Artificial Intelligence

- Goal:
  - Understand how intelligent behavior is possible
  - i.e. Come up with a theory that explains intelligent behavior

- Methodology:
  - Design, build and experiment with computational systems that perform tasks commonly viewed as intelligent

Flying Analogy

First approach:
+ Dissect known flying animals
+ Figure out what they have in common
+ Flapping of wings made of some structure covered with feathers

A better approach:
+ Understand principles of flight
+ Don't restrict to just natural occurrences of flight
+ Construct objects that embody hypothesized principles
Limitation of Mapping Approach

- Not enough training data
- Might be interested in the internals of the model
- Might need to incorporate well-known knowledge
- Might want to understand the decision process
- Could be interested in the structure of the model
- No enough training data

Overview

- Books are physical things, and physical things have a weight
- Trump's wives are Ivana, Marla, and Melania
Declarative Approach: Reasoning Algorithm

- Rules encode how new symbols are created from existing ones.
- From rules and facts, we should be able to make conclusions that follow from internal state.

- Facts that are not explicitly represented.
- Assumptions that seem reasonable.
- Plans of actions.
- Action to perform right now.

Declarative Approach: Rules

- Not only can facts be represented with symbols, but also more general knowledge can be represented.
- Examples:
  - having cereal means having food.
    \[ \text{have(cereal)} \rightarrow \text{have(food)} \]
  - if X is connected to Y and there is a path from Y to Z, then there is a path from X to Z.
    \[ \text{connected}(X,Y) \land \text{path}(Y,Z) \rightarrow \text{path}(X,Z) \]
  - Use tokens that start with an uppercase letter for variables.
    - if X is connected to Y, then there is a path between them.
      \[ \text{connected}(X,Y) \rightarrow \text{path}(X,Y) \]

Declarative Approach: Symbols

- Model knowledge with symbols.
  - Symbols will have meaning to us (us = the designer).
  - Meaning of symbols should be unambiguous, unlike English.
- Need to express complex relations with minimum of symbols.
  - Need language for representing the internal state.
- Example:
  - \[ \text{have(milk)} \land \text{have(cereal)} \land \text{want(sugar)} \]
    - \( A \land B \) means both \( A \) and \( B \) are true for the agent.
    - \( \text{have}(X) \) means agent has \( X \) in its possession.
    - \( \text{want}(X) \) means agent wants \( X \) in its possession.
  - Don't need a symbol such as \text{havemilkhavecerealwantsugar}.
Overview

- Agent Approach
- Symbolic Reasoning

Example Problems

Fundamental Issues

- What are good languages for representing the facts of an agent's internal state?
- What are good languages for representing the rules that define the agent's reasoning?
- What are good algorithms that can produce the conclusions that correspond to reasoning?
- What do we mean by a 'good' algorithm?
- What constraints are needed on the language that allow good algorithms?

Declarative Approach

- Intelligence is in:
  - Having an appropriate language for representing internal state
  - Being able to reason about symbols to form new symbols

Knowledge engineer:
- Decides the set of facts and rules for a particular domain

Programmer:
- Constructs algorithms that can take arbitrary sets of facts and rules to make conclusions
- Can reuse algorithms over and over again for any domain
- Knowledge engineer:
  - Provides the set of facts and rules for a particular domain
  - Has an appreciation for representing internal state
  - Has an appreciation for representing internal state

Inference is in:

- Bookkeeping
- Example Problems
- Symbolic Reasoning
- Agent Approach
Overview

Agent Approach
Symbolic Reasoning
Example Problems

Wiring

- Needs to know
  - What each device is
  - What is connected to what
  - Whether each switch is on
  - Whether each circuit breaker is on
  - Whether each light is on

- Needs to reason about
  - How to turn on a light
  - Whether there is a fault in the system
    + And where it is likely to be

- What is a good way of representing that knowledge?
  - How do we specify what we want to reason about?
  - How do we specify what we want to reason about?

Delivery Robot

- Needs to know
  - Layout of space
  - Where things are
  - Which doors are open

- Needs to reason about
  - How to get from one point to another
  - Delivery robot needs to know

- What is a good way of representing that knowledge?
  - How do we specify what we want to reason about?
Tcl versus Python

- I've used Tcl in the past for homeworks
  - Tcl is not as commercially used as Python
  - Tcl has some idiosyncrasies
    + Very picky about tokening
    + Hard to differentiate between a token and a list with one token
- Most of you already know Python
  - Can do homeworks in Tcl or Python
  - If you don't have a lot of programming experience, use Python
  - If you want to minimize time spent on homeworks, use Python
  - If you know Python really well, and want to add another programming language to your Resume, use Tcl

Two versions of each homework: Tcl and Python versions

Grading

- Assignments 50%
- Midterm 25%
- Final 25%

Course website: cslu.ohsu.edu/~heeman/cs560

Slack channel

Course Outline

1. Knowledge and Reasoning
   - Datalog: Syntax, Semantics, Inference
   - Search procedures
   - Representing knowledge
     - Richer formalisms
   - Reasoning about Equality
   - Integrity constraints
   - Disjunctive knowledge
   - Quantification
   - First order predicate logic

2. Actions & Planning
   - Agents might have goals, have knowledge about actions
   - Non-monotonic Reasoning
     - Making assumptions and learning new information
   - Belief and Knowledge
     - Representing and reasoning about beliefs/knowledge of other agents

3. Building Agents
   - Tie together concepts into a system
Academic Integrity

- You can do the homework with your colleagues, but you cannot bring any part of your homework into the meeting.
- After the meeting, you rehash the solution from scratch. What is the point of taking the course?
- Corollary: Unless both people have photographic memories, homework assignments should look different.

Critique

- Answer key given out via Sakai when you submit your answers. On honor system not to share it, nor post questions about the answers.
- Have until Sunday at 11:55pm to submit a critique. Explain what you did wrong, and why you made that mistake. Worth up to half the marks that you lost. Should show that you reviewed and understood answer key and understood whether your answer was correct. See sample homework for how to format this.

Homework

- Homework usually given out Monday by 11:55pm. Should have it done by Friday at 9:00pm. But not officially due till Sunday at 9:00pm. Can have a 2 day extension on one homework. Homework must be submitted through Sakai.
- Homework must be submitted through Sakai. If you have problems email me at heemanp@ohsu.edu. Single pdf with your answers on it. Sample homework on the website.
- Homework should include relevant code as well if it is algorithmic and works.
- You need to typeset it if it is algorithmically and works.
- Submit your code as a single python file and prolog file.
We will be following the textbook closely.

You are responsible for material in the textbook.

Reading assignments are posted on the course website.

Read chapter 1, 2.1-2.5 for next class.

Reading Assignments:

- Read chapter 1, 2.1-2.5 for next class.