CS 2740 Knowledge representation
Lecture 1

Knowledge representation

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

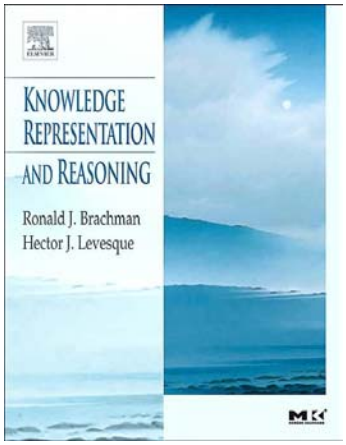
Course administrivia

Instructor: Milos Hauskrecht
5329 Sennott Square
milos@cs.pitt.edu

TA: just assigned (Jiang Zheng ?)

Course web page:
http://www.cs.pitt.edu/~milos/courses/cs2740/

Textbook



Brachman, Levesque.
Knowledge Representation and Reasoning.
Morgan Kaufman, 2004

Grading

- **Lectures** **10%**
- **Homework assignments** **30%**
- **Exams** **30%**
- **Final project** **30%**

Lectures

- **10 % of the grade**
- **Attendance + short quizzes**
- **Short quizzes:**
  - 10 minutes at the beginning of the lecture
  - Random
  - Short question(s) from previous lectures

Homework assignments

- **Homework assignments:**
  - 30 % of the grade
  - Weekly / Biweekly assignments
  - A mix of pencil and paper, and programming assignments
  - No extensions. Homework due dates are strict.
- **Collaborations:**
  - No collaborations on homework assignments
- **Programming language:**
  - your choice
  - AI programming language like Scheme, Python or Lisp are encouraged

Exams

- **Midterm and Final exam**
  - 15 % of the grade each
  - In-class

Academic honesty

- All the work in this course should be done independently.
- **Collaborations on homework assignments, quizzes and exams are not permitted.**
- Cheating and any other anti-intellectual behavior, including giving your work to someone else, will be dealt with severely.

- **Academic Integrity Code for the Faculty and College of Arts and Sciences:**
  - **http://www.fcas.pitt.edu/academicintegrity.html**

Knowledge representation

Knowledge representation

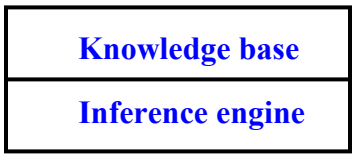
- **Knowledge representation (KR)** is
  - the study of how knowledge about the world can be represented and what kinds of reasoning can be done with that knowledge.
- **Important questions include the tradeoffs:**
  - representational adequacy,
  - representational fidelity,
  - computational cost of related inferences,
  - representation of default, commonsense, or uncertain probabilistic information.

Knowledge representation

We want a representation that is:

- rich enough to express the knowledge needed to solve the problem
- as close to the problem as possible: compact, natural and maintainable, amenable to efficient computation
- able to express features of the problem we can exploit for computational gain
- able to trade off accuracy and computation time

Knowledge-based agent



- **Knowledge base (KB):**
  - A set of sentences that describe facts about the world in some formal (representational) language
  - **Typically domain specific** but large knowledge corpuses are built to provide general knowledge resource (Cyc)
- **Inference engine:**
  - A set of procedures that use the representational language to infer new facts from known ones or answer a variety of KB queries. Inferences typically require search.
  - **Typically domain independent**

Example: MYCIN

- MYCIN: an expert system for diagnosis of bacterial infections
- **Knowledge base** represents
  - Facts about a specific patient case
  - Rules describing relations between entities in the bacterial infection domain

If	1. The stain of the organism is gram-positive, and 2. The morphology of the organism is coccus, and 3. The growth conformation of the organism is chains
Then	the identity of the organism is streptococcus

- **Inference engine:**
  - manipulates the facts and known relations to answer diagnostic queries (consistent with findings and rules)

Knowledge representation

- The objective of knowledge representation is to express the knowledge about the world in a computer-tractable form

- Key aspects of knowledge representation languages:
  - **Syntax:** describes how sentences are formed in the language
  - **Semantics:** describes the meaning of sentences, what is it the sentence refers to in the real world
  - **Computational aspect:** describes how sentences and objects are manipulated in concordance with semantical conventions

Many KB systems rely on some variant of logic

Topics

- Propositional logic and inference
- First order logic and inference
- Frame-based representations
- Inheritance and Defaults
- Ontologies and commonsense knowledge
- Semantic Web
- Modeling Uncertainty
- Probabilistic Logic

Knowledge representation

Many different ways of representing the same knowledge.
Representation may make inferences easier or more difficult.

Example:

- How to represent: "Car #12 is red."

  Solution 1: ?

Knowledge representation

Many different ways of representing the same knowledge.
Representation may make inferences easier or more difficult.

Example:

- How to represent: "Car #12 is red."

  Solution 1: Red(car12).

  – It's easy to ask "What's red?"

  – But we can't ask "what is the color of car12?"

  Solution 2: ?

Knowledge representation

Many different ways of representing the same knowledge.
Representation may make inferences easier or more difficult.

Example:

- How to represent: "Car #12 is red."

  Solution 1: Red(car12).

  – It's easy to ask "What's red?"

  – But we can't ask "what is the color of car12?"

  Solution 2: Color (car12, red).

  – It's easy to ask "What's red?"

  – It's easy to ask "What is the color of car12?"

  – Can't ask "What property of pen7 has value red?"

  Solution 3: ?

Knowledge representation

Many different ways of representing the same knowledge.
Representation may make inferences easier or more difficult.

Example:

- How to represent: "Car #12 is red."

  Solution 1: Red(car12).
  - It's easy to ask "What's red?"
  - But we can't ask "what is the color of car12?"

  Solution 2: Color (car12, red).
  - It's easy to ask "What's red?"
  - It's easy to ask "What is the color of car12?"
  - Can't ask "What property of pen7 has value red?"

  Solution 3: Prop(car12, color , red).
  - It's easy to ask all these questions.

Knowledge representation

- Prop(Object, Property, Value)
- **Called:** object-property-value representation

- Triplets can be represented by a semantic network
- If we merge many properties of the same object we get the frame-based (object-centered) representation:

  Prop(Object, Property1, Value1)
  Prop(Object, Property2, Value2)
  …
  Prop(Object, Property-n, Value-n)

Ontology

If more than one person is building a knowledge base,
they must be able to share the conceptualization.

- A conceptualization is a mapping from the problem domain into the representation.
- A conceptualization specifies:
  - What types of objects are being modeled
  - The vocabulary for specifying objects, relations and properties
  - The meaning or intention of the relations or properties
- An ontology is a specification of a conceptualization.

Commonsense knowledge

- Our ability of answering questions intelligently relies heavily on general knowledge about the world
- General knowledge about the world and relations that hold in the world is referred to as **commonsense knowledge**
- **Commonsense knowledge**
  - a very large corpus of knowledge
  - helps us to understand things like:
    - A pen can fit in the box
    - A box can fit in the pen
- **Challenge:** representation of commonsense knowledge that allows us to answer queries and make inferences
  - Recent advances: Cyc project

Cyc project

- Cyc is the world's largest and most complete general knowledge base and commonsense reasoning engine.
  - 15000 relations
  - 300000 concepts
  - 3200000 assertions
  - Temporal relations: 37

OpenCyc is the open source version of the Cyc technology. OpenCyc contains the full set of (non-proprietary) Cyc terms as well as millions of assertions about the. Cycorp offers this ontology at no cost and encourages you to make use of it as you see fit.