High-level Learning

Jane Li

Assistant Professor
Mechanical Engineering Department, Robotic Engineering Program
Worcester Polytechnic Institute
Quiz 10 (pts)

• (3 pts) Describe the training process when using decision tree to learn high-level policy

• (3 pts) Describe how to use Bayesian method to learn policy?

• (2 pts) In high-level learning, how to represent a plan?

• (2 pts) How to perform interactive correction in plan learning?
Training process

- Learn state transition function as hierarchical classifier over features
Bayesian Method [7]

- Observe and memorize a sequence of states (sub-goals)
- Pick up the action that maximizes the chance of taking the agent from current state to the memorized next state

$$a^* = \arg\max_a p(S_t \rightarrow S_{t+1})$$
How to represent a plan?

- Bayesian models
- Finite-state Automaton
Interactive correction

• Re-do parts of the demo
• Re-segment old data
• Add new corrections and rebuild FSA
Learning task objectives
Learning task objectives

- Compare states before and after an action to infer the task goal

- Infer the reward function
  - Reward states similar to those in human demonstrations
  - Segment demonstrations into sub-skills with pre-defined goals
  - Derive explicit reward function (IRL)

- Extended application
  - Use learned objective function to guide future motion planning
Create a Mind-reading Social Robot [11]

• How to infer the mental states of a human partner from their observable behavior?
  • Beliefs, intents, and desires
Cognitive Architecture: Perception System
Cognitive Architecture: Motor System

- Body
  - Desired Body Pose
  - Mapped Body Pose
- Motors
- Sensors
- Perspective Transformation
- Perception System
  - "true"
  - "any-utterance"
  - "down"
  - "up"
- Other's Perception
- Robot's Perception
- Intention System
  - Schema Recognition and Execution
  - Task Learning
    - Human
    - Hypothesis
    - Self
    - Hypothesis
- Belief System
  - Other's Beliefs
  - Robot's Beliefs
Cognitive Architecture: Belief System

Body
- Desired Body Pose
- Mapped Body Pose

Motors
- Generation

Sensors
- Recognition/Inference (Mindreading)

Motor System
- Desired Movement
- Matched Movement

Intention System
- Schema Recognition and Execution

Perspective Transformation

Perception System
- Other's Perception
- Robot's Perception

Belief System
- Other's Beliefs
- Robot's Beliefs
Cognitive Architecture: Intention System

Body

Desired Body Pose

Motor System

Ready

Point

Desired Movement

Matched Movement

Recognition/Inference (Mindreading)

Generation

Perspective Transformation

Motors

Sensors

Perception System

"true"

"any-utterance"

"down"

"sit"

Belief System

Other's Perception

Robot's Perception

Intention System

Schema: Recognition and Execution

Task Learning

Hypothesis: Self, Hypothesis: Other

Perspective: A, Perspective: B

Beliefs: Other, Beliefs: Robot
Build Robot’s Model of Human Belief

Data From Sensors

- Human 1 (50,100,0)
- Object $O_1$ (0,50,0)
- Object $O_2$ (50,150,0)

Updated Robot Belief System

Transform and Rotate data to human coordinate frame using human location in Robot's Beliefs

Transformed Data From Sensors

- Human 1 (0,0,0)
- Object $O_1$ (50,50,0)
- Object $O_2$ (0,-50,0)

Final Input to Human 1’s Belief System

- Self (0,0,0)
- Object $O_1$ (50,50,0)
Inferring Intent from Observed Behavior

- **Have**
  - **Grab**
    - **PARAMS:**
      - **OBJECT:** Cookies
    - **Parameter Mapping:**
      - Object: Box
      - Dispenser: DA
  - **Unobstructed**
    - **Dispense**
      - **Open**
        - **PARAMS:**
          - **BOX:** BOX A
          - **DISPENSOR:** DA
        - **Parameter Mapping:**
          - Box: Lock
      - **In Stock**
    - **Unlocked**
      - **Unlock**
        - **PARAMS:**
          - **LOCK:** Lock A

- **Inter-Context Mapping**:
  - **Context1**
    - **Param1:**
    - **Context2**
      - **Param2:**
  - **Parameter Generation:**
    - **Param1:**
    - **Human Activity:**
Generating goal-directed behavior

- **Have**
  - **Grab**
    - **Unobstructed**
      - **Open**
        - **Dispense**
          - **In Stock**
          - **Unlocked**
            - **Unlock**

**Parameter Mapping**:
- **OBJECT**: Cookies
- **BOX**: BOX A
- **DISPENSER**: DA
- **LOCK**: Lock A

**Parameter Mapping**:
- **Box**, **Dispenser**
- **Box**, **Lock**
Learning model and reward function for a pole-balancing task [12]

- Human demonstration

- Robot Learning
  - Reproduce human hand trajectory – fail
  - Mimic human response to each situation – policy learning
  - Learn a task model and an optimization criterion

\[
x_{k+1} = f(x_k, u_k)
\]

\[
C = \sum_k (x_k, u_k, k)
\]
Learned task model and optimization criterion

- Dynamics of balancing inverted pendulum
  \[ \dot{\theta}_{k+1} = 0.0051x_k + 0.0058\dot{x}_k + 0.47\theta_k + 0.997\dot{\theta}_k + 0.052\ddot{x}_k \]

- Learned from 30 sec demonstration data

- Step cost to minimize
  \[ r(x, u, k) = 125x^2 + 50\dot{x}^2 + 1200\theta^2 + 25\dot{\theta}^2 + 1.5\ddot{x}^2 \]
Learned poll-balancing
Further improvement using RL [13]

GMM/GMR

Probabilistic Encoding and Generalisation

Kinesthetic Demonstration

Dynamical System

Execution by the Robot

Reinforcement Learning

Task Fails

Metric

Natural actor-critic (NAC) algorithm

Close enough to target?

(NAC)
Constructing skill tree (CST) [14]
Merging two skill chains into a skill tree

(a)  
(b)  
(c)  
(d)
Demonstration
Inverse Reinforcement Learning [15]

• Generate explicit reward function
  • Hand-engineer a set of reward features
  • Define reward function as a linear combination of these features
  • Learning = learn feature coefficients from demonstration

Apprenticeship Learning

• Bayesian Inverse Reinforcement Learning
LEArning to seaRCH (LEARCH)
Learning object affordance
Learning object affordance

- Affordance
  - Properties of the environment that afford a certain action to be performed
  - Enable the user to categorize objects by their functions
  - A compact and useful representation of manipulation skills
Learning affordance

• Exploration
  • Act on object and observe the reaction → correspondence

• Visually observing human or other robot
  • Manipulating tasks
  • Full-body environmental interaction
Bayesian Network

• Learn structure using Markov Chain Monte Carlo (MCMC)

• Estimate parameters for each node

• Resulting model
  • Interpret the effects of observed actions
# Bayesian Network

![Bayesian Network Diagram]

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Action</td>
<td>grasp, tap, touch</td>
</tr>
<tr>
<td>H</td>
<td>Height</td>
<td>discretized in 10 values</td>
</tr>
<tr>
<td>C</td>
<td>Color</td>
<td>green1, green2, yellow, blue</td>
</tr>
<tr>
<td>Sh</td>
<td>Shape</td>
<td>ball, box</td>
</tr>
<tr>
<td>S</td>
<td>Size</td>
<td>small, medium, big</td>
</tr>
<tr>
<td>V</td>
<td>Object velocity</td>
<td>small, medium, big</td>
</tr>
<tr>
<td>HV</td>
<td>Hand velocity</td>
<td>small, medium, big</td>
</tr>
<tr>
<td>Di</td>
<td>Object-hand velocity</td>
<td>small, medium, big</td>
</tr>
<tr>
<td>Ct</td>
<td>Contact duration</td>
<td>none, short, long</td>
</tr>
</tbody>
</table>
**Bayesian Network**

- **Demo 1**
  - A tap on a small ball resulting in high velocity and medium hand–object distance

- **Demo 2**
  - A grasp on a small square resulting in small velocity and small hand–object distance

<table>
<thead>
<tr>
<th>obj \ action</th>
<th>grasp</th>
<th>tap</th>
<th>touch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue, big, ball</td>
<td>0.00</td>
<td>0.20</td>
<td>0.00</td>
</tr>
<tr>
<td>Yellow, small box</td>
<td>0.00</td>
<td>0.06</td>
<td>0.00</td>
</tr>
</tbody>
</table>
Detection of objects in human action sequences

(a) Ground truth: book  (b) Ground truth: magazine  (c) Ground truth: box

(d) Ground truth: box+pitcher  (e) Ground truth: cup+pitcher  (f) Ground truth: cup+pitcher

= book, = magazine, = hammer, = box, = cup, = pitcher.
Detection of objects in human action sequences

- Diagram with nodes and edges labeled:
  - $x^a_1$, $x^a_2$, $x^a_3$
  - $a_1$, $a_2$, $a_3$
  - $o_1$, $o_2$, $o_3$
  - $x^o_1$, $x^o_2$, $x^o_3$

- Table with three columns for actions:
  - Open
  - Hammer
  - Pour

- Actions with corresponding probability values:
Learning affordance in full-body environmental interactions – FOCUS algorithm

- Model *inanimate* objects in the environment by *structural* and *functional* definitions
  - Structural definition = Capture a simple and generalized visual definition of an object by feature detection
  - Functional definition = Capture object *affordance* properties

- Object classification
  - Recognize an object by associate an observed action with a particular environmental feature
Process of learning visual features

[Diagram showing the process of learning visual features]

1. Camera -> Raw Video
2. Raw Video -> Feature Detection
3. Features -> Object Grab and Generalize
4. Object Class -> Object Class Library
5. Activity Recognition
6. People's Positions
7. Activity

Instructor: Jane Li, Mechanical Engineering Department & Robotic Engineering Program - WPI
Examples
Learning task features
Learning task features

- Feature selection
- Eliminate redundant and irrelevant features
Learning frame of reference
Learning frame of reference

- Five possible frames
Assignment 15 (30 pts) – Due Dec 4

• Read
  • Section 5.6 Learning frame of reference

• Prepare 4-6 presentation slides

• To reflect your understanding
  • Add notes to your presentation slides, or
  • Submit 2-page review
Assignments 15 (30 pts) – Due Dec 4

- Grading and reward policy
  - 4 best work: grade = 100% for this assignment
  - Select 2-4 for student presentation: replace a low-grade assignment or quiz with 100%
Extra Assignment (50 pts) – Due Dec 11

• Read
  • Chapter 7 in Robot learning from human teachers

• Prepare 20 presentation slides

• To reflect your understanding
  • Add notes to your presentation slides, or
  • Submit 2-page review
Reference


Reference


