Learning Dialogue POMDP Models from Data

Hamid R. Chinaei and Brahim Chaib-draa

Computer Science and Software Engineering Department,
Laval University, Quebec, Canada
hrchinaei@damas.ift.ulaval.ca,
Brahim.Chaib-Draa@ift.ulaval.ca

Abstract. In this paper, we learn the components of dialogue POMDP models from data. In particular, we learn the states, observations, as well as transition and observation functions based on a Bayesian latent topic model using unannotated human-human dialogues. As a matter of fact, we use the Bayesian latent topic model in order to learn the intentions behind user's utterances. Similar to recent dialogue POMDPs, we use the discovered user's intentions as the states of dialogue POMDPs. However, as opposed to previous works, instead of using some keywords as POMDP observations, we use some meta observations based on the learned user's intentions. As the number of meta observations is much less than the actual observations, i.e. the number of words in the dialogue set, the POMDP learning and planning becomes tractable. The experimental results on real dialogues show that the quality of the learned models increases by increasing the number of dialogues as training data. Moreover, the experiments based on simulation show that the introduced method is robust to the ASR noise level.

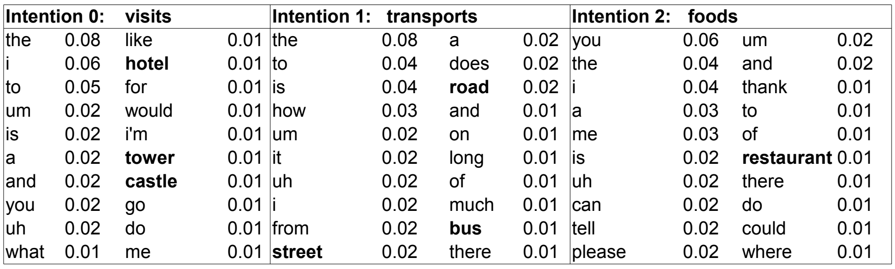
1 Introduction

Consider the following example taken from the dialogue set SACTI-2 [6], where SACTI stands for Simulated ASR-Channel Tourist Information:

U1 Is there a good restaurant we can go to tonight
U'1 [Is there a good restaurant week an hour tonight]
M1 Would you like an expensive restaurant
U2 No I think we'd like a medium priced restaurant
U'2 [No I think late like uh museum price restaurant]
M2 Cheapest restaurant is eight pounds per person

The first line shows the first user's utterance, $U1$. Because of Automatic Speech Recognition (ASR) this utterance is corrupted and is received by the system as $U'1$ in the following line in braces. $M1$ in the next line shows the system's response to the user. For each dialogue utterance, the system's goal is first to capture the user's intention and then to perform the best action which satisfies the user's intention. For instance, in the second received user's utterance, *U'2 [No I think late like uh museum price restaurant]*, the system has difficulty in finding the user's intention. In fact, in $U'2$, the system is required to understand that the user is looking for a *restaurant*; though this utterance is

C. Butz and P. Lingras (Eds.): Canadian AI 2011, LNAI 6657, pp. 86–91, 2011.



Intention 0:	**visits**			**Intention 1:**	**transports**			**Intention 2:**	**foods**		
the	0.08	like	0.01	the	0.08	a	0.02	you	0.06	um	0.02
i	0.06	**hotel**	0.01	to	0.04	does	0.02	the	0.04	and	0.02
to	0.05	for	0.01	is	0.04	**road**	0.02	i	0.04	thank	0.01
um	0.02	would	0.01	how	0.03	and	0.01	a	0.03	to	0.01
is	0.02	i'm	0.01	um	0.02	on	0.01	me	0.03	of	0.01
a	0.02	**tower**	0.01	it	0.02	long	0.01	is	0.02	**restaurant**	0.01
and	0.02	**castle**	0.01	uh	0.02	of	0.01	uh	0.02	there	0.01
you	0.02	go	0.01	i	0.02	much	0.01	can	0.02	do	0.01
uh	0.02	do	0.01	from	0.02	**bus**	0.01	tell	0.02	could	0.01
what	0.01	me	0.01	**street**	0.02	there	0.01	please	0.02	where	0.01

Fig. 1. Intentions learned by HTMM for SACTI-1, with their 20-top words and their probabilities

highly corrupted. Specifically, it contains misleading words such as *museum* that can be strong observations for another user's intention, i.e. user's intention for museums.

Recently, there has been a great interest for modelling the dialogue manager (DM) of spoken dialogue systems (SDS) using Partially Observable Markov Decision Processes (POMDPs) [8]. However, in POMDPs, similar to many other machine learning frameworks, estimating the environment dynamics is a significant issue; as it has been argued previously, for instance in [4]. In other words, the POMDP models highly impact the planned strategies. Nevertheless, a good learned model can result in desired strategies. Moreover, it can be used as a prior model in all Bayesian approaches so that the model be further updated and enhanced. As such, in this work we are interested in learning proper POMDP models for dialogue POMDPs based on human-human dialogues.

In this paper, we present a method for learning the components of dialogue POMDP models using unannotated data available in SDSs. In fact, using an unsupervised method based on Dirichlet distribution, one can learn states and observations as well as transition and observation POMDP functions. In addition, we develop a simple idea for reducing the number of observations while learning the model, and define a small practical set of observations for the designed dialogue POMDP.

2 Capturing Dialogue POMDP Model for SACTI-1

This section describes the method for learning POMDP transition and observation functions. For background about POMDPs, the reader is referred to [5]. We used Hidden Topic Markov Model (HTMM) [3] to design a dialogue POMDP, for SACTI-1 dialogues [7], publicly available at: `http://mi.eng.cam.ac.uk/projects/sacti/corpora/`. There are about 144 dialogues between 36 users and 12 experts who play the role of a DM for 24 total tasks on this data set. Similar to SACTI-2, the utterances here are also first confused using a speech recognition error simulator, and then are sent to the human experts. For an application of HTMM on dialogues in particular for learning states of the domain, the reader is referred to [1].

Figure 1 shows 3 captured user's intentions and their top 20 words with their probabilities learned by HTMM. For each intention, we have highlighted the keywords which best distinguish the intention. These intentions are for the user's intentions for request information about some visiting places, the transportation, and food places, respectively.

Without loss of generality, we can consider the user's intention as the system's state [2]. Based on the above captured intentions, we defined 3 primary states for the SACTI-1 DM as follows: *visits (v)* , *transports (t)* , and *foods (f)*. Moreover, we defined two absorb states, i.e., *Success (S)* and *Failure (F)* for dialogues which end successfully and unsuccessfully, respectively. The notion of successful or unsuccessful dialogue is defined by user. After finishing each dialogue, the user assigns the level of precision and recall. These are the only explicit feedback which we require from the user, to be able to define absorb states of dialogue POMDP. A dialogue is successful if its precision and recall is above a predefined threshold.

The set of actions are coming directly from SACTI-1 dialogue set, and they include: *GreetingFarewell*, *Inform*, *StateInterp*, *IncompleteUnknown*, *Request*, *ReqRepeat*, *RespondAffirm*, *RespondNegate*, *ExplAck*, *ReqAck*, etc. For instance *GreetingFarewell* is used for initiating or ending a dialogue, *Inform* is for giving information for a user's intention, *ReqAck* is for the DM's request for user's acknowledgement, *StateInterp* for interpreting the intentions of user, and it can be considered as implicit confirmation, etc.

The transition function is calculated using maximum likelihood with add-one smoothing to make a more robust transition model:

$$T(s_1,a_1,s_2) = \frac{Count(s_1,a_1,s_2)+1}{Count(s_1,a_1)+K}$$

where $K = |S|^2|A|$, S is the state set, and S equals to number of intentions N which is 5 in our example. For each utterance U, its corresponding state is the intention with highest probability.

For the choice of observation function, we assumed 5 observations, each one is specific for one state, i.e. user's hidden intention. we use the notation O= $\{$ *VO, TO, FO, SuccessO, FailureO* $\}$ for the meta observations for *visits, transports, foods, Success*, and *Failure*, respectively. For each user's intention, one can capture POMDP observations given each utterance $W = \{w_1, \ldots, w_{|W|}\}$ using vector β. Notice that $\beta_{w_i z}$ is the learned vector for the probability of each word w_i given each user's intention z noted as $\beta_{w_i z}$ [3]. Then, in dialogue POMDP interaction, given any arbitrary user's utterance POMDP observation o is captured as:

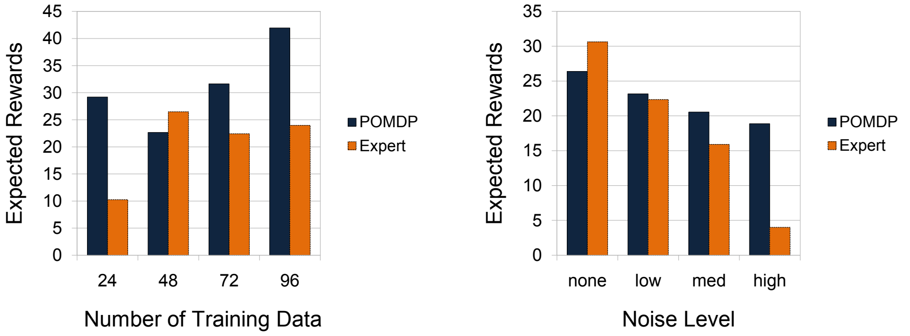
$$o = argmax_z \prod_i \beta_{w_i z}$$

Then, the observation function is estimated by taking average over belief of states given each action and state.

For the choice of reward model, similar to previous works we penalized each action in primary states by -1, i.e. -1 reward for each dialogue turn [8]. Moreover, actions in *Success* state get $+50$ as reward, and those which lead to *Failure* state get -50 reward.

3 Experiments

We generated dialogue POMDP models as described in the previous section for SACTI-1. The automatic generated dialogue POMDP models consist of 5 states, 14 actions and



(a) (b)

Fig. 2. (a): Comparison of performance in dialogue POMDPs v.s. experts with respect to the number of expert dialogues. (b): Comparison of performance in dialogue POMDPs v.s. experts with respect to the noise level.

5 meta observations (each of which is for one state) which are drawn by HTMM using 817 primitive observations (words).

We solved our POMDP models, using ZMDP software available online at: http://www.cs.cmu.edu/~trey/zmdp/. We set a uniform distribution on 3 primary states (*visits, transports, and foods*), and set discount factor to 90%. Based on simulation, we evaluated the performance of dialogue POMDP by increasing the number of expert dialogues based on the gathered rewards.

Figure 2 (a) shows that by increasing expert dialogues the dialogue POMDP models perform better. In other words, by increasing data the introduced method learns better dialogue POMDP models. The only exception is when we use 48 dialogues where the dialogue POMDP performance decreases compared to when 24 dialogues were used, and it has average performance worse than performance of experts in corresponding 48 dialogues. The reason could be use of EM for learning the model which is depended on priors α and η [3]. Moreover, EM is prone to local optima. In this work, we set the priors based on heuristic given in [3], and our trial and error experiments, which is indeed a drawback for use of parametric models in real applications.

Furthermore, based on our simulations, we evaluated the robustness of generated POMDP models to ASR noise. There are four levels of ASR noise: no noise, low noise, medium noise, and high noise. For each noise level, we randomly took 24 expert dialogues and made a dialogue POMDP model. Then, for each POMDP we performed 24 simulations and gathered their expected rewards, and compared to corresponding expert dialogues. Figure 2 (b) shows the results of these experiments. As the figure shows the dialogue POMDP models are more robust to ASR noise levels compared to expert dialogues. The only exception is with the presence of no noise, where the experts perform better. This also might be because of use of EM for learning model, where the model can converge in local minima. Nevertheless, our preliminarily results based on simulation shows that dialogue POMDP models are much more robust to higher levels of noise compared to expert performance.

Table 1. Left: Sample results of applying HTMM on SACTI-1 Right: Sample results of simulation for SACTI-1 dialogue POMDP

	. . .
$U1$	yeah hello this is johan schmulka uh and i'm uh searching for a bar in this town can you may be tell me where the cafe blu is
U'1	[hello this is now seven four bus and do you tell me where to cafe blu is]
$o1$	*FO*
$B1$	*t:0.000000 v:0.000000 f:1.000000*
$a1$:	*Inform(foods)*
$M1$	cafe blu is on alexander street
$U2$	oh um yeah how can i get to alexander street and where exactly is it i know there a shopping area on alexander street um
U'2	[i am yeah i am at the alexander street and where is it was on a the center of alexander street]
$o2$	*TO*
$B2$	*t:0.999992 v:0.000008 f:0.000000*
$a2$:	*Inform(transports)*
	. . .

	. . .
$a1$:	*GreetingFarewell*
$M1$:	How can I help you?
$o2$:	*VO*
$B1$:	*t:0.048145 v:0.912760 f:0.039093*
$a2$:	*Inform(visits)*
$M2$:	Here is information about visiting areas
$o2$:	*TO*
$B2$:	*t:0.967322 v:0.008186 f:0.024490*
$a3$:	*Inform(transports)*
$M3$:	*Here is information about transportation*
$o3$:	*TO*
$B3$:	*t:0.993852 v:0.000314 f:0.005833*
$a4$:	*ReqAck(transports)*
$M4$:	*Are you looking for transportation*
$o4$:	*TO*
$B4$:	*t:0.945658 v:0.048333 f:0.006008*
$a5$:	*Inform(transports)*
$M5$:	Here is information about transportation

Moreover, the left side of Table 1 shows a sample dialogue from SACTI-1 dialogue set after applying HTMM on dialogues. In fact, this is a sample of data used for learning dialogue POMDP model. The first line of the table shows the first user's utterance ($U1$). Because of ASR this utterance is corrupted which is the following line in braces, $U'1$. The next line $o1$ is the observation behind $U'1$ which is used in the time of dialogue POMDP interaction. Note that it is assumed that each user utterance corresponds to one user's intention. So, for each system's observation the values in the following line show the system's belief over possible hidden intentions ($B1$). The next line, $a1$ shows the DM's action in the form of dialogue acts. For instance, *Inform(foods)* is the dialogue act for the actual DM's utterance in the following line, i.e. *M1: cafe blu is on alexander street*.

Furthermore, the right side of Table 1 shows samples of our simulation of dialogue POMDP. In the simulation time, for instance action $a1$, *GreetingFarewell* is generated by dialogue POMDP manager, the description of this action is shown in $M1$, *How can I help you?*. Then, the observation $o2$ is generated by environment, *VO*. For instance, the received user's utterance could have been something like *U'1=I would like a hour there museum first*, which easily the intention behind this can be calculated using β_{ws} and equation 1. However, notice that these results are only based on dialogue POMDP simulation; where there is no actual user's utterance, but only simulated meta observations o_i. As the table shows, dialogue POMDP performance seems intuitive. For instance, in $a4$ the dialogue POMDP requests for acknowledgement that the user actually looks for *transports*, since dialogue POMDP already informed the user about *transports* in a_3.

4 Conclusion and Future Work

A common problem in dialogue POMDP frameworks is calculating the dialogue POMDP policy. If we can estimate the POMDP model in particular the transition, observation, and reward functions then we are able to use common dynamic programming approaches for calculating POMDP policies. In this context, [8] used POMDPs for modelling a DM and defined the observation function based on confidence scores which are in turn based on some recognition features. However, the work here is tackled differently. We consider all the words in an utterance and consider the highest intention under the utterance as the meta observation for the POMDP. This makes the work presented here particularly different from [2] where the authors simply used some state keywords together with a few other words for modelling SDS POMDP observations and observation function.

However, the evaluation done here is in a rather small domain for real dialogue systems. The number of states needs to be increased and the learned model should be evaluated accordingly. Moreover, the definition of states here is a simple intention state whereas in real dialogue domains the information or dialogue states are more complex. Then, the challenge would be to compare in particular the learned observation function presented here with confidence score based ones such as in in [8], as well as keyword based ones as presented in [2].

References

1. Chinaei, H.R., Chaib-draa, B., Lamontagne, L.: Learning user intentions in spoken dialogue systems. In: Filipe, J., Fred, A., Sharp, B. (eds.) ICAART 2009. CCIS, vol. 67, pp. 107–114. Springer, Heidelberg (2010)
2. Doshi, F., Roy, N.: Spoken language interaction with model uncertainty: an adaptive human-robot interaction system. Connection Science 20(4), 299–318 (2008)
3. Gruber, A., Rosen-Zvi, M., Weiss, Y.: Hidden topic markov models. In: Artificial Intelligence and Statistics (AISTATS), San Juan, Puerto Rico (2007)
4. Liu, Y., Ji, G., Yang, Z.: Using Learned PSR Model for Planning under Uncertainty. Advances in Artificial Intelligence, 309–314 (2010)
5. Pineau, J., Gordon, G., Thrun, S.: Point-based value iteration: An anytime algorithm for pomdps. In: International Joint Conference on Artificial Intelligence (IJCAI), pp. 1025–1032 (August 2003)
6. Weilhammer, K., Williams, J.D., Young, S.: The SACTI-2 Corpus: Guide for Research Users, Cambridge University. Technical report (2004)
7. Williams, J.D., Young, S.: The SACTI-1 Corpus: Guide for Research Users. Cambridge University Department of Engineering. Technical report (2005)
8. Williams, J.D., Young, S.: Partially observable markov decision processes for spoken dialog systems. Computer Speech and Language 21, 393–422 (2007)