# Learning Belief Representations for Partially Observable Deep RL

Andrew Wang*[12] Andrew C. Li*[12] Toryn Q. Klassen[123] Rodrigo Toro Icarte[45] Sheila A. McIlraith[123]

[1]Department of Computer Science, University of Toronto [2]Vector Institute [3]Schwartz Reisman Institute
[4]Pontificia Universidad Católica de Chile [5]Centro Nacional de Inteligencia Artificial * denotes equal contribution

## Summary

We study how RL agents can exploit ground-truth state information available during training in partially observable environments. We propose a novel algorithm and outperform baselines.

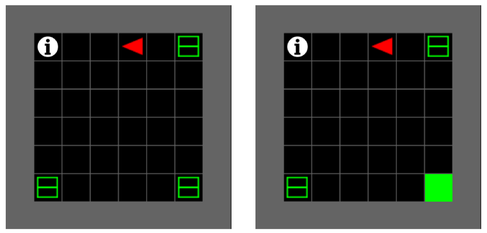
- ✔ We assume ground-truth state information is available at training time, but not deployment time.
- ✔ We observe that existing algorithms get stuck in local optimum, especially in problems that require information-seeking behavior
- ✔ We propose a method that learns a belief over a compact state representation, and train a policy conditioned on the current observation and belief
- ✔ Experiments show our method works on a variety of challenging environments, including a high-dimensional continuous state, action environment

## Introduction

**Setting:** Assume knowledge of underlying ground-truth state information at training time.

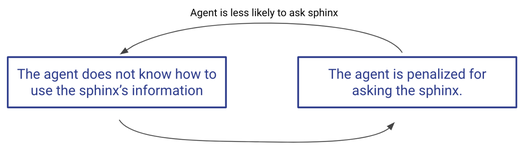
### Motivating Example (Sphinx Domain)

- ✔ One of the boxes contains a +1 reward.
- ✔ Can only open one box per episode.
- ✔ You can ask the sphinx to reveal the correct box for one timestep (-0.2 reward)



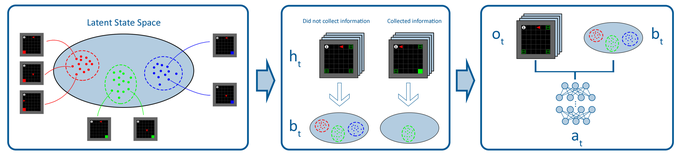
- ✔ **Optimal Strategy:** Ask the sphinx, remember the correct box, and go to it.
- ✔ **Common Trap:** Open a random box for 0.33 return.
- ✔ **Requires:** Information-seeking, Memory, Probabilistic Inference

### Information Seeking Dilemma




**Idea:** Use state information as a stronger training signal vs. only rewards

## Method




### 1. Representation Learning

**Goal: Learn a compact neural representation $\phi(s_t)$ of state.**

- ✔ Train a representation $(\phi(s_t), \psi(o_t))$ that's bisimilar to $s_t$, where:

| | | |
|---|---|---|
| **Task-irrelevant information** | This information appears in both $s_t$ and $o_t$ | discarded |
| **Task-relevant, observable information** | This information appears in both $s_t$ and $o_t$ | $\psi(o_t)$ |
| **Task-relevant, unobservable information** | This information appears in $s_t$ but not $o_t$ | $\phi(s_t)$ |

**Input:** Dataset $\mathcal{D} = \{(s_i, o_i, a_i, r_i, s_i', o_i') : 1 \le i \le N\}$
**repeat**
  Sample data $(s, o, a, r, s', o') \sim \mathcal{D}$.
  Sample stochastic encodings $u_s \sim \phi(s)$, $u_o \sim \psi(o)$.
  Predict rewards/dynamics $\hat{r}$, $\hat{u}_{s'}$, $\hat{u}_{o'} = g(u_s, u_o, a)$.
  Compute stochastic targets $u_{s'} \sim \phi(s')$, $u_{o'} \sim \psi(o')$.
  $\mathcal{L}_r = (r - \hat{r})^2$
  $\mathcal{L}_s = ||\text{stop\_grad}(u_{s'}) - \hat{u}_{s'}||^2$
  $\mathcal{L}_o = ||\text{stop\_grad}(u_{o'}) - \hat{u}_{o'}||^2$
  $\mathcal{L}_{\text{KL}} = \text{KL}[\phi(s) || \mathcal{N}(0, 1)]$
  Update $\phi, \psi, g$ on loss $\lambda_r \mathcal{L}_r + \lambda_s \mathcal{L}_s + \lambda_o \mathcal{L}_o + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}}$.
**until** convergence
**Output:** State encoder $\phi$.

### 2. Belief Modelling

**Goal: Model the belief distribution over compact state features with VAE.**

- ✔ Uses state labels as training targets when modelling the belief $p(\phi(s_t)|h_t)$
- ✔ $\mathcal{L}^{\text{VAE}}_{\theta,\xi}(\phi(s), h) = \mathbb{E}_{q_\xi(z|\phi(s),h)}\big[\log p(z) + \log p_\theta(\phi(s)|h, z) - \log q_\xi(z|\phi(s), h)\big]$

### 3. Policy Learning

**Goal: Learn a belief-conditioned policy $\pi(a_t|o_t, b(h_t))$.**

- ✔ Represent the belief state as $k$ samples from VAE $b(h_t) = (\hat{u}_1, \ldots, \hat{u}_k) \sim p_\theta(\phi(s)|h)$.
- ✔ Aggregate the samples: $W(b(h_t)) = W_{\text{agg}}\Big(\frac{1}{n}\sum_{i=1}^{n} W_{\text{enc}}(\hat{u}_{t,i})\Big)$.
- ✔ Does not require access to states to execute

## Experiments

### Domains

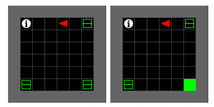


Fig. 1: Goal: reach the box containing +1 reward on each episode. $8 \times 8 \times 3$ grid. NoisyTV variation: $s_t, o_t$ flash random colours. Lying sphinx variation: sphinx lies 50% of time.

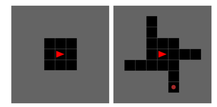
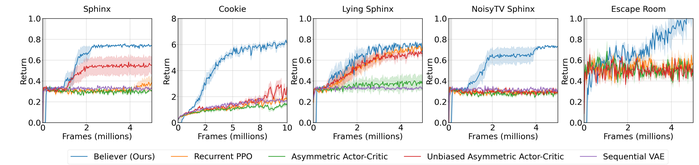


Fig. 2: Goal: eat as many cookies as possible that generates in one of three hallways. Press a button in the North hallway to spawn cookie. $9 \times 9 \times 3$ grid.

Fig. 3: Goal: Find the exit pointed to by statues across the map. Continuous-state, continuous-action environment. Observations represented as $100 \times 100$ image.

### Main Results

Our method improves upon baselines in all domains. We compared vs. Recurrent PPO, Asymmetric Actor-Critic, Unbiased Asymmetric Actor-Critic, Sequential VAE.




### Ablations

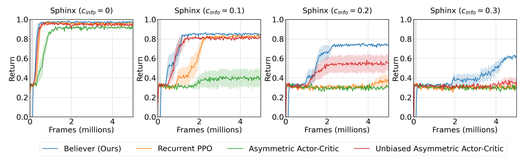



Fig. 4: Varying the cost of information for the Sphinx domains. Most approaches learn to seek information when cost of information is low, but when cost of information is high, only our method can make any progress.

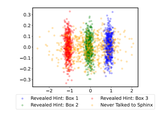



Fig. 5: A visualization of various belief states learned in the Sphinx task.

## Takeaways

| | Standard PO-RL | Our Approach |
|---|---|---|
| **Goal** | Learn a POMDP policy $\pi(a_t\|h_t)$ | Learn a POMDP policy $\pi(a_t\|h_t)$ |
| **Learns** | Agent learns by observing **observations** and **rewards** | Agent learns by observing **observations**, **states**, and **rewards** |
| **Predicts** | Agent directly predicts **actions** + **values** given history. | Agent predicts **belief states** given history, and **actions** + **values** given belief states. |