turns, the state at time t is referred as q_t and the final state is referred as q_N.
- M, the number of observed symbols, represented as a set of system intentions $V = \{v_1, v_2, ..., v_M\}$, the symbol observed at time t is referred as c_t.
- The discrete random variable A describes the flow of system turns by $P(q_{t+1}|q_t)$.
- The discrete random variable B describes the system intentions generated in each state by $P(c_t|q_t)$.
- The initial state distribution $\pi = P(q_0)$ represents the start of the conversation within a goal.

Standard HMMs consider state transitions (system turns) and observations (system intentions) independent of user responses (see figure 2b), meaning that the control flow of the conversation does not take into account the previous user responses. This fact motivated the use of models with more dependencies in their structure. Therefore, we use Input Hidden Markov Models (IHMMs) and Input-Output Hidden Markov Models (IOHMMs), which are extensions of the standard HMMs [14], see figures 2c and 2d. IHMMs condition the next state transition q_{t+1} on the current state q_t and the current user response u_t, the state transition probability is rewritten as $P(q_{t+1}|q_t, u_t)$. IOHMMs extend IHMMs by conditioning the current observation c_t on the current state q_t and the previous user response u_{t-1}, the observation symbol probability distribution is rewritten as $P(c_t|q_t, u_{t-1})$.

2.2. The User Model

The task of the user model is to interact with the system model by providing answers to system intentions. Our user model is based on the assumption that a user response is conditional only on the previous system response [5]. The observed symbols are represented by the set of user intentions $H = \{h_1, h_2, ..., h_L\}$, where L is the number of intentions and the intention at time t is referred as u_t. Thus, the discrete random variable U describes the user intentions generated in each state by $P(u_t|q_t, c_t)$. Figure 3 illustrates the structure of an IOHMM including user responses.

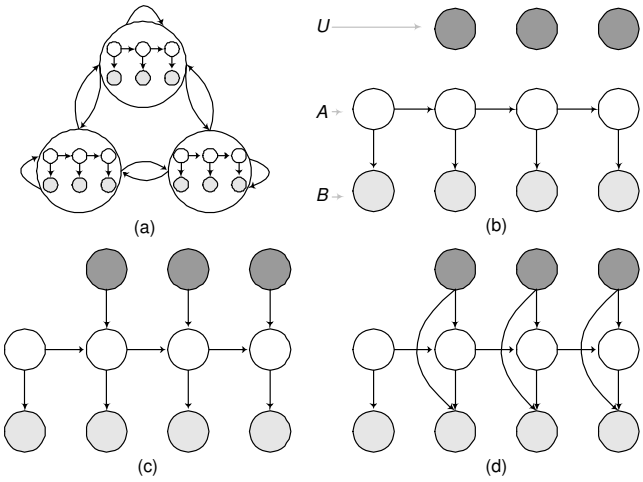

Fig. 2. HMM-based system models. (a) a language model defining a network of Hidden Markov Models (HMMs), (b) a standard HMM, (c) an Input HMM (IHMM) and (d) an Input-Output HMM (IOHMM). The empty circles represent visible states, the lightly shaded circles represent observations and the dark shaded circles represent user responses.

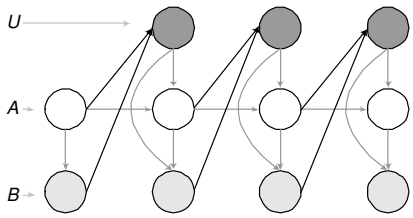

Fig. 3. The IOHMM including user responses (dark shaded circles), the black arrows correspond to the user model.

2.3. The Simulation Algorithm

The language model and HMMs are used as a generator in order to simulate task-oriented human-computer dialogues at the intention level. A simplified version of the dialogue simulation algorithm using standard HMMs is shown in figure 4. The function *DialogueSimulator* generates sequences of goals using the language model Λ, choosing initial goals from σ and goal transitions from δ, until reaching the final goal g_N. For each goal, the function *SimulateHMM* is invoked with the corresponding model λ, which generates a sequence of system intentions c_t and user intentions u_t, until reaching the final state q_N. The probability distributions from lines 18 and 21 may be replaced with the ones specified by IHMMs or IOHMMs. The algorithm assumes that the system starts the conversation and the user ends it.

3. DIALOGUE SIMILARITY

This section describes a measure to evaluate the realism of simulated dialogues. The motivation for proposing another measure is due to the fact that previous measures are either very general (such as dialogue length [8]) or very strict (such as precision-recall [11], which highly penalizes unseen dialogues). Therefore, in an at-

01. **function** DialogueSimulator()
02. load parameters of the language model Λ
03. *current_goal* $\leftarrow$ random goal from σ

04. **while** *current_goal* != g_N **do**
05. $\lambda \leftarrow$ parameters of the HMM given *current_goal*
06. SimulateHMM(λ)
07. *current_goal* $\leftarrow$ random goal from δ
08. **end**
09. **end**

10. **function** SimulateHMM(λ)
11. $t \leftarrow 0$
12. $q_t \leftarrow$ random system turn from π
13. $c_t \leftarrow$ random system intention from $P(c_t|q_t)$

14. **loop**
15. print c_t
16. $u_t \leftarrow$ random user intention from $P(u_t|q_t, c_t)$
17. print u_t
18. $q_t \leftarrow$ random system turn from $P(q_{t+1}|q_t)$
19. **if** $q_t = q_N$ **then return**
20. **else** $t \leftarrow t + 1$
21. $c_t \leftarrow$ random system intention from $P(c_t|q_t)$
22. **end**
23. **end**

Fig. 4. The dialogue simulation algorithm.

tempt to address the deficiencies of the previous measures we propose a dialogue similarity measure. The purpose of this measure is to evaluate the similarity between two sets of dialogues. For our purposes, we compare a corpus of real dialogues against a corpus of simulated dialogues[1], training a set of standard HMMs (one per dialogue goal) for each corpus. This measure computes the normalized distance between HMMs trained from each corpus, where γ_r represents a set of HMMs trained with real dialogues and γ_s represents another set of HMMs trained with simulated dialogues. The similarity is the distance between γ_r and γ_s given by equation 1. Notice that this measure can evaluate the system model (including the variables q and c), the user model (including the variable u) or both (including all variables). This measure attempts to provide an indication of how far all the simulated dialogues are from the real dialogues.

$$D^*(\gamma_r, \gamma_s) = \frac{1}{L}\sum_{l=1}^{L}\frac{1}{N}\sum_{i=1}^{N}\frac{1}{M_i}\sum_{j=1}^{M_i} D(\omega_j; \lambda_{r_i}, \omega_j; \lambda_{s_i}), \quad (1)$$

where L is the number of variables to compare, N is the number of HMMs (one per goal), M is the number of probability distributions in the model λ_i, ω is the variable (e.g., q, c, u), and D is a distance between HMMs expressed as

$$D(p, q) = \frac{D_{KL}(p \parallel q) + D_{KL}(q \parallel p)}{2}, \quad (2)$$

and D_{KL} is the Kullback-Leibler divergence expressed as

$$D_{KL}(p \parallel q) = \sum_i p_i \, log_2 \left(\frac{p_i}{q_i}\right). \quad (3)$$

[1] Under the assumption that the more similarity the more realism.

4. EXPERIMENTAL DESIGN

4.1. Training the System and User Model

Our experiments use the DARPA Communicator corpora 2001, which is annotated using the DATE annotation scheme [15]. These corpora (available from the LDC), consists of task-oriented human-computer dialogues in the domain of travel information. The DATE scheme annotates dialogues using dialogue acts, which characterize behaviour of human-computer dialogues. Both system turns and user turns are annotated, with a focus on system turns, assuming that system behaviour is correlated to user behaviour. As a consequence, system turns are annotated with dialogue acts, whilst user turns provide the ASR and user transcriptions at the word level embedding semantic tags. Using this data we trained our models using the following five steps:

1. **Dialogue segmentation**, where each segment corresponds to a goal, these segments are application dependent. Figure 5 shows the goal delimiters (dialogue acts) of the systems used in our experiments. This step was used to train the language models, the rest of the steps were used to train the HMMs.
2. **Classification of system turns into states for the HMMs**. Briefly, the system turns with speech acts *request_info*, *offer*, and *acknowledgement* were classified as states, using such order in order to avoid duplicated states. System turns without any of these speech acts were classified according to their most recent state.
3. **Classification of system turns into intentions**. Due to the fact that system turns have many combinations of dialogue acts, we collapsed them into the set of system intentions $V = \{$*start, apology, instruction, confirmation*$\}$. Briefly, and using the following order, the system turns with speech acts *explicit_confirm* were classified as *confirmation*; the system turns with speech acts *apology* as *apology*; the system turns with speech acts *request_info*, *offer*, and *acknowledgement* as *start*; the system turns with speech acts *instruction* as *instruction*; and any other system turn as *start*.
4. **Classification of user turns into intentions**. As we are interested in intention-based dialogues, information from transcriptions was used in order to classify user turns into the set of user intentions H = {*oov, command, yes, no, CITY, DATE_TIME, RENTAL, CAR, AIRLINE, HOTEL, AIRPORT, NUMBER, CITY CITY, DATE_TIME DATE_TIME, CITY DATE_TIME, AIRLINE DATE_TIME, AIRLINE NUMBER, CITY CITY DATE_TIME*}. The items in capital are the semantic tags that occur in most of the systems. The use of more than one semantic tag allows user initiative. The full set of user intentions H was used to provide user responses and to train the state transitions in IHMMs and IOHMMs, but for training observations in IOHMMs we collapsed the semantic tags into the intention *iv*, in an effort to reduce the data sparsity problem. Finally, subsets of H (system vocabulary) where allowed in each HMM, according to the user intentions observed in the data.
5. **Smoothing of intentions in order to consider unseen entries**. Due to the fact that many intentions may not have occurred in the data, the probability distributions of the HMMs (system turns, system intentions and user intentions) were smoothed using Backoff estimation with Witten-Bell discounting [16].

SYSTEM	GOAL DELIMITERS	STATES PER GOAL	USER INITIATIVE	# DIALOGUES	
				TRAIN	TEST
BBN	g0:about_task|request_info|top_level_trip g1: about_task|request_info|hotel g2: about_task|request_info|rental g3:about_task|request_info|return_date g4: about_task|request_info|flight	10 9 8 9 3	0.43	105	56
CMU	g0:about_task|request_info|dest_city g1:about_task|request_info|return_date g2: about_task|request_info|continue_trip g3: about_task|request_info|hotel g4: about_task|request_info|rental g5: about_task|request_info|flight	7 7 3 7 5 4	0.42	45	31
COL	g0:about_task|request_info|top_level_trip g1:about_task|request_info|return_date g2: about_task|request_info|continue_trip g3: about_task|request_info|rental g4: about_task|request_info|hotel g5: about_task|request_info|flight	9 8 5 10 7 8	0.46	126	81
IBM	g0: about_task|request_info|top_level_trip g1: about_task|request_info|continue_trip g2:about_task|offer|flight	9 8 9	0.67	134	85
LUC	g0:about_task|request_info|top_level_trip g1: about_task|request_info|continue_trip g2:about_task|request_info|return_date g3: about_task|request_info|rental g4: about_task|request_info|hotel	11 10 8 10 10	0.50	103	74
MIT	g0: about_task|request_info|top_level_trip g1: about_task|request_info|return_date g2: about_task|request_info|price	13 10 10	0.94	116	94

Fig. 5. Information extracted from the Communicator data. User initiative is the ratio between number of semantic tags and number of utterances (from user transcriptions). These corpora is a subset of the original dialogue data.

4.2. Evaluation Metrics

Evaluating simulated dialogues is a difficult process due to the fact that we do not know in advance if the simulated dialogues would occur in real environments. Nevertheless, we evaluate our method using the following metrics that compare two sets of dialogues. For our purposes we are mainly interested in comparing real dialogues (test set) against simulated dialogues.

- **Dialogue Length**: This measure computes the average number of turns per dialogue, giving a rough indication of agreement between two sets of dialogues.
- **Precision-Recall**: This measure evaluates how well a model can predict training and test data, but it highly penalizes the simulated dialogues that did not occur in the real data. This measure is illustrated in figure 6, where recall is given by $R_{train} = (A \cap C)/A$ or $R_{test} = (B \cap C)/B$, and precision is given by $P_{train} = (A \cap C)/C$ or $P_{test} = (B \cap C)/C$. An average of recall and precision is given by $F = 2PR/(P + R)$ [16].
- **Dialogue Similarity**: This proposed measure computes the normalized distance of standard HMMs between two sets of dialogues, penalizing unseen behaviour, but taking into account seen and unseen dialogues (see section 3).

In this paper our evaluation focuses on the HMM-based system models, but such measures can also be used to evaluate the user model or both. In the case of dialogue length we only consider system turns. In the case of precision-recall we consider fragments (one per goal) compounded by state plus system intention. Finally, in the case of dialogue similarity we only consider system turns (states) and system intentions (observations), but other parameters might be incorporated such as user intentions.

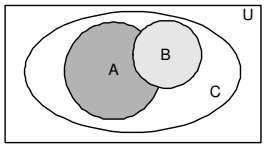

Fig. 6. Data sets used by the Precision-Recall measure (*A=real dialogues in the training set, B=real dialogues in the test set, C=simulated dialogues*), if the set C covers completely A and B this measure will mean realism in the simulated dialogues.

4.3. Experiments and Results

We trained the proposed models for six Communicator systems: BBN, CMU, COL, IBM, LUC, MIT. From the original data we filtered dialogues with missing annotations that impede to induce system intentions, the size of the corpora used for experiments is shown in figure 5. We performed experiments for each system in order to compare the proposed HMM models. In each comparison 10^4 simulated dialogues were generated.

Figures 7 and 8 illustrate results from closed and open tests using the three evaluation metrics: Dialogue Length (DL), Precision-Recall (PR) and Dialogue Similarity (DS). The bars in each plot represent: real dialogues (comparing the training and test sets), random dialogues (using same setup as standard HMMs but with flat probabilities), and HMM-based simulations (HMMs, IHMMs and IOHMMs). Ideally, we would like our models to behave similarly to the real dialogues; we assume that reaching similar scores as the real dialogues our simulations may be considered realistic (it is only an indication). From the results we can observe that random dialogues obtain the worst performance (meaning the they are strongly unrealistic), whilst the HMM-based models are better than random. From the PR (goals) results we can observe that the HMM-based models result in a similar performance due to the fact that they use the same language model. From the PR (intentions) results we can observe that the HMM-based models obtain similar performance as the real dialogues. Thus, PR is partially useful because it only tell us how much our models can predict training and test data, but penalizes the unseen dialogues. This fact raises the question *"What proportion of dialogues penalized by PR may occur in real environments?"*

In another side, from the DS measure we can observe that the HMM-based simulations are considerably distant from the real dialogues. This measure is promising due to the fact that it is strongly evaluating dialogue behavior in comparison to the other metrics. This fact raises the question *"How realistic might be the simulated dialogues if they obtain similar distance to the real ones?*" In the meantime, all measures agree that the random dialogues are significantly unrealistic, whilst our trained models generate dialogues closer the to real ones; this can be observed from the average results in figure 9. According to PR and DS we can observe that IHMMs and IOHMMs perform slightly closer to real dialogues, but still cannot be considered realistic. This suggests exploring more effective dependencies in the HMMs.

Finally, figure 10 illustrates a simulated dialogue based on the CMU simulated system with IHMMs, the left column uses intentions and the right column is an instantiation in natural language. Because of our method is purely probabilistic, some incoherencies may occur; for instance, the system offers a return flight without asking for a return date.

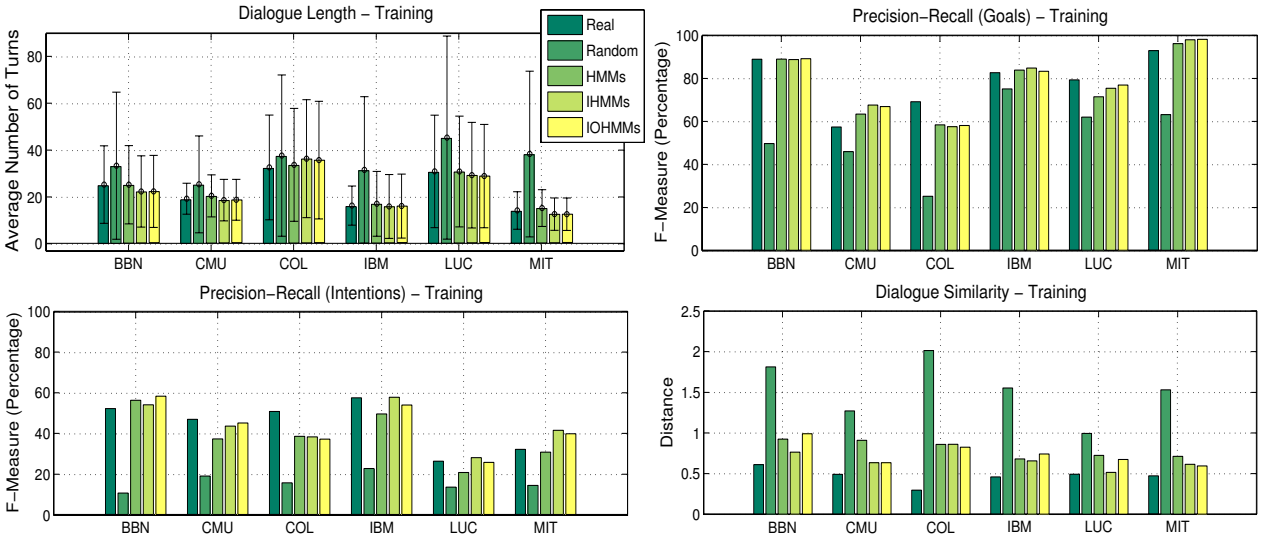

Fig. 7. Results using training data with three different metrics (the thin bars represent the standard deviation).

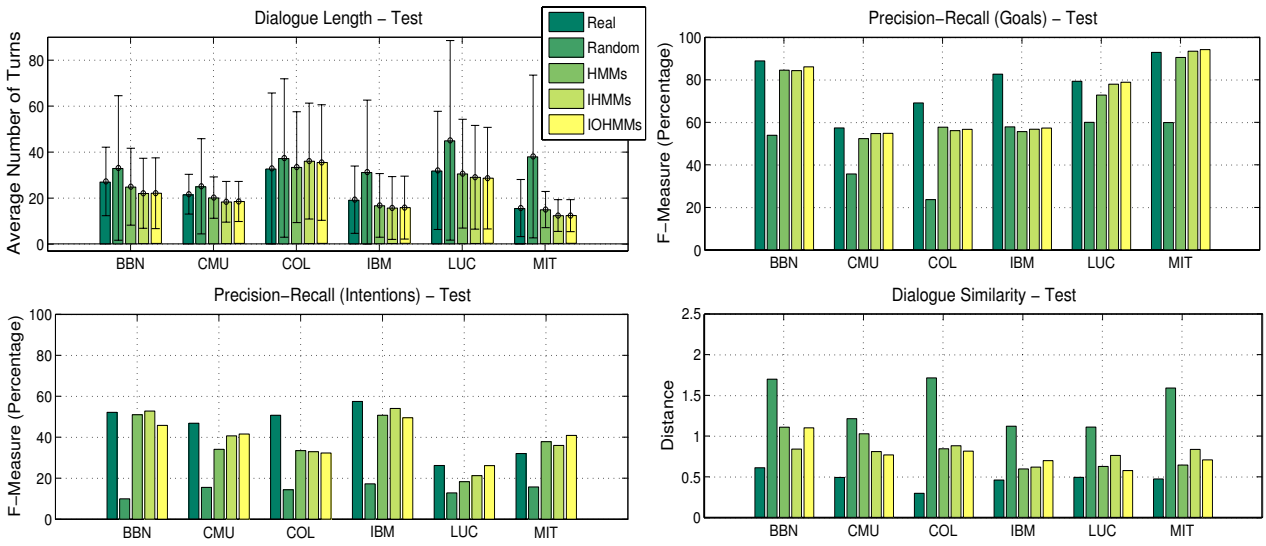

Fig. 8. Results using test data with three different metrics (Precision-Recall of goals use the same language model).

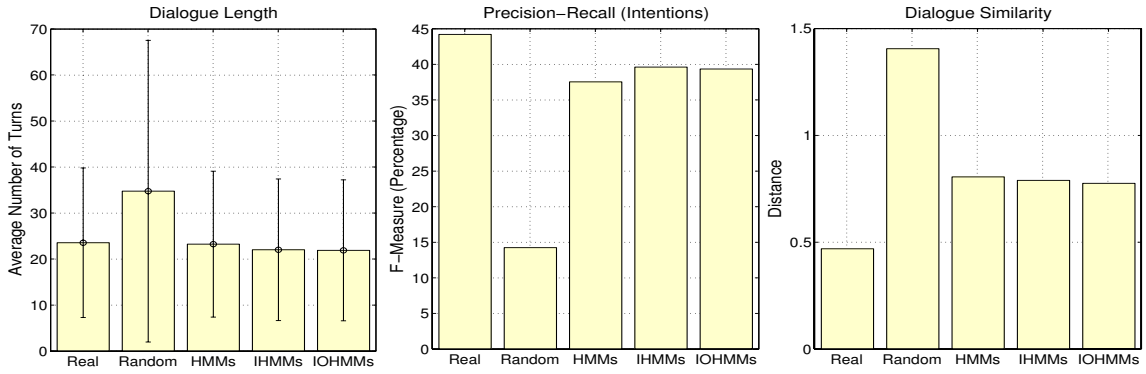

Fig. 9. Average test results from all Communicator systems (assumption: the closer to "Real" the better).

SYS about_task|request_info|dest_city	**SYS** where do you want to go?
USR CITY	**USR** Atlanta
SYS about_task|request_info|depart_arrive_date	**SYS** and what day would you be departing Boston?
USR oov	**USR** [cough]
SYS about_task|request_info|depart_arrive_time	**SYS** and what time would you like to depart Boston?
USR DATE_TIME	**USR** september twenty seventh early afternoon
SYS about_task|offer|flight	**SYS** [info] do you want to take that?
USR no	**USR** no
SYS about_task|offer|flight	**SYS** [info] do you want to take that?
USR DATE_TIME	**USR** morning
SYS about_task|offer|flight	**SYS** [info] do you want to take that?
USR yes	**USR** yes
SYS about_task|request_info|return_date	**SYS** will you return to Boston?
USR oov	**USR** [noise]
SYS about_communication|apology|meta_slu_reje	**SYS** i'm sorry. i didn't catch that.
USR yes	**USR** yes
SYS about_task|offer|flight	**SYS** [info] do you want to take that?
USR yes	**USR** yes
SYS about_task|request_info|hotel	**SYS** will you need a hotel in Atlanta?
USR yes	**USR** yes
SYS about_task|request_info|hotel_location	**SYS** would you like a hotel downtown or near the ...
USR HOTEL	**USR** downtown
SYS about_task|request_info|flight	**SYS** would you like me to summarize your trip?
USR yes	**USR** yes
SYS about_task|acknowledgement|rental_booking	**SYS** i've requested a rental car with no preference ...
USR command	**USR** repeat

Fig. 10. Fragment of a simulated dialogue.

5. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a corpus-based method to simulate task-oriented human-computer dialogues at the intention level using a network of HMMs connected by a bigram language model, where each HMM represents a dialogue goal. This method learnt a system model and a user model: the system model is a probabilistic dialogue manager that models the sequence of system intentions, and the user model consists of conditional probabilities of the possible user responses. We argue that our method is independent of application and annotation scheme. Due to the fact that in the proposed method all the possible system and user intentions may occur in each state, more exploratory dialogues may be generated than observed in the real data. We compared three models with different structures: HMMs, IHMMs and IOHMMs. Our experiments with the DARPA Communicator data reveal that the HMM-based models obtain very similar performance, clearly outperforming random dialogues, and are close to being considered realistic. We believe that Precision-Recall and Dialogue Similarity are potentially complementary metrics due to the fact that precision-recall penalizes the unseen dialogues, whilst dialogue similarity considers all the dialogues. This suggest that a combination of measures may better evaluate the realism of simulated dialogues, but there is no guarantee that these metrics are directly related to dialogue realism.

Immediate work in dialogue simulation follows two directions: 1) better evaluation measures and 2) improve the performance of our proposed method including: degrees of initiative in user responses, investigate the application of balanced number of goals and states, duration modelling, model system and user intentions according to the dialogue history (this should yield more coherent sequences of goals and intentions), model confidence levels, model different kinds of users, and explore richer dependecies in the models but avoiding the data sparsity problem. Future work consists in using the proposed method within the reinforcement learning framework to learn optimal dialogue strategies for large-scale spoken dialogue systems.

6. REFERENCES

[1] S. Young, "Probabilistic Methods in Spoken Dialogue Systems," in *Philosophical Transactions of the Royal Society*, (Series A) 358(1769), pp. 1389-1402, 2000.

[2] G. Chung, "Developing a Flexible Spoken Dialog System Using Simulation," in *ACL*, Barcelona, Spain, pp. 63-70, 2004.

[3] B. Lin and L. Lee, "Computer-Aided Analysis and Design for Spoken Dialogue Systems Based on Quantitative Simulations," in *Proc. of the IEEE, Transactions on Speech and Audio Processing*, 9:5, pp. 534-548, 2001.

[4] R. Lopez-Cozar, A. De la Torre, J.C. Segura, and A.J. Rubio, "Assessment of Dialogue Systems by Means of a New Simulation Technique," in *Speech Communication*, 40, pp. 387-407, 2003.

[5] W. Eckert, E. Levin and R. Pieraccini, "User Modeling for Spoken Dialogue System Evaluation," in *Proc. of IEEE ASRU Workshop*, Santa Barbara, Cal., USA, 1997.

[6] E. Levin, R. Pieraccini and W. Eckert, "A Stochastic Model of Computer-Human Interaction for Learning Dialog Strategies," in *Proc. of Eurospeech*, Rhodes, Greece, pp. 1883-1886, 1997.

[7] E. Levin, R. Pieraccini and W. Eckert, "A Stochastic Model of Human-Machine Interaction for Learning Dialog Strategies," in *Proc. of the IEEE, Transactions on Speech and Audio Processing*, 8:1, pp. 11-23, 2000.

[8] K. Scheffler and S. Young, "Probabilistic Simulation of Human-Machine Dialogues," in *Proc. of the IEEE ICASSP*, Istanbul, Turkey, pp. 1217-1220, 2000.

[9] K. Scheffler and S. Young "Corpus-Based Dialogue Simulation for Automatic Strategy Learning and Evaluation," in *Proc. of NAACL Workshop on Adaptation in Dialogue Systems*, Pittsburgh, USA, 2001.

[10] O. Pietquin and S. Renals, "ASR System Modeling for Automatic Evaluation and Optimization of Dialogue Systems," in *Proc. of the IEEE ICASSP*, Florida, USA, pp. 46-49, 2002.

[11] J. Schatzmann, K. Georgila and S. Young, "Quantitative Evaluation of User Simulation Techniques for Spoken Dialogue Systems," in *Proc. of Workshop on Discourse and Dialogue*, Lisbon, Portugal, 2005.

[12] K. Georgila, J. Henderson and O. Lemon, "Learning User Simulations for Information State Update Dialogue Systems," in *Proc. of Eurospeech*, Lisbon, Portugal, pp. 893-896, 2005.

[13] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," in *Proc. of the IEEE*, 77:2, pp. 257-286, February, 1989.

[14] Y. Bengio and P. Frasconi, "Input-Output HMMs for sequence processing," in *IEEE Trans. Neural Networks*, 7:5, pp. 1231-1249, September 1996.

[15] M. Walker and R. Passonneau, "DATE: A Dialogue Act Tagging Scheme for Evaluation of Spoken Dialogue Systems," in *Proc. of HLT*, San Diego, USA, pp. 1-8, 2001.

[16] D. Jurafsky and J. Martin, "*Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*," Prentice Hall, Upper Saddle River, 2000.