Learning Environmental States via Communication

Masoumeh Heidari Kapourchali
MHDRKPRC@MEMPHIS.EDU
Bonny Banerjee
BBNERJEE@MEMPHIS.EDU
Institute for Intelligent Systems, and Department of Electrical & Computer Engineering, University of Memphis, Memphis, TN 38152 USA

Abstract

Discovering unknown states of the environment is a problem of interest in many fields. This paper investigates how an agent can refine existing states and learn new states in its environment by communicating with another agent. We simulate an environment consisting of a learning agent, a mentor agent, and a set of objects. The learning agent, implemented in a predictive coding framework, communicates with the mentor agent to resolve ambiguity regarding causes generating an observation. Both agents are modeled using human body skeletons from the UTD-MHAD dataset. Communication takes place using body gestures and movements. Our experiments show that communication helps the agent to learn: (1) environmental states that are beyond its perceptual field, and (2) the action classes related to each object without supervision or reinforcement. Over time, the latter leads to learning appropriate body movements for each state and more efficient communication.

1. Introduction

Learning unknown environmental states is necessary for an autonomous agent to function in an open-ended environment. Most existing models passively perceive their environment to learn the states. In order to learn from an unknown partially-observable and ever-changing environment, it is imperative for an agent to actively perceive its environment to sample more efficiently.

A number of models have been proposed to circumvent the limitations of passive perception or passively inferring the causes of sensory observations. Sequential decision-making frameworks are used to control sensor parameters so that the information content in data is enhanced (Bajcsy et al., 2018). Using actions to change environment and generate more informative observations is also recently explored (Bohg et al., 2017). Most existing models formulate the problem of active perception using a reinforcement learning framework which is a natural decision-theoretic approach to model agents (Satsangi et al., 2018). Reinforcement learning is known to be computationally expensive (Satsangi et al., 2018) and requires an explicit reward function to learn optimal behavior. Active inference under the free-energy principle is an alternative approach for modeling an agent which is more general and efficient, and does not require any reinforcement (Friston et al., 2012, 2009). Also, most existing work is limited to modeling a single agent with sensors in a single modality. In our previous work (Kapourchali & Banerjee, 2018a, 2019), the set of environmental
states was given; we showed that communication helps an agent overcome its sensory limitations, thereby making causal inference accurate and efficient.

This paper investigates the role of communication in learning about the states in an unknown, partially-observable and dynamic environment. We simulate an environment with two agents (learner and mentor) and a set of objects. The learner communicates with the mentor agent opportunistically to resolve ambiguity regarding causes generating an observation. The agents are modeled using human body skeletons from the UTD-MHAD dataset. Communication occurs using body gestures and movements. In our model, communication is an action that can be optimized under the free-energy principle. The model is evaluated for its effectiveness in learning the states and its efficiency of communication. Our experimental results show that due to communication, the agent can learn about unknown states of the environment, the observations due to which are beyond its perceptual field. Also, the communication is optimized over time. That is, the learner can successfully predict the mentor agent over time which leads to a reduction in the number of communications.

The rest of the paper is organized as follows. The related literature is briefly reviewed in Section 2. Section 3 introduces the necessary concepts. The problem statement and proposed model are discussed in Sections 4 and 5 respectively, followed by experimental results in Section 6.

2. Related Work

Our work draws inspiration from and contributes to multiple research directions. This section highlights these areas and explains our contribution over existing work.

2.1 Active Learning, Active Perception and Interactive Perception

Active learning has been widely researched in machine learning and data mining with reference to problems where unlabeled data is abundant but obtaining labels is difficult, time-consuming, or expensive. The key idea is that the active learner can ask an oracle (such as a human annotator) to label a subset of unlabeled instances (Cohn et al., 1996; Settles, 2009). Usually, these algorithms solve a supervised learning problem (Taniguchi et al., 2018); the goal is to learn what constitutes a valuable datapoint. Assuming that the annotator is part of the environment and the learner solves the problem of information acquisition, active learning requires sampling the data stream as in active perception but the two are fundamentally different (Taniguchi et al., 2018).

Active perception is one of the most important human cognitive abilities (Taniguchi et al., 2018) and is imperative in artificial agents (Bajcsy et al., 2018). It refers to the problem of controlling sensor parameters to maximize the information content in transmitted data (Bajcsy et al., 2018). Even though artificial intelligence (AI) has been largely involved with analyzing passively sampled data, recently there has been significant interest in extending the scope of active perception in computer vision and robotics (Bajcsy et al., 2018; Fitch et al., 2018). Active inference (Friston, 2010) addresses the challenges of active perception by minimizing long-term average of surprise or prediction error without requiring an explicit reward function. The framework has been also used for speech production (Najnin & Banerjee, 2017), self-other distinction (Kahl & Kopp, 2018) and learning sensorimotor dynamics (Butz et al., 2019).
In interactive perception (Bohg et al., 2017), action not only changes the perception but also the environment. Lesort et al. (2018) reviews the most recent works on interactive perception for the task of state representation learning (SRL). SRL is a kind of representation learning in which the learned features are in low dimension, evolve through time, and are influenced by actions of an agent. Most of these models are neural networks-based and prone to the catastrophic forgetting phenomena (Caselles-Dupré et al., 2018) where the learner forgets past knowledge as the training data distribution changes (French, 1999). Caselles-Dupré et al. (2018) proposes a continual state representation learning in which Welch’s t-test is used in conjunction with a variational autoencoder to automatically detect environmental changes. So instead of saving the observations seen before the change point, it stores the parameters learned from previously seen observations and can reconstruct them.

Our proposed model has similarities with states representation learning using interactive perception in the sense that the mentor agent is part of the environment and the learned features are in low dimension and evolve over time. In our model, both the features and the environment (mentor agent’s actions) are influenced by actions of the learning agent. Furthermore, the agent does not need to store any observation since it updates the parameters in an online manner. It does not need any statistical test and adds to the previously learned parameters if the communication messages does not converge to its expectations.

2.2 Structure Learning

Causality shapes how agents perceive from and act on the environment. Graphical models provide a powerful framework for describing causal structures in the environment. In most cases, it is assumed that the structure of the network which represents causal knowledge is given (Boyen et al., 1999). This limits their application on real-world problems with large number of states (Gershman & Niv, 2010). The problem of structure learning is concerned with learning the causal model of environment (Russell & Norvig, 2016). When data is complete and fully observable, this problem reduces to searching, dependency analysis (Cheng et al., 1998) or density estimation (Devroye, 1987). Several methods are proposed to improve efficiency of these solutions (Cheng et al., 1998; Koski & Noble, 2012; Rosset & Segal, 2003). The problem is more challenging when there are unobserved variables which may influence the observable data (Russell & Norvig, 2016). Expectation-maximization (EM) is widely used to solve this problem (Benjumeda et al., 2019). In the simplest case, an expert defines the unknown environmental variables as input to the algorithm which is infeasible in general (Friedman et al., 1997). Friedman & Koller (2003) proposed structural EM (SEM) algorithm, an extension of EM, to minimize dependence on the expert by simultaneously learning the structure and parameters of a Bayesian network (BN) from incomplete data. Convergence to a local optimum is guaranteed in this algorithm. However, since inference in BNs is NP-hard, SEM is known to be a computationally expensive algorithm (Benjumeda et al., 2019) motivating researchers to propose approximation techniques to make SEM tractable (Benjumeda et al., 2019; Scanagatta et al., 2018).

Deep learning models which are very powerful in many machine learning tasks can be used to learn multiple layers of hidden variables (i.e. causal model of environment). However, they have a large number of parameters (Nie, 2016) which require considerable computational resources. Furthermore, they need plenty of training data as they are unable to learn concepts from a few
examples by reusing previously learned conceptual knowledge in new contexts (Hay et al., 2018). Pruning (Han et al., 2015; Sun et al., 2016) and regularization (Fan et al., 2017; Lu et al., 2018; Luo et al., 2017; Ravikumar et al., 2010) techniques are proposed to reduce redundancy in the parameters of deep models. Gibbs sampling is also used in (Nie, 2016) for drawing samples from data in order to handle the intractable E-step in the EM. This method heavily relies on the quality of the samples and is applicable only to binary variables. Further improvements are also needed to adapt for dynamic and multimodal environments (Nie, 2016).

Our model can handle categorical and continuous variables. The agent starts with a few states and communicates with the mentor agent if it cannot explain its observations using the previously learned states. It influences the mentor agent’s actions through communication and adds a new state only if the communication does not help to explain the observation, i.e. the agents’ behaviors do not converge after the communication sequence. The model is learned online and can deal with dynamic changes in the environment. An important advantage of communication is that the unknown states can be captured even if the environmental data is completely ambiguous from the first agent’s point of view; that is, different causal states generate the same observation. The agent uses variational inference which is an exact approximation in contrast to Gibbs sampling or other stochastic approximation methods. The samples are chosen in an informed manner via agent’s optimal actions.

2.3 Agents’ Communication

Efficiency in agents’ communication is essential in multi-agent systems. Prior work in this field assumes the communication to be ubiquitous. In order to avoid excessive communications, periodic communication is proposed (Liu et al., 2010) where the communication protocols are independent of data and agents’ internal model. In these methods, all agents need to transmit their messages synchronously and usually a conservative sampling period for worst case situations is chosen (Garcia et al., 2016). Learning when to communicate is a fundamental problem in this area where some approaches use a metric to value the communication and then utilize a reinforcement learning technique to decide on when to communicate (Roth et al., 2005). These approaches can be of two types: offline, i.e. methods that learn communication strategy offline before the plan execution starts (Nair et al., 2004; Roth et al., 2005; Singh et al., 2018), and online (Unhelkar & Shah, 2016; Williamson et al., 2009; Wu et al., 2011).

Our model has similarity with these works in the sense that the agent decides when to communicate with the mentor agent and how long. However, it does not require a reward function or a different objective function for learning when to communicate. The agent, modeled in a predictive coding framework, follows the free energy principle which is a generalized form of reinforcement learning (Friston et al., 2009). The other difference with previous work is that our agents communicate using their body gestures and movements and cannot communicate their internal states or observations directly. Therefore, communication is an action which can influence the environment and generate new observation which allows the agent to maintain a distinct internal model and still sample from the other agent’s knowledge.
3. Preliminaries

This section introduces the relevant terms and concepts.

**Definition 1. (Agent)** An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators (Russell & Norvig, 2016). In this paper, an agent is implemented in software; it acts with body movements to communicate with other agent in the environment.

**Definition 2. (Variational free energy)** Variational free energy (VFE) is a measure of salience based on the divergence between the recognition density \( q(\Psi) \) and generative density \( p(\Phi, \Psi) \) (Friston, 2010):

\[
F = -\langle \ln p(\Phi, \Psi) \rangle_q + \langle \ln q(\Psi) \rangle_q.
\]

**Definition 3. (Recognition density)** Recognition density is a probabilistic representation of causes (environmental hidden states in this paper) which is encoded by internal states \( \mu \). If the density is assumed to be Gaussian, it is called *Laplace approximation* (Friston, 2010):

\[
q(\Psi) = \mathcal{N}(\Psi; \mu, \zeta) = \frac{1}{\sqrt{2\pi\zeta}} \exp\left(-\frac{(\Psi - \mu)^2}{2\zeta}\right).
\]

**Definition 4. (Generative density)** Generative density \( p(\Phi, \Psi) \) is a joint probability density relating environmental states and sensory data. It is usually specified in the form of a prior \( p(\Psi) \) and a likelihood \( p(\Phi|\Psi) \) (Friston, 2010).

**Definition 5. (The free-energy principle)** Under the free-energy principle, an agent minimizes VFE by perception, action and learning (Friston, 2010). It means

\[
\mu, a, m = \text{argmin} F(\Phi, \mu|m).
\]

Under Laplace approximation, \( F = -\ln p(\mu, \Phi) + C \) where \( C \) is a constant (Buckley et al., 2017).

**Definition 6. (Predictive coding)** The problem of inferring the causes from sensations is ill-posed as different causes can generate the same sensation. Predictive coding (Friston, 2010) is a brain-inspired framework for solving this problem by minimizing VFE.

Important variables used in the paper are listed in Table 1.

### Table 1: Symbols and notations.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z )</td>
<td>The learner agent’s latent states</td>
</tr>
<tr>
<td>( \Psi )</td>
<td>Environmental hidden state</td>
</tr>
<tr>
<td>( \Phi^{(e)} )</td>
<td>The agent’s observation from object</td>
</tr>
<tr>
<td>( \Phi^{(msg)} )</td>
<td>The agent’s observation from communication</td>
</tr>
<tr>
<td>( \Phi^{(e)} )</td>
<td>Reconstruction of ( \Phi^{(e)} )</td>
</tr>
<tr>
<td>( \Phi^{(msg)} )</td>
<td>Reconstruction of ( \Phi^{(msg)} )</td>
</tr>
<tr>
<td>( g_e )</td>
<td>Generative model of agent from object</td>
</tr>
<tr>
<td>( g_{comm} )</td>
<td>Generative model of agent from communication message (mentor’s body movements)</td>
</tr>
<tr>
<td>( g_a )</td>
<td>Generative model for producing actions</td>
</tr>
<tr>
<td>( \Theta_{g_e} )</td>
<td>Parameters for ( g_e )</td>
</tr>
<tr>
<td>( \Theta_{g_{comm}} )</td>
<td>Parameters for ( g_{comm} )</td>
</tr>
<tr>
<td>( \Theta_{g_a} )</td>
<td>Parameters for ( g_a )</td>
</tr>
</tbody>
</table>
4. Problem Statement

State representation learning is formulated as a mapping $f$ from the history of observations to the current states $\psi_t = f(\phi_{1:t})$ (Lesort et al., 2018), where actions $a_{1:t}$ can be added to the parameters of $f$. In this paper, we are primarily interested in the setting in which this mapping is learned without accessing the true state, cardinality of state space is not known, and data distribution is non-stationary.

5. Models and Methods

In this paper, we model an agent which is situated in an unknown partially-observable dynamic environment, and performs actions using its body to engage in nonverbal communication with its mentor agent. Communication helps to resolve ambiguities and learn unknown states by reusing previously learned concepts. The agent is modeled in a predictive coding framework (Banerjee & Dutta, 2014).

5.1 Learner Agent’s Architecture

In order to mathematically formulate the agent’s architecture, we need to define the model $m$ for our problem. Figure 1 provides a block diagram of the model. The information flow in the learner agent’s internal model is shown in the diagram. The agent is able to observe the environment in two modalities. First, the observation which is caused by hidden states of environment $\Phi^{(e)}$ and second, the communication messages $\Phi^{(msg)}$. The mentor agent’s internal model is also not observable. The agents can influence the observations $\Phi^{(msg)}$ by moving their body to generate new messages. The VFE in our model can be written as:

$$F = -\ln p(\mu, \Phi^{(e)}, \Phi^{(msg)}_{1:T}) + C = -\ln p(\Phi^{(e)}|\mu)p(\Phi^{(msg)}_{1:T}|\mu)p(\mu) + C$$

where $T$ is the maximum number of communications to reason about $\Phi^{(e)}$. The agent can infer the causes and predict observations using its generative functions, iteratively. Assuming random Gaussian noise as errors, the generative functions of environment, communication and priors can be defined as $N(\Phi^{(e)}; g_e(\mu, \Theta_{g_e}), \Sigma_{\varphi(e)})$, $N(\Phi^{(msg)}; g_{comm}(\mu, \Theta_{g_{comm}}), \Sigma_{\varphi(msg)})$ and $N(\mu; \Psi_p, \Sigma_p)$, respectively.

The generative functions ($g_\star$) map agent’s observation $\Phi$ to environmental hidden states $\Psi$ where hidden states are approximated by agents’ belief $\mu$ which is mean of recognition density (Def. 3). In our model the parameters of $g_e$ are learned using winner-takes-it-all clustering (Zhong, 2005) while parameters of $g_{comm}$ are learned using shift-invariant sparse coding (Bristow et al., 2013) which is more appropriate for streaming data (Kapourchali & Banerjee, 2018b) like body movements. The agent also has a generative function $g_a(\mu, \Theta_{g_a})$ which is used for producing actions. That is, if there is a prediction error greater than what was expected, the learner agent moves its body using $g_a$. It learns two different sets of parameters ($\Theta_{g_a}$ and $\Theta_{g_{comm}}$), one to learn about characteristics of its body (to generate actions) and the other to learn about the mentor agent in order to predict its behavior and infer the hidden state.
5.2 Mentor Agent’s Architecture

The mentor agent performs actions appropriate for each object but the actions are not unique and even the same action can be done in different ways causing prediction errors for the learner agent. Therefore, the mentor needs to 1) pay attention to the learner who initiates a communication to minimize prediction error, 2) infer what the learner is trying to perform which is not trivial since the learner may have errors in moving its body accurately; and 3) optimize its action so that it is understandable by the learner.

The object is observable to the mentor but the internal models of both agents (self and other) are not; so the mentor does not know which patterns the learner has already learned or what action it tries to perform. The mentor agent infers internal state of the learner through communication and optimize its action.

5.3 Communication Mechanism

At each time instant an object is chosen randomly. The communication iterates in a loop until the prediction error is minimized or the agent learns a new state of the environment. This problem is challenging for the agents since none of them can observe the internal models of the other agent or its observations so there is no way to verify whether an agent’s interpreted cause of another’s behavior corresponds to the latter’s actual cause or not (Friston & Frith, 2015). The best the agents can do is to invent a coherent story that minimizes all conflicts in their mind. In our model by
observing enough samples over time and updating the agents’ model of other agent in an online manner, the prediction error is decreased. If agents’ behaviours are not converged within \( T \) steps, a new environmental state and its joint probability with mentor’s behaviour is added to the learner’s known states. If the communication helps agents to solve the conflicts, the learner updates its model of the mentor which leads to learning a new behaviour for an already known state. However, if the mentor’s action was as expected from beginning, all the learner needs to do is refining previously learned parameters for all the three generative models (model of environment, mentor and its action). The procedure for the mentor is the same and the only difference is that it needs to infer only the learner’s internal state as opposed to environmental state.

6. Experimental Results

This section discusses the simulated environment and evaluation of our model.

6.1 Simulated Environment

We simulate an environment where a mentor agent helps another agent to learn about unknown states of the environment using nonverbal communication. The agents’ internal models are distinct but they can influence each other’s actions and therefore observations (\( \Phi^{(msg)} \)). The following sections define abilities of each agent as well as characteristics of environment.

6.1.1 The learner Agent

The agent is limited in its sensory-motor system. It has two sensors which can capture data in two different modalities. The first sensor can capture a noisy version of dominant color from pictures. The second sensor can capture the position of 20 joints in the Kinect skeleton data. The agent’s body is also modeled as a skeleton so it can communicate with body movements. Internal models of both agents (self and mentor) are also hidden to the agent.

6.1.2 The Mentor Agent

The mentor agent acts differently with respect to each object. The same action (e.g. walking) can be done in different ways and also for each object it can choose different actions (e.g. throwing or catching a ball). The mentor agent’s actions can be influenced by the learner agent’s communication.

6.1.3 The Environment

The two agents and a set of objects are situated in the environment. The set of objects and how the agent can see them are shown in Figure 2. The mentor agent’s actions for each object are in the Table 2. The learner agent initiates communication if it cannot explain the mentor’s action using previously learned concepts. The communication iterates several times for each sample so the agents can refine their movements to reduce prediction errors. We consider the maximum number of communications \( T = 20 \) but the learner can decide to terminate the communication if the mentor’s action is as expected.
Learning Environmental States via Communication

Figure 2: Actual objects in the environment (top) and the agent’s sensory observation for each object (bottom)

Table 2: Set of mentor agent’s actions for each object.

<table>
<thead>
<tr>
<th>Objects</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Door</td>
<td>1. Push, 2. Knock</td>
</tr>
<tr>
<td>Tennis ball</td>
<td>1. Tennis swing, 2. Tennis serve</td>
</tr>
<tr>
<td>Walkway</td>
<td>1. Jog, 2. Walk</td>
</tr>
<tr>
<td>Chair</td>
<td>1. Sit to stand, 2. Stand to sit</td>
</tr>
<tr>
<td>Eraser</td>
<td>1. Swipe left, 2. Swipe right, 3. Wave</td>
</tr>
</tbody>
</table>

The body skeleton movements of mentor agent are taken from Kinect skeleton data in the UTD-MHAD dataset (Chen et al., 2015) where each action is repeated four times. There are eight subjects in the dataset performing 27 actions. After removing three corrupted sequences, the dataset includes 861 sequences. To evaluate the model, the learner agent’s body characteristics (e.g. position of joints and the primary movements) are taken from data of odd subjects while the even subjects are considered as mentor agents (e.g. the second subject is mentor for the agent who is embodied in first subject’s skeleton). The results are averaged over the four cases. One object is presented in the environment at each time instant.

6.1.4 Evaluation of the Model

Without paying attention to the mentor, the agent can learn four unknown states in the environment which are shown in the Figure 3, middle. Agent’s representation of the objects is shown in the Figure 3, left. It can be seen that there is no way for the agent to distinguish between objects with the same color so there is ambiguity regarding causes generating the observation from learner’s point of view. Figure 3, right shows the learned states after communication. A sample frame from
some instances of the learned patterns of mentor agent’s actions is shown near each state in Figure 4. It separates the state based on the communication messages receives from mentor.

Figure 5 shows a sample scenario of agents’ communication. In the initial steps of learning, the mentor draws an \( X \) using the \textit{Marker} so the agent learns this action when it observes something \textit{White}. Next time, for the same color, the mentor performs \textit{Jog}, Figure 5, top. In this case, it is assumed that the mentor agent has performed jogging by mistake. The agent is surprised and shows what action it was expecting the mentor agent performs (Figure 5, middle). Even though the agent could not draw a perfect \( X \), the mentor realized the pattern and corrected its action (bottom row).
Learning Environmental States via Communication

![Sample Scenario of Communication](image)

Figure 5: A sample scenario of communication. The mentor corrects its action via communication. The mentor Jog (top subfigure) when the object is a Marker, the agent shows what it was expecting to see (middle) and the mentor corrects its action by drawing an X (bottom).

The figures are sample frames which are skipped by steps of four. However, if the object was the walkway, the mentor agent would repeat jogging or walking so that after maximum number of allowed communication (20 in our implementation), the agent will have to add a new state since the mentor’s actions did not converge to its expectations. On the other hand, if the mentor performs Draw circle first, after seeing the learner’s observation, it performs Draw X. If this scenario repeats in multiple trials, the first agent learns that Draw X and Draw circle are two different actions, and can be performed in the same state.

Figure 6 shows the average prediction error and number of communications for the Chair and Eraser versus number of times these objects are shown to the agent. It can be seen that only first time the agents communicate 20 times and since the actions do not converge to expectations, the learner learns it as a new state. Over time it learns different actions associated with the object so prediction error and number of communication are reduced.

Figure 7 shows similarities of the learned actions with the actions of the learner agent in the UTD-MHAD dataset. It is worth noting that the actions are not learned from data of same subject but from the mentor. The results show that the agent could successfully learn variants of actions for an object. Similarity is measured using cross-correlation and averaged for all 3-D joints in the skeleton.

7. Conclusions

This paper investigated how an agent which is limited in its sensory-motor system, can refine existing states and learn new states in its environment by communicating with another agent. In an environment consisting of a learning agent, a mentor agent, and a set of objects, the learning agent, implemented in a predictive coding framework, communicates with the mentor agent to resolve ambiguity regarding causes generating an observation. Our experiments show that using communication the agent can learn environmental states which are beyond its perceptual field. It also learns
Figure 6: Prediction errors and number of communications is decreased when the agent observes more samples of the same object.

a variety of actions related to each object and appropriate body movements for each state. Learning more actions adds to efficiency of communication since the agent does not need to communicate for an already learned state.

References


Figure 7: Similarity between the actions learned from mentor for each object and the actions from UTD-MHAD dataset.


