Knowledge representation

Artificial Intelligence

- The field of **Artificial intelligence**:
  - The design and study of computer systems that behave intelligently
- **AI programs:**
  - Go beyond numerical computations and manipulations
  - Focus on problems that require reasoning (intelligence)
  - and often a great deal of **knowledge** about the world
- Success in solving the problems depends naturally on our ability to:
  - Represent the knowledge about the world
  - Reason with the knowledge to obtain meaningful answers

Knowledge representation

- **Knowledge representation (KR)** is the study of
  - how knowledge and facts about the world can be represented, and
  - what kinds of reasoning can be done with that knowledge.

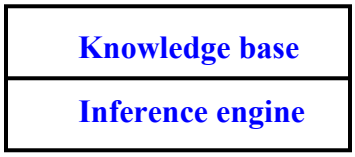
- **Important KR questions one has to consider:**
  - representational adequacy,
  - representational quality,
  - computational cost of related inferences,
  - representation of default, commonsense, or uncertain information.

Knowledge representation: goals

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 the world and its behavior in some formal (representational) language
  - **Typically domain specific** but large knowledge corpuses are built to provide general knowledge resources (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 languages

- Goal: 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

Tentative topics

- **Introduction**
- **AI programming languages - LISP**
- **Propositional logic and inference**
- **First order logic and inference**
- **Extensions of PL and FOL:**
  - Semantic networks, Frame-based representations
  - Inheritance and Defaults
  - Ontologies/Semantic Web
  - Modeling time
- **Planning and acting:**
  - Situational calculus
  - STRIPS
  - …

Tentative topics

- **Modeling Uncertainty**
  - Extensional models
  - Probabilistic models
  - Bayesian belief networks
  - Markov processes
- **Decision-making in the presence of uncertainty**
  - Decision trees
  - Markov decision processes

AI programming languages

Focus on symbolic processing

Special AI Languages:

- **LISP (since 1956)** 
  - Symbolics machines – in 80s, special LISP processors – LISP functions hardwired
- Prolog
- Smalltalk
- Python

- **Nowadays:**
  - C
  - Java

Logic

- Many knowledge representation systems rely on some variant of logic, e.g.:
  - **Propositional logic**
  - **First order logic**
  - **Temporal logic**
- And variety of extensions

Logic defines:

- **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

Propositional logic

- **Simplest type of logic**

- A **proposition** is a statement that is either true or false
- **Examples:**
  - Pitt is located in the Oakland section of Pittsburgh.
  - It is raining today.
- **More complex sentences:**
  - It is raining outside and the traffic in Oakland is heavy.


It is raining outside $\wedge$ the traffic in Oakland is heavy

First order logic

- **More complex: objects, relations, properties are explicit**

- **Examples:**
  - Red(car12)
  - Brother(Peter, John)

- **More complex sentences:**

$$\forall x, y \; parent(x, y) \Rightarrow child(y, \mathrm{x})$$

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 car12 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 car12 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

- If we merge many properties of the same object we get the **frame-based** (object-centered) representation:

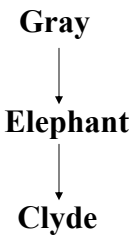
  Prop(Object, Property1, Value1)
  Prop(Object, Property2, Value2)
  …
  Prop(Object, Property-n, Value-n)

Knowledge representation

- **Inheritance**
- Properties are inherited from more general concepts

Example:

- Clyde is an Elephant & Elephant is Gray,

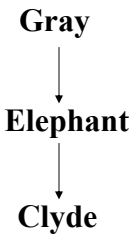


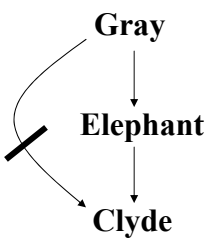
Knowledge representation

- **Inheritance**
- Properties are inherited from more general concepts

Example:

- Clyde is an Elephant & Elephant is Gray & Clyde is not grey





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.

Topics

- **Planning and acting:**
  - Situational calculus
  - STRIPS
- **Modeling Uncertainty**
  - Extensional models
  - Probabilistic models
  - Bayesian belief networks
  - Markov processes
- **Decision-making in the presence of uncertainty**
  - Decision trees
  - Markov decision processes