Learning Models for Dialog Management

Finale Doshi and Nicholas Roy
CSAIL, Massachusetts Institute of Technology

Introduction

Spoken dialog managers allow for natural human-computer interaction, but noisy voice recognition and linguistic ambiguities make it difficult to decipher the user's intent. Partially Observable Markov Decision Processes (POMDPs) have succeeded in dialog management domain because they are robust to uncertainty in the dialog ([1],[2],[3]). However, POMDP models are generally require a large number of parameters that are difficult to specify from domain knowledge, and gathering enough data to estimate them a priori is expensive. We take a Bayesian approach to learning the parameters online [4].

POMDP Background

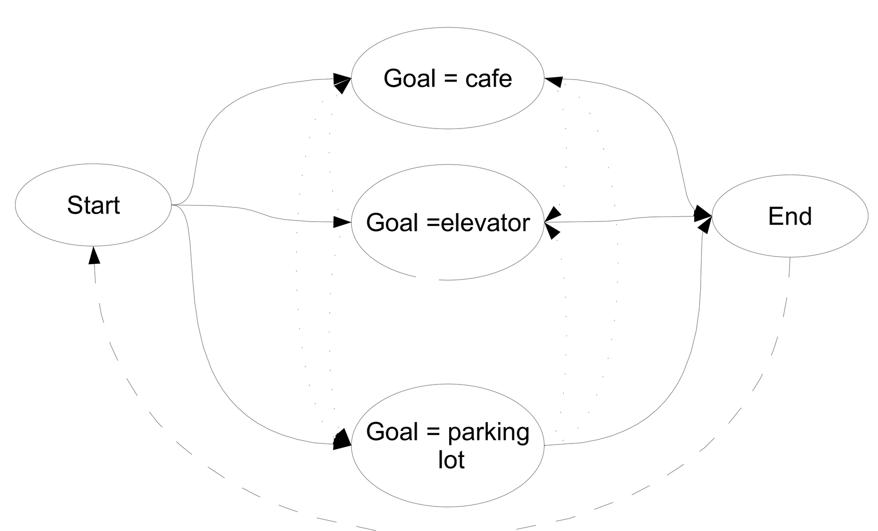
Model:
- Set of states (S), actions (A), and observations (O)
- Transition Model T(s'|s,a). Observation Model Ω(o|s,a), and Reward Model R(s,a)
- Discount factor γ

Solve for the value of a belief over states, V(b), via the dynamic programming recursion:

$$V(b) = \max_{a \in A} Q(b,a),$$
$$Q(b,a) = R(b,a) + \gamma \sum_{b' \in B} T(b'|b,a)V(b'),$$
$$Q(b,a) = R(b,a) + \gamma \sum_{o \in O} \Omega(o|b,a)V(b_a^o),$$

POMDP Dialog Model

- Place Gaussian priors over rewards R(s,a)
- Place Dirichlet priors over each transition T(.|s,a) and Ω(.|s,a)
- Expert specifies a mode value and "pre-observation count" confidence score for each prior.



[1] J. Williams and S. Young (2005). "Scaling up POMDPs for Dialogue Management: the Summary POMDP Method." IEEE workshop on Automatic Speech Recognition and Understanding (ASRU2005), Cancun, Mexico.

[2] N. Roy, J. Pineau and S. Thrun. ``Spoken Dialog Management for Robots''. Association for Computational Linguistics (ACL 2000). Hong Kong, Oct. 2000

[3] D. Litman. ,S. Singh,, M. Kearns, and M. Walker. 2000. NJFun: a reinforcement learning spoken dialogue system. In ANLP/NAACL 2000 Workshop on Conversational Systems - Volume 3 (Seattle, Washington, May 04 - 04, 2000). ANLP/NAACL Workshops. Association for Computational Linguistics, Morristown, NJ, 17-20.

[4] R. Dearden, N. Friedman, and D. Andre (1999). "Model based Bayesian Exploration." Proceedings of Fifteenth Conference on Uncertainty in Artificial Intelligence. San Francisco: Morgan Kaufmann.

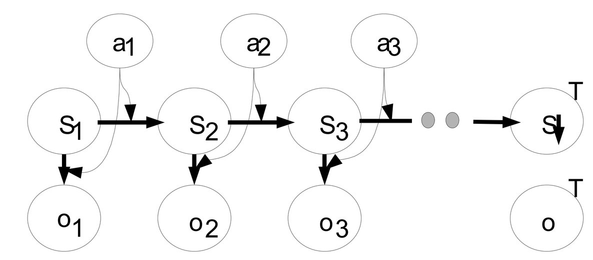
Maximizing Expected Performance

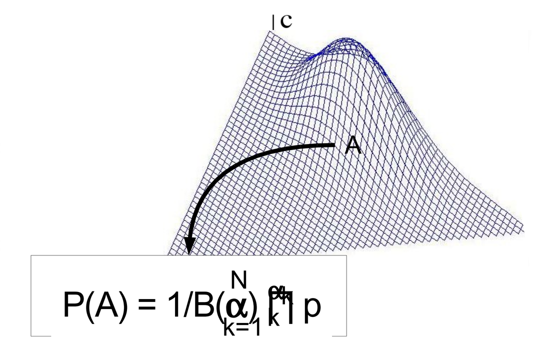
Key Fact: The policy using expected value of the parameters maximizes expected reward.

Approach:

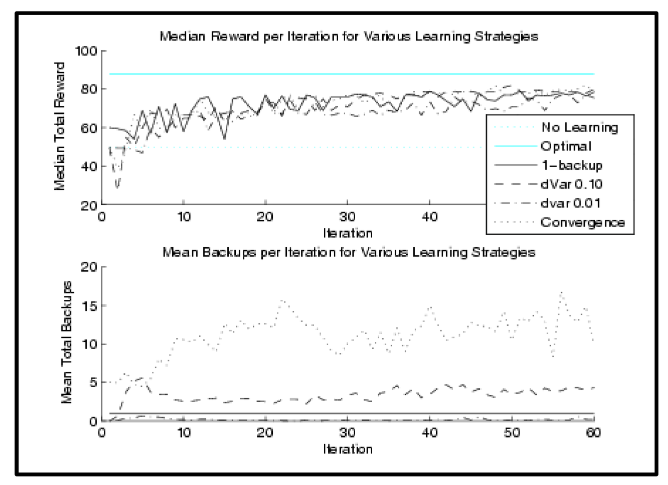
(1) Store history from an interaction. Use history to determine how to update the parameters.

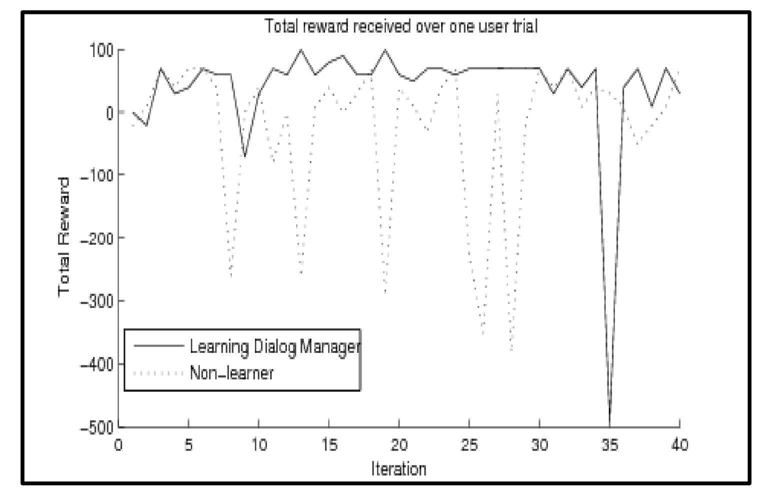
(2) Update expected-value POMDP solution with additional backups. Heuristic: backup proportionally to the variance reduction in the parameters.





Result: Heuristic has similar performance with less computation. We are able to learn user preferences and ambiguous words on a wheelchair dialog manager.





Non-learner
User: Take me to the elevator.
Robot: Where did you want to go?
User: The Gates elevator please.
Robot: Do you want to go to the Gates Elevator?
User: Yes.
Robot: Going to Gates.

Learner
User: Take me to the elevator.
Robot: Do you want to go to the Gates Tower?
User: Yes
Robot: Going to Gates

Incorporate Parameters into the POMDP

Why? If the POMDP state includes the parameters, the POMDP can take conscious actions to reduce parameter uncertainty. For now, we only learn the reward model.

Explicit Reward Approach:

(1) Begin with a prior on rewards (user preferences).

(2) The dialog manager observes a reward after each action and updates its prior.

(3) We resample a new set of reachable beliefs and perform further backups on the policy.

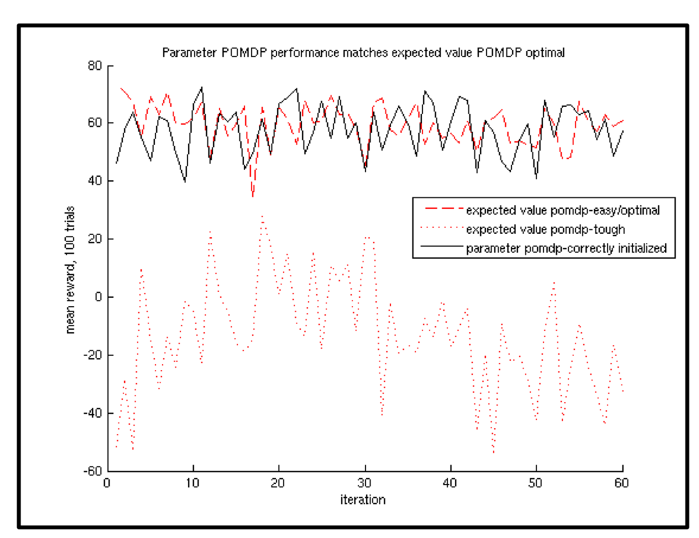
Implicit Reward Approach: Instead of receiving a direct reward, the dialog manager now has the option of asking questions about what it should do. Thus, it can infer the consequences of a poor decision without suffering the consequences.

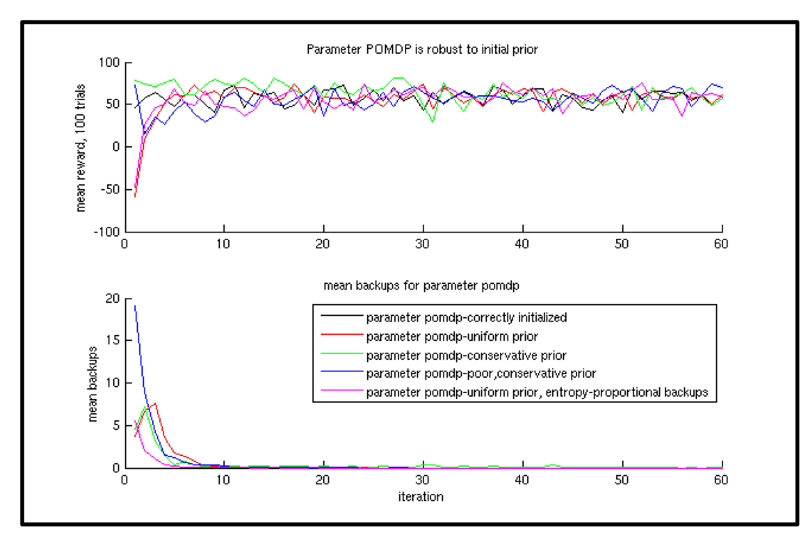
(1) Begin with a prior on rewards (user preferences).

(2) The dialog manager updates its prior based on meta-action queries.

(3) We resample a new set of reachable beliefs and perform further backups on the policy.

Results:





Continuing Work

Improving Meta-Actions: What are the best queries to ask? How can we encode them in ways that will still allow us to solve the POMDP efficiently?

Solving Structured POMDPs: Our current approach does not take into account the structure (especially the action-symmetry) in the dialog POMDP. What kind of factorizations and graph-based approaches can help us solve these POMDPs more efficiently?