Combining POMDPs trained with User Simulations and Rule-based Dialogue Management in a Spoken Dialogue System

Sebastian Varges, Silvia Quarteroni, Giuseppe Riccardi, Alexei V. Ivanov, Pierluigi Roberti
Department of Information Engineering and Computer Science
University of Trento
38050 Povo di Trento, Italy
{varges|silviaq|riccardi|ivanov|roberti}@disi.unitn.it

Abstract

Over several years, we have developed an approach to spoken dialogue systems that includes rule-based and trainable dialogue managers, spoken language understanding and generation modules, and a comprehensive dialogue system architecture. We present a Reinforcement Learning-based dialogue system that goes beyond standard rule-based models and computes on-line decisions of the best dialogue moves. The key concept of this work is that we bridge the gap between manually written dialog models (e.g. rule-based) and adaptive computational models such as Partially Observable Markov Decision Processes (POMDP) based dialogue managers.

1 Reinforcement Learning-based Dialogue Management

In recent years, Machine Learning techniques, in particular Reinforcement Learning (RL), have been applied to the task of dialogue management (DM) (Levin et al., 2000; Williams and Young, 2006). A major motivation is to improve robustness in the face of uncertainty, for example due to speech recognition errors. A further motivation is to improve adaptivity w.r.t. different user behaviour and application/recognition environments. The Reinforcement Learning framework is attractive because it offers a statistical model representing the dynamics of the interaction between system and user. This is in contrast to the supervised learning approach of learning system behaviour based on a fixed corpus (Higashinaka et al., 2003). To explore the range of dialogue management strategies, a simulation environment is required that includes a simulated user (Schatzmann et al., 2006) if one wants to avoid the prohibitive cost of using human subjects.

We demonstrate the various parameters that influence the learnt dialogue management policy by using pre-trained policies (section 4). The application domain is a tourist information system for accommodation and events in the local area. The domain of the trained DMs is identical to that of a rule-based DM that was used by human users (section 2), allowing us to compare the two directly. The state of the POMDP keeps track of the SLU hypotheses in the form of domain concepts (10 in the application domain, e.g. main activity, star rating of hotels, dates etc.) and their values. These values may be abstracted into 'known/unknown,' for example, increasing the likelihood that the system re-visits a dialogue state which it can exploit. Representing the verification status of the concepts in the state, influences – in combination with the user model (section 1.2) and N best hypotheses – if the system learns to use clarification questions.

1.1 The exploration/exploitation trade-off in reinforcement learning

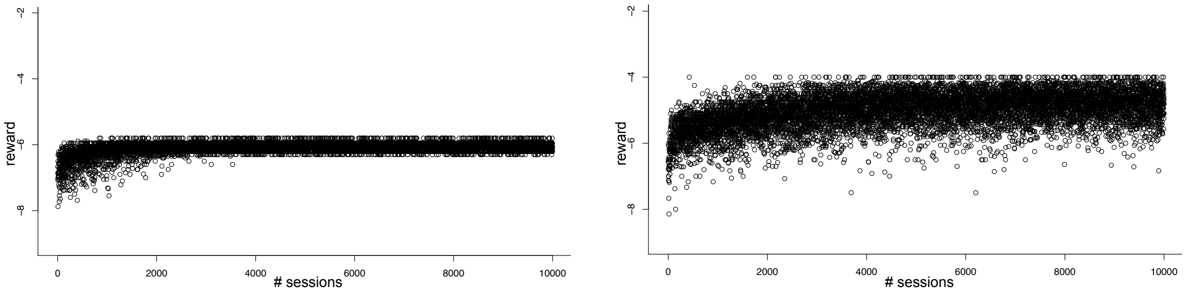
The RL-DM maintains a policy, an internal data structure that keeps track of the values (accumulated rewards) of past state-action pairs. The goal of the learner is to optimize the long-term reward by maximizing the 'Q-Value' $Q^{\pi}(s_t, a)$ of a policy π for taking action a at time t. The expected cumulative value V of a state s is defined recursively as $V^{\pi}(s_t) =$

$$\sum_a \pi(s_t, a) \sum_{s_{t+1}} P^a_{s_t,s_{t+1}} [R^a_{s_t,s_{t+1}} + \gamma V^{\pi}(s_{t+1})].$$

Since an analytic solution to finding an optimal value function is not possible for realistic dialogue scenarios, $V(s)$ is estimated by dialogue simulations.

To optimize Q and populate the policy with expected values, the learner needs to explore untried actions (system moves) to gain more experiences, and combine this with exploitation of the

Proceedings of the ACL-IJCNLP 2009 Software Demonstrations, pages 41–44,
Suntec, Singapore, 3 August 2009. ©2009 ACL and AFNLP



(a) 0% exploration, 100% exploitation: learner does not find optimal dialogue strategy

(b) 20% exploration, 80% exploitation: noticeable increase in reward, hitting upper bound

Figure 1: Exploration/exploitation trade-off

already known successful actions to also ensure high reward. In principle there is no distinction between training and testing. Learning in the RL-based dialogue manager is strongly dependent on the chosen exploration/exploitation trade-off. This is determined by the action selection policy, which for each system turn decides probabilistically (ϵ-greedy, softmax) if to exploit the currently known best action of the policy for the believed dialogue state, or to explore an untried action. Figure 1(a) shows, for a subdomain of the application domain, how the reward (expressed as minimizing costs) reaches an upper bound early during 10,000 simulated dialogue sessions (each dot represents the average of 10 rewards at a particular session number). Note that if the policy provides no matching state, the system can only explore, and thus a certain amount of exploration always takes place. In contrast, with exploration the system is able to find lower cost solutions (figure 1(b)).

1.2 User Simulation

In order to conduct thousands of simulated dialogues, the DM needs to deal with heterogeneous but plausible user input. For this purpose, we have designed a User Simulator (US) which bootstraps likely user behaviors starting from a small corpus of 74 in-domain dialogs, acquired using the rule-based version of the SDS (section 2). The task of the US is to simulate the output of the SLU module to the DM, hence providing it with a ranked list of SLU hypotheses.

A list of possible user goals is stored in a database table (section 3) using a frame/slot representation. For each simulated dialogue, one or more user goals are randomly selected. The User Simulator's task is to mimic a user wanting to perform such task(s). At each turn, the US mines the previous system dialog act to obtain the concepts required by the DM and obtains the corresponding values (if any) from the current user goal.

The output of the user model proper is passed to an error model that simulates the "noisy channel" recognition errors based on statistics from the dialogue corpus. These concern concept values as well as other dialogue phenomena such as noInput, noMatch and hangUp. If the latter phenomena occur, they are propagated to the DM directly; otherwise, the following US step is to attach plausible confidences to concept-value pairs, also based on the dialogue corpus. Finally, concept-value pairs are combined in an SLU hypothesis and, as in the regular SLU module, a cumulative utterance-level confidence is computed, determining the rank of each of the n hypotheses. The probability of a given concept-value observation at time $t+1$ given the system act at time t, named $a_{s,t}$, and the session user goal g_u, $P(o_{t+1}|a_{s,t}, g_u)$, is obtained by combining the error model and the user model:

$$P(o_{t+1}|a_{u,t+1}) \cdot P(a_{u,t+1}|a_{s,t}, g_u)$$

where $a_{u,t+1}$ is the true user action.

2 Rule-based Dialogue Management

A rule-based dialogue manager was developed as a meaningful comparison to the trained DM, to obtain training data from human-system interaction for the user simulator, and to understand the properties of the domain (Varges et al., 2008). Rule-based dialog management works in two stages: retrieving and preprocessing facts (tuples) taken from a dialogue state database (section 3), and inferencing over those facts to generate a system response. We distinguish between the 'context model' of the first phase – essentially allowing more recent values for a concept to override less recent ones – and the 'dialog move engine' (DME) of the second phase. In the second stage, acceptor rules match SLU results to dialogue context, for example perceived user concepts to open questions. This may result in the decision to verify the application parameter in question, and the action is verbalized by language generation rules. If the parameter is accepted, application dependent task rules determine the next parameter to be acquired, resulting in the generation of an appropriate request.

3 Data-centric System Architecture

All data is continuously stored in a database which web-service based processing modules (such as SLU, DM and language generation) access. This architecture also allows us to access the database for immediate visualization. The system presents an example of a "thick" inter-module information pipeline architecture. Individual components exchange data by means of sets of hypotheses complemented by the detailed conversational context. The database concentrates heterogeneous types of information at various levels of description in a uniform way. This facilitates dialog evaluation, data mining and online learning because data is available for querying as soon as it has been stored. There is no need for separate logging mechanisms. Multiple systems/applications are available on the same infrastructure due to a clean separation of its processing modules (SLU, DM, NLG etc.) from data storage (DBMS), and monitoring/analysis/visualization and annotation tools.

4 Visualization Tool

We developed a live web-based dialogue visualization tool that displays ongoing and past dialogue utterances, semantic interpretation confidences and distributions of confidences for incoming user acts, the dialogue manager state, and policy-based decisions and updating. An example of the visualization tool is given in figures 3 (dialogue logs) and 4 (annotation view). We are currently extending the visualization tool to display the POMDP-related information that is already present in the dialogue database.

The visualization tool shows how our dedicated SLU module produces a number of candidate semantic parses using the semantics of a domain ontology and the output of ASR.

The visualization of the internal representation of the POMDP-DM includes the N best dialogue states after each user utterance and the reranking of the action set. At the end of each dialogue session, the reward and the policy updates are shown, i.e. new or updated state entries and action values. Another plot relates the current dialogue's reward to the reward of previous dialogues (as in plots 1(b) and 1(a)).

Users are able to talk with several systems (via SIP phone connection to the dialogue system server) and see their dialogues in the visualization tool. They are able to compare the rule-based system, a randomly exploring learner that has not been trained yet, and several systems that use various pre-trained policies. These policies are obtained by dialogue simulations with user models based on data obtained from human-machine dialogues with the original rule-based dialogue manager. The web tool is available at `http://cicerone.dit.unitn.it/DialogStatistics/`.

Acknowledgments

This work was partially supported by the European Commission Marie Curie Excellence Grant for the ADAMACH project (contract No. 022593) and by LUNA STREP project (contract No. 33549).

References

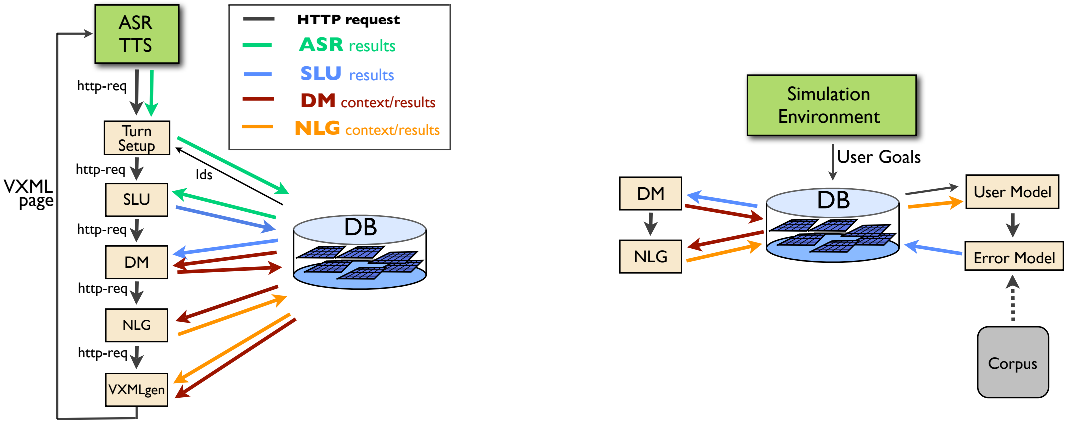
R. Higashinaka, M. Nakano, and K. Aikawa. 2003. Corpus-based discourse understanding in spoken dialogue systems. In *ACL-03*, Sapporo, Japan.

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

J. Schatzmann, K. Weilhammer, M. Stuttle, and S. Young. 2006. A Survey of Statistical User Simulation Techniques for Reinforcement-Learning of Dialogue Management Strategies. *Knowledge Engineering Review*, 21(2):97–126.

S. Varges, G. Riccardi, and S. Quarteroni. 2008. Persistent Information State in a Data-Centric Architecture. In *SIGDIAL-08*, Columbus, Ohio.

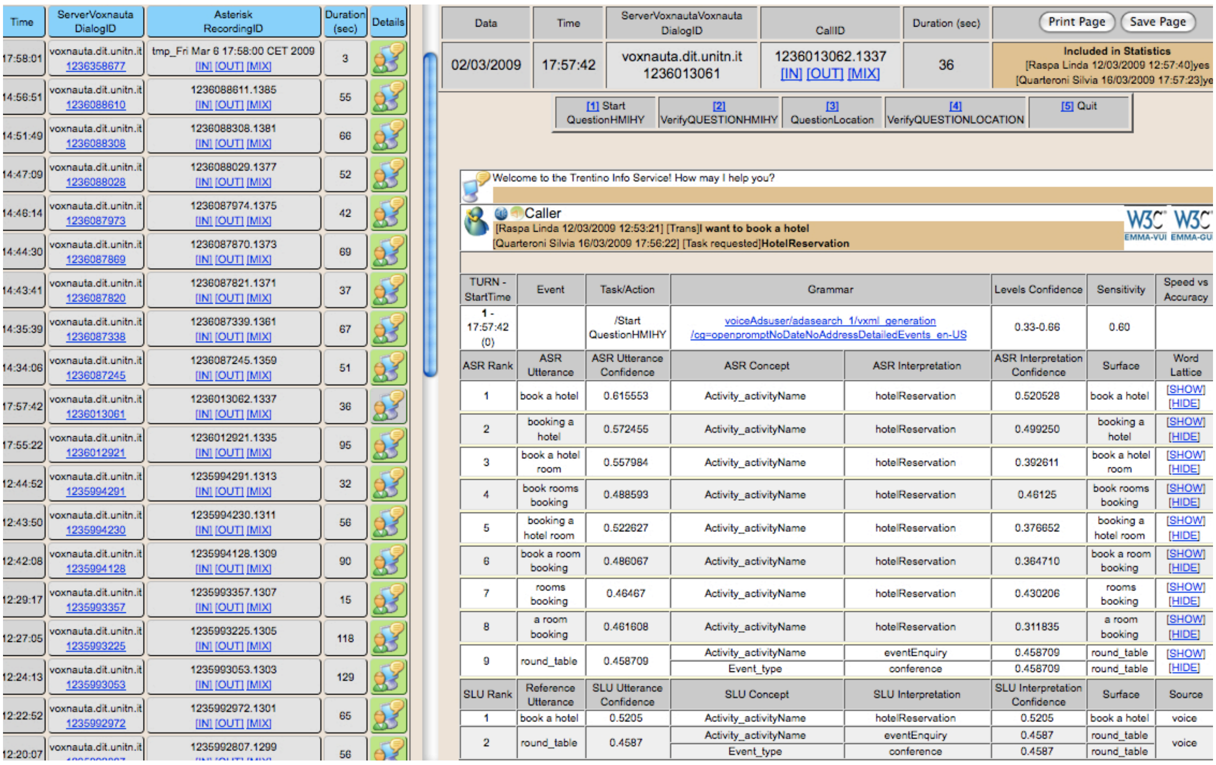
J. D. Williams and S. Young. 2006. Partially Observable Markov Decision Processes for Spoken Dialog Systems. *Computer Speech and Language*, 21(2):393–422.



(a) Turn-level information flow in the data-centric SDS architecture

(b) User simulator interface with the dialogue manager

Figure 2: Architecture for interacting with human user (left) and simulated user (right)



Time	ServerVoxnauta DialogID	Asterisk RecordingID	Duration (sec)	Details
17:58:01	voxnauta.dit.unitn.it 1236358677	tmp_Fri Mar 6 17:58:00 CET 2009 [IN] [OUT] [MIX]	3	
14:56:51	voxnauta.dit.unitn.it 1236088610	1236088611.1385 [IN] [OUT] [MIX]	55	
14:51:49	voxnauta.dit.unitn.it 1236088308	1236088308.1381 [IN] [OUT] [MIX]	66	
14:47:09	voxnauta.dit.unitn.it 1236088028	1236088029.1377 [IN] [OUT] [MIX]	52	
14:46:14	voxnauta.dit.unitn.it 1236087973	1236087974.1375 [IN] [OUT] [MIX]	42	
14:44:30	voxnauta.dit.unitn.it 1236087869	1236087870.1373 [IN] [OUT] [MIX]	69	
14:43:41	voxnauta.dit.unitn.it 1236087820	1236087821.1371 [IN] [OUT] [MIX]	37	
14:35:39	voxnauta.dit.unitn.it 1236087338	1236087339.1361 [IN] [OUT] [MIX]	67	
14:34:06	voxnauta.dit.unitn.it 1236087245	1236087245.1359 [IN] [OUT] [MIX]	51	
17:57:42	voxnauta.dit.unitn.it 1236013061	1236013062.1337 [IN] [OUT] [MIX]	36	
17:55:22	voxnauta.dit.unitn.it 1236012921	1236012921.1335 [IN] [OUT] [MIX]	95	
12:44:52	voxnauta.dit.unitn.it 1235994291	1235994291.1313 [IN] [OUT] [MIX]	32	
12:43:50	voxnauta.dit.unitn.it 1235994230	1235994230.1311 [IN] [OUT] [MIX]	56	
12:42:08	voxnauta.dit.unitn.it 1235994128	1235994128.1309 [IN] [OUT] [MIX]	90	
12:29:17	voxnauta.dit.unitn.it 1235993357	1235993357.1307 [IN] [OUT] [MIX]	15	
12:27:05	voxnauta.dit.unitn.it 1235993225	1235993225.1305 [IN] [OUT] [MIX]	118	
12:24:13	voxnauta.dit.unitn.it 1235993053	1235993053.1303 [IN] [OUT] [MIX]	129	
12:22:52	voxnauta.dit.unitn.it 1235992972	1235992972.1301 [IN] [OUT] [MIX]	65	
12:20:07	voxnauta.dit.unitn.it	1235992807.1299	56	

Data	Time	ServerVoxnautaVoxnauta DialogID	CallID	Duration (sec)	Print Page Save Page
02/03/2009	17:57:42	voxnauta.dit.unitn.it 1236013061	1236013062.1337 [IN] [OUT] [MIX]	36	**Included in Statistics** [Raspa Linda 12/03/2009 12:57:40]yes [Quarteroni Silvia 16/03/2009 17:57:23]ye

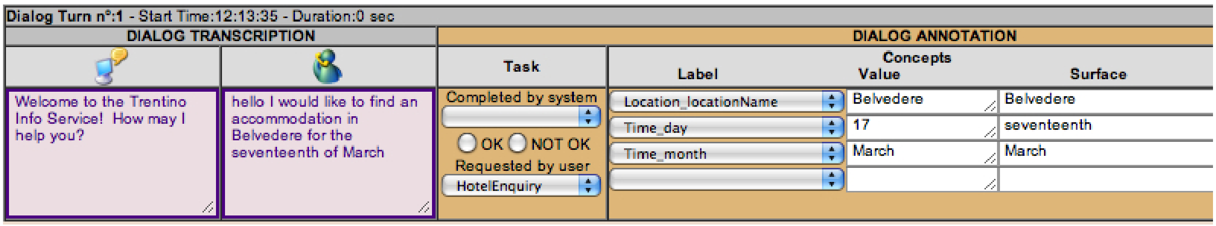
[1] Start QuestionHMIHY	[2] VerifyQUESTIONHMIHY	[3] QuestionLocation	[4] VerifyQUESTIONLOCATION	[5] Quit

Welcome to the Trentino Info Service! How may I help you?

Caller
[Raspa Linda 12/03/2009 12:53:21] [Trans]**I want to book a hotel**
[Quarteroni Silvia 16/03/2009 17:56:22] [Task requested]**HotelReservation**

TURN - StartTime	Event	Task/Action	Grammar		Levels Confidence	Sensitivity	Speed vs Accuracy
1 - 17:57:42 (0)		/Start QuestionHMIHY	voiceAdsuser/adasearch_1/vxml_generation /cg=openpromptNoDateNoAddressDetailedEvents_en-US		0.33-0.66	0.60	
ASR Rank	**ASR Utterance**	**ASR Utterance Confidence**	**ASR Concept**	**ASR Interpretation**	**ASR Interpretation Confidence**	**Surface**	**Word Lattice**
1	book a hotel	0.615553	Activity_activityName	hotelReservation	0.520528	book a hotel	[SHOW] [HIDE]
2	booking a hotel	0.572455	Activity_activityName	hotelReservation	0.499250	booking a hotel	[SHOW] [HIDE]
3	book a hotel room	0.557984	Activity_activityName	hotelReservation	0.392611	book a hotel room	[SHOW] [HIDE]
4	book rooms booking	0.488593	Activity_activityName	hotelReservation	0.46125	book rooms booking	[SHOW] [HIDE]
5	booking a hotel room	0.522627	Activity_activityName	hotelReservation	0.376652	booking a hotel room	[SHOW] [HIDE]
6	book a room booking	0.486067	Activity_activityName	hotelReservation	0.364710	book a room booking	[SHOW] [HIDE]
7	rooms booking	0.46467	Activity_activityName	hotelReservation	0.430206	rooms booking	[SHOW] [HIDE]
8	a room booking	0.461608	Activity_activityName	hotelReservation	0.311835	a room booking	[SHOW] [HIDE]
9	round_table	0.458709	Activity_activityName Event_type	eventEnquiry conference	0.458709 0.458709	round_table round_table	[SHOW] [HIDE]
SLU Rank	**Reference Utterance**	**SLU Utterance Confidence**	**SLU Concept**	**SLU Interpretation**	**SLU Interpretation Confidence**	**Surface**	**Source**
1	book a hotel	0.5205	Activity_activityName	hotelReservation	0.5205	book a hotel	voice
2	round_table	0.4587	Activity_activityName Event_type	eventEnquiry conference	0.4587 0.4587	round_table round_table	voice

Figure 3: Left pane: overview of all dialogues. Right pane: visualization of a system opening prompt followed by the user's activity request. All *distinct* SLU hypotheses (concept-value combinations) deriving from ASR are ranked based on concept-level confidence (2 in this turn).



Dialog Turn n°:1 - Start Time:12:13:35 - Duration:0 sec

DIALOG TRANSCRIPTION		DIALOG ANNOTATION			
		Task	Concepts Label	Value	Surface
Welcome to the Trentino Info Service! How may I help you?	hello I would like to find an accommodation in Belvedere for the seventeenth of March	Completed by system ○ OK ○ NOT OK Requested by user HotelEnquiry	Location_locationName Time_day Time_month	Belvedere 17 March	Belvedere seventeenth March

Figure 4: Turn annotation of task success based on previously filled dialog transcriptions (left box).