BAYESIAN REINFORCEMENT LEARNING FOR POMDP-BASED DIALOGUE SYSTEMS

ShaoWei Png, Joelle Pineau

McGill University
School of Computer Science
Montreal, QC H3A 1A8
shaowei.png@mail.mcgill.ca, jpineau@cs.mcgill.ca

ABSTRACT

Spoken dialogue systems are gaining popularity with improvements in speech recognition technologies. Dialogue systems can be modeled effectively using POMDPs, achieving improvements in robustness. However, past research on POMDPs-based dialogue system assumes that the model parameters are known. This limitation can be addressed through model-based Bayesian reinforcement learning, which offers a rich framework for simultaneous learning and planning. However, due to the high complexity of the framework, a major challenge is to scale up these algorithms for complex dialogue systems. In this work, we show that by exploiting certain known components of the system, such as knowledge of symmetrical properties, and using an approximate online planning algorithm, we are able to apply Bayesian RL on a realistic spoken dialogue system domain.

***Index Terms*—** POMDPs (Partially Observable Markov Decision Processes), Spoken Dialogue, Reinforcement Learning, Bayesian Learning

1. INTRODUCTION

Spoken dialogue systems are getting increasingly popular with improvements in speech recognition technologies. They enable tasks to be completed using spoken language, and have been applied in various domains such as an in-car spoken dialogue system [1], an automated receptionist [2], and a robotics wheelchair [3].

In a typical dialogue system, the intention of the user is unknown. The system has to guess the intention, usually by using a voice recognition interface, which is often ambiguous due to noise in human communications. Hence, it is difficult to determine which action to take at any given point in the conversation.

In the past few years, it has been shown that spoken dialogue systems can be modeled as Partially Observable Markov Decision Processes (POMDPs), achieving improvements in robustness [4, 5, 6]. Using a POMDPs framework helps to incorporate uncertainty in the dialogue system, and allows actions to be chosen based on an optimization criteria.

However, past research in POMDP-based dialogue system has always assumed a fixed and known POMDP model, which is unrealistic in many applications. It is not possible to know exactly how noisy the speech recognition is because of several factors that are hard to determine, for instance, the reliability of the voice recognition device, or the accent of the speaker.

Reinforcement Learning (RL) models have proved to be effective for learning, but most RL methods do not explicitly minimize the learning cost. This is particularly important in spoken dialogue systems. Querying too few times leads to a wrong decision, whereas querying too frequently frustrates users as they have to repeat what they have already said. Thus, it is important to develop methods for efficient low-cost learning.

Bayesian RL maintains a posterior distribution over all possible model parameters, and computes an action selection policy that is optimal with respect to this posterior [7, 8]. This enables us to make use of prior knowledge to learn the parameters more efficiently. Model-based Bayesian RL methods are typically computationally intensive, thus applications are still limited to small domains.

This paper focuses on how we can obtain good decisions from the spoken dialogue model despite inaccurate initial model parameters. This is achieved by modeling the dialogue system as a Bayes-Adaptive POMDP (BAPOMDP) model [9], and exploiting certain known symmetrical properties of the system to obtain scalable solutions.

2. POMDPS

POMDPs provide a principled mathematical framework for modeling non-deterministic, sequential decision-making problems [10, 11]. Formally, a discrete POMDP is specified as a tuple $(S, A, O, T, Z, R, \gamma)$, where S is a set of states, A is a set of actions, and O is a set of observations. When applied to dialogue systems, the state of the model captures the user's

The authors gratefully acknowledge support from the Natural Sciences and Engineering Council of Canada (NSERC) and the Fonds Québécois de la Recherche sur la Nature et les Technologies (FQRNT)

intent and the dialogue state. The actions define the set of possible responses.

At each time step, the agent lies in some state $s \in S$. It takes an action $a \in A$ and moves from s to a new state s'. Due to the uncertainty in action, the end state s' is modeled as a conditional probability function $p(s'|s,a)$. The agent then makes an observation to gather information on its own state. The observation result $o \in O$ is modeled as a conditional probability function $p(o|s',a)$.

At each time step, the agent receives a reward $R(s,a)$ if it takes action a in state s. The agent's goal is to maximize its expected total reward by choosing a suitable sequence of actions. For infinite-horizon POMDPs, the sequence of actions has infinite length. We specify a discount factor, $\gamma \in [0,1)$ so that the total reward is finite and the problem is well defined.

A belief state b is a probability distribution over S. We let $b(s)$ denote the probability assigned to world state s by belief state b. Solving a POMDP consists of two steps executed iteratively. The first step is action selection. If the agents current belief is b, it finds the action a that maximizes the future reward. The second step is belief estimation. After the agent takes an action a and receives an observation o, its new belief b' is given by

$$b'(s) = \eta p(o|s',a) \sum_{s \in S} p(s'|s,a) b(s) \qquad (1)$$

where η is a normalizing constant. The process then repeats.

The POMDP model has to go through a planning phase. During this phase, it finds an optimal policy which describes an optimal mapping of action to belief for all possible beliefs.

The dialogue system uses this policy to decide how to interact with the users. The optimal policy for a POMDP is one that chooses an action that maximises the expected reward. Finding an optimal policy exactly for non trivial POMDPs problems is computationally intractable. A near-optimal policy can be computed significantly faster than an exact one. This is called an offline approach [12, 13, 14, 15, 16].

On the other hand, online approaches reduce the complexity of the problem by planning online for only the current information state [17, 18, 19]. It considers only a small horizon of possible scenarios.

3. BAPOMDPS

The Bayes-Adaptive POMDP (BAPOMDP) model is an algorithm for learning and planning in POMDPs under parameter uncertainty [9]. Here, we assume that the state, action, observation spaces are finite and known. In this paper, we focus on the case where only $p(o|s,a)$ is unknown, as this is most relevant for practical dialogue systems. The other model parameters $p(s'|s,a), R(s,a)$ are known. In general, the BAPOMDP can solve problems when $p(s'|s,a), R(s,a)$ are also unknown [9].

To account for the uncertainty, the BAPOMDP framework uses Dirichlet distributions, which are probability distributions over the parameters of multinomial distributions. The objective is to learn an optimal policy, such that actions are chosen to maximize reward with respect to the posterior captured by the Dirichlet distribution.

The state space S' of the BAPOMDP is defined as $S' = S * \mathcal{O}$ where $\mathcal{O} = \{\psi \in N^{|S||A||Z|} | \forall (s,a), \sum_{z \in Z} \psi^a_{sz} > 0\}$ represents the space in which ψ lies. ψ is the vector of all observation counts, and $\psi^a_{s'z}$ is the numnber of times observation z was made in state s' after doing action a.

It has been shown that the BAPOMDP is an instance of POMDP [9]. As such we need to track b. To do this in a tractable way, we consider an approximation whereby we do the exact belief update (Eqn. 1) at a given time step, but only keep the K most probable belief states in the new belief b' and renormalise b'.

In most realistic domains, an exact online planning algorithm is not tractable. We approximate the solution using Real Time Belief State Search (RTBSS) [20] with a heuristic $\sum_{i=1}^{d} \gamma^i R_{max}$, where d is the depth of the search and γ is the discount factor. RTBSS is a forward branch and bound search in the belief space.

4. SMARTWHEELER DIALOGUE DOMAIN

The SmartWheeler Dialogue domain is a POMDP model used for dialogue management between a user and an intelligent wheelchair. It is a modification of the POMDP model described in [21]. In this domain, the user has an unknown intent, and the robot has to execute an action based on its guess of the user's intent. When it has identified the user's intent, the robot can execute a command action, receiving a positive reward if correct, and zero reward otherwise. The robot can also execute one query action, which is strictly information gathering. This returns an observation giving an indication of the user's intent. Observations are not fully accurate.

We do not know the observation parameters, but we assume that we know the state transitions and the rewards. We also have some prior knowledge of similiarities in the observation parameters. Our goal is to come up with a policy that gives us a reasonable action for all beliefs despite not knowing the parameters at the onset. This is achieved by learning the observation parameters using the BAPOMDP framework.

In this domain, there are 25 states in the POMDP model. Each state corresponds to the user's intent, such as *"drive one meter forward"* or *"set speed to fast"*. There are $25 * 25 = 625$ unknown observation parameters to learn. They are represented by the squares in Fig 1. Each corresponds to the probability $p(o|s,a)$. We learn each parameter separately by making use of Dirichlet counts, and updating the counts each time we make an observation.

In a typical dialogue system, many parameters are likely

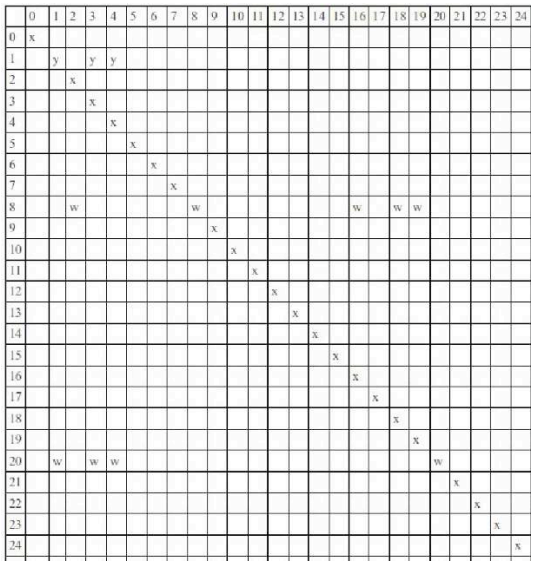

Fig. 1. Matrix for observation probability parameters with symmetry

to be similar. For instance, the phrase *"drive slowly backward"* is similar to *"drive slowly forward"* and *"drive slowly one meter backward"*, but is very different from *"avoid obstacle"*. Using the knowledge that certain observation parameters have similar values, we can learn the parameters in a faster manner. In Fig 1, similar value parameters are represented by the same letters, w, x, y or o. Note that all the unlabeled squares are actually o.

This is how we make use of symmetry to update the Dirichlet parameters upon receiving a new observation. Since certain observation parameters have approximately the same values, whenever we have an observation for a particular observation parameter, besides updating the corresponding Dirichlet parameter, we also update the Dirichlet parameters corresponding to other similar observation parameters.

5. RESULTS

The aim of our experiments is to evaluate the performance of the BAPOMDP approach under different conditions. First, we investigate the effects of having different initial estimation of the observation parameters, and second, we measure the impact of using symmetry to update the counts.

In the actual dialogue model, the value of the observation parameter x is set to 0.97. In our experiments, we consider different priors from 0.5 to 0.9.

We run our experiments using two different ways of updating the counts for estimating the observation probability parameters. The first approach is the usual way of updating each parameter independently. The second approach makes use of symmetry with the parameters.

Each BAPOMDP simulation consists of 100 episodes.

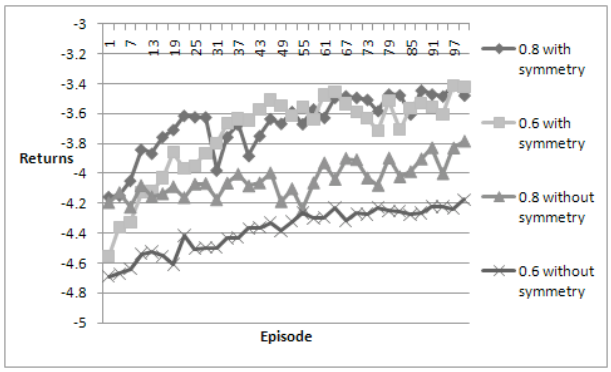

Fig. 2. Returns with different priors and whether symmetry was used in the count updates

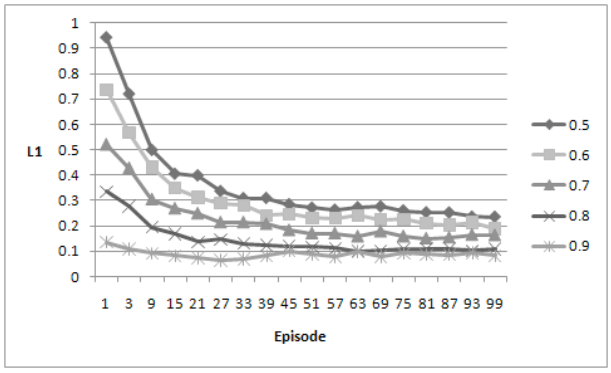

Fig. 3. Model accuracy with different priors, using symmetry for count updates

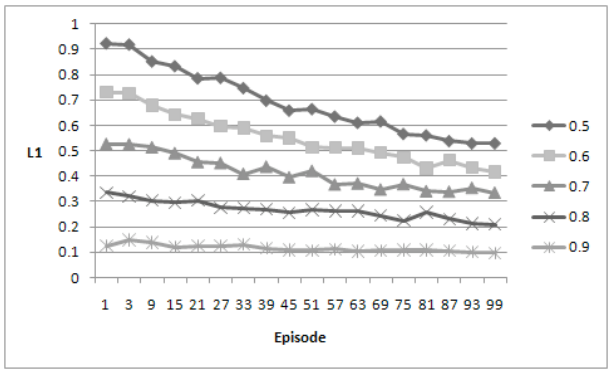

Fig. 4. Model accuracy with different priors, without using symmetry for count updates

Each episode is a short dialogue sequence trial, which terminates when the agent chooses a "command" action. At this point, the POMDP state (the user's intent) is reset, but the distribution over the observation count vector is carried over to the next episode.

We measure the empiricial returns of the policy under the various conditions. This corresponds to the total rewards

achieved by the robot. We also measure the L1-distance, measured as $\sum_s |b(s) - b'(s)|$. This is an indication of the accuracy of the estimated model. The smaller the distance between the real belief and the estimated belief, the more accurate the model is.

Our experiments show that using symmetry results in a larger return, and leads to a faster convergence of the eventual returns. This is illustrated in Fig 2. Using symmetry to update the observation counts also results in a faster convergence to the correct model as shown in Fig 3 and 4. Even with poor intial estimation of the observation parameters (poor priors), we obtain good convergence to the presumed optimal returns and good model accuracy, assuming we leverage symmetry when updating the observation parameter counts.

6. DISCUSSION

In this paper, we propose a Bayesian reinforcement learning framework for simultaneous learning and decision making on a robust spoken dialogue management, and present tractable algorithms for applying this framework in large domains. We also demonstrate the benefits of such an approach on a human-robot interaction task.

This framework is mathematically sound, and the algorithms are tractable. Even though knowledge engineering in terms of defining the states, actions, priors and rewards, is still a challenge, this is still applicable in many domains.

As voice user interfaces become more ubiquitious in our daily lives, such as in our mobile devices and automated telephone operators, we believe this is a first step towards customizable user-specific interfaces.

7. REFERENCES

[1] K. Georgila, J. Henderson, and O. Lemon, "User simulation for spoken dialogue systems: Learning and evaluation," in *ICSLP*, 2006.

[2] R. Nisimura, T. Uchida, A. Lee, H. Saruwatari, K. Shikano, and Y. Matsumoto, "ASKA: receptionist robot with speech dialogue system," in *ICRA*, 2002.

[3] A. Atrash and J. Pineau, "A bayesian reinforcement learning approach for customizing human-robot interfaces," in *IUI*, 2009.

[4] S.P. Singh, D.J. Litman, M.J. Kearns, and M.A. Walker, "Optimizing dialogue management with reinforcement learning: Experiments with the NJFun system," *JAIR*, vol. 16, no. 1, 2002.

[5] N. Roy, J. Pineau, and S. Thrun, "Spoken dialogue management using probabilistic reasoning," in *ACL*, 2000.

[6] J.D. Williams, P. Poupart, and S. Young, "Partially observable Markov decision processes with continuous observations for dialogue management," *Recent Trends in Discourse and Dialogue*, 2008.

[7] R. Dearden, N. Friedman, and D. Andre, "Model based Bayesian exploration," in *UAI*, 1999.

[8] P. Poupart, N. Vlassis, J. Hoey, and K. Regan, "An analytic solution to discrete Bayesian reinforcement learning," in *ICML*, 2006.

[9] S. Ross, B. Chaib-draa, and J. Pineau, "Bayes-Adaptive POMDPs," in *NIPS*, 2007.

[10] E.J. Sondik, "The Optimal Control of Partially Observable Markov Processes.," in *PhD thesis, Stanford University*, 1971.

[11] L.P. Kaelbling, M.L. Littman, and A.R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, 1998.

[12] J. Pineau, G. Gordon, and S. Thrun, "Point-based value iteration: An anytime algorithm for POMDPs," in *IJCAI*, 2003.

[13] T. Smith and R. Simmons, "Heuristic search value iteration for POMDPs," in *UAI*, 2004.

[14] M.T.J. Spaan and N. Vlassis, "A point-based POMDP algorithm for robot planning," in *ICRA*, 2004.

[15] H. Kurniawati, D. Hsu, and W.S. Lee, "SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces," in *RSS*, 2008.

[16] S. Ong, S. Png, D. Hsu, and W. Lee, "POMDPs for robotic tasks with mixed observability," in *RSS*, 2009.

[17] S. Ross, J. Pineau, S. Paquet, and B. Chaib-Draa, "Online planning algorithms for POMDPs," *JAIR*, vol. 32, no. 1, 2008.

[18] G. Shani, R. Brafman, and S. Shimony, "Model-based online learning of POMDPs," *Machine Learning: ECML 2005*, pp. 353–364, 2005.

[19] D.P. Bertsekas and D.A. Castanon, "Rollout algorithms for stochastic scheduling problems," *Journal of Heuristics*, vol. 5, no. 1, 1999.

[20] S. Paquet, L. Tobin, and B. Chaib-draa, "Real-time decision making for large POMDPs," *Advances in Artificial Intelligence*, 2005.

[21] A. Atrash, R. Kaplow, J. Villemure, R. West, H. Yamani, and J. Pineau., "Development and validation of a robust speech interface for improved human-robot interaction," *IJSR*, vol. 1, 2009.