Artificial Intelligence
4. Knowledge Representation

Representation

- AI agents deal with knowledge (data)
  - Facts (believe & observe knowledge)
  - Procedures (how to knowledge)
  - Meaning (relate & define knowledge)
- Right representation is crucial
  - Early realisation in AI
  - Wrong choice can lead to project failure
  - Active research area

Choosing a Representation

- For certain problem solving techniques
  - ‘Best’ representation already known
  - Often a requirement of the technique
  - Or a requirement of the programming language (e.g. Prolog)
- Examples
  - First order theorem proving… first order logic
  - Inductive logic programming… logic programs
  - Neural networks learning… neural networks
- Some general representation schemes
  - Suitable for many different (and new) AI applications

Some General Representations

1. Logical Representations
2. Production Rules
3. Semantic Networks
   - Conceptual graphs, frames
4. *Description Logics* (see textbook)

What is a Logic?

- A language with concrete rules
  - No ambiguity in representation (may be other errors!)
  - Allows unambiguous communication and processing
  - Very unlike natural languages e.g. English
- Many ways to translate between languages
  - A statement can be represented in different logics
  - And perhaps differently in same logic
- **Expressiveness** of a logic
  - How much can we say in this language?
- Not to be confused with logical reasoning
  - Logics are languages, reasoning is a process (may **use** logic)

Syntax and Semantics

- Syntax
  - Rules for constructing legal sentences in the logic
  - Which symbols we can use (English: letters, punctuation)
  - How we are allowed to combine symbols
- Semantics
  - How we interpret (read) sentences in the logic
  - Assigns a meaning to each sentence
- Example: “All lecturers are seven foot tall”
  - A valid sentence (syntax)
  - And we can understand the meaning (semantics)
  - This sentence happens to be false (there is a counterexample)

Propositional Logic

- Syntax
  - Propositions, e.g. “it is wet”
  - Connectives: and, or, not, implies, iff (equivalent)

    $\wedge \quad \vee \quad \neg \quad \longrightarrow \quad \longleftrightarrow$
  - Brackets, T (true) and F (false)
- Semantics (Classical AKA Boolean)
  - Define how connectives affect truth
    - “P and Q” is true if and only if P is true and Q is true
  - Use **truth tables** to work out the truth of statements

Predicate Logic

- Propositional logic combines atoms
  - An atom contains no propositional connectives
  - Have no structure (today_is_wet, john_likes_apples)
- **Predicates** allow us to talk about objects
  - Properties: is_wet(today)
  - Relations: likes(john, apples)
  - True or false
- In predicate logic each atom is a predicate
  - e.g. first order logic, higher-order logic

First Order Logic

- More expressive logic than propositional
  - Used in this course (Lecture 6 on representation in FOL)
- **Constants** are objects: john, apples
- **Predicates** are properties and relations:
  - likes(john, apples)
- **Functions** transform objects:
  - likes(john, fruit_of(apple_tree))
- **Variables** represent any object: likes(X, apples)
- **Quantifiers** qualify values of variables
  - True for all objects (Universal): $\forall$X. likes(X, apples)
  - Exists at least one object (Existential): $\exists$X. likes(X, apples)

Example: FOL Sentence

- "Every rose has a thorn"

$$\forall X.(rose(X) \rightarrow \exists Y.(has(X, Y) \wedge thorn(Y)))$$

- For all X
  - if (X is a rose)
  - then there exists Y
    - (X has Y) and (Y is a thorn)

Example: FOL Sentence

- "On Mondays and Wednesdays I go to John's house for dinner"

$$\forall X.((is\_mon(X) \vee is\_wed(X)) \rightarrow$$
$$eat\_meal(me, houseOf(john), X))$$

- Note the change from "and" to "or"
  - Translating is problematic

Higher Order Logic

- More expressive than first order
- Functions and predicates are also objects
  - Described by predicates: binary(addition)
  - Transformed by functions: differentiate(square)
  - Can quantify over both
- E.g. define red functions as having zero at 17

$$\forall F.(red(F) \leftrightarrow F(0) = 17)$$

- Much harder to reason with

Beyond True and False

- Multi-valued logics
  - More than two truth values
  - e.g., true, false & unknown
  - **Fuzzy logic** uses probabilities, truth value in [0,1]
- Modal logics
  - Modal operators define mode for propositions
  - **Epistemic logics** (belief)
    - e.g. $\Box$p (necessarily p), $\Diamond$p (possibly p), …
  - **Temporal logics** (time)
    - e.g. $\Box$p (always p), $\Diamond$p (eventually p), …

Logic is a Good Representation

- Fairly easy to do the translation when possible
- Branches of mathematics devoted to it
- It enables us to do logical reasoning
  - Tools and techniques come for free
- Basis for programming languages
  - Prolog uses logic programs (a subset of FOL)
  - λProlog based on HOL

Non-Logical Representations?

- Production rules
- Semantic networks
  - Conceptual graphs
  - Frames

- Logic representations have restricitions and can be hard to work with
  - Many AI researchers searched for better representations

Production Rules

- Rule set of <condition,action> pairs
  - "if condition then action"
- Match-resolve-act cycle
  - **Match**: Agent checks if each rule's condition holds
  - **Resolve**:
    - Multiple production rules may fire at once (**conflict set**)
    - Agent must choose rule from set (**conflict resolution**)
  - **Act**: If so, rule "fires" and the action is carried out
- Working memory:
  - rule can write knowledge to working memory
  - knowledge may match and fire other rules

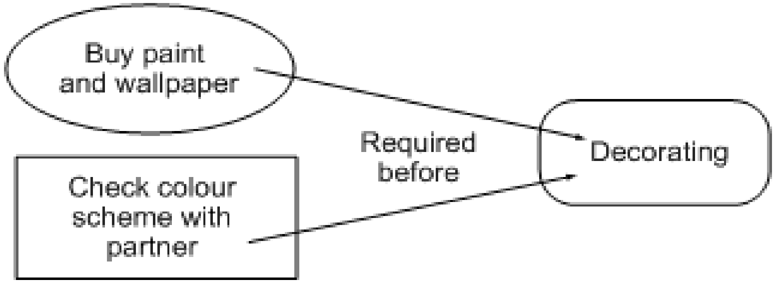
Production Rules Example

- IF (at bus stop AND bus arrives) THEN action(get on the bus)
- IF (on bus AND not paid AND have oyster card) THEN action(pay with oyster) AND add(paid)
- IF (on bus AND paid AND empty seat) THEN sit down

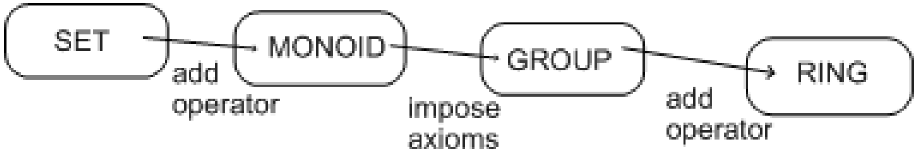
- conditions and actions must be clearly defined
  - can easily be expressed in first order logic!

Graphical Representation

- Humans draw diagrams all the time, e.g.
  - Causal relationships

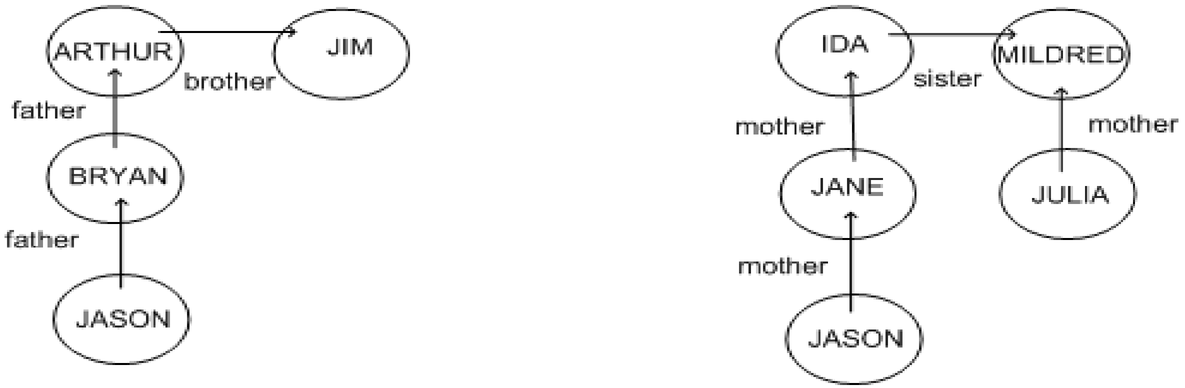


  - And relationships between ideas



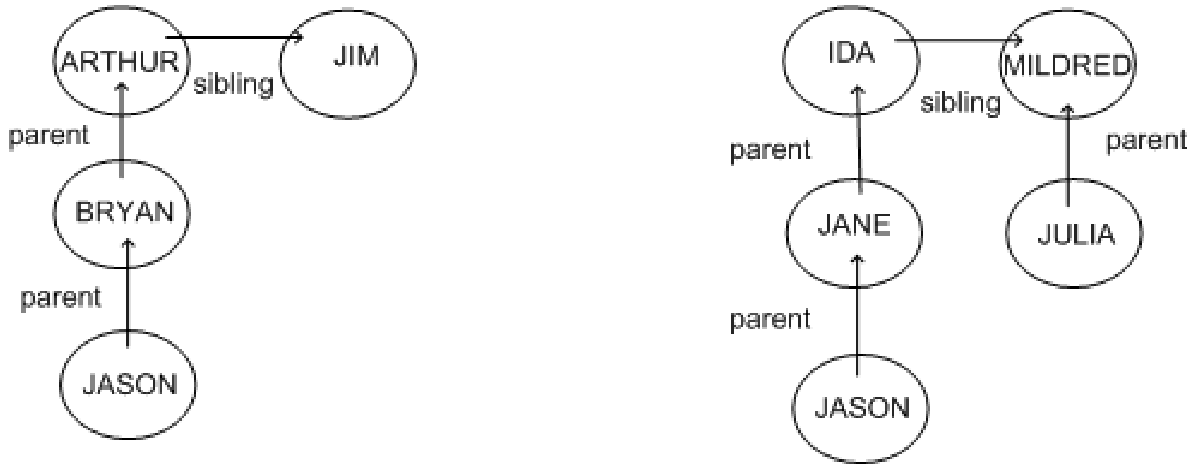
Graphical Representation

- Graphs easy to store in a computer
- To be of any use must impose a formalism



  - Jason is 15, Bryan is 40, Arthur is 70, Jim is 74
  - How old is Julia?

Semantic Networks



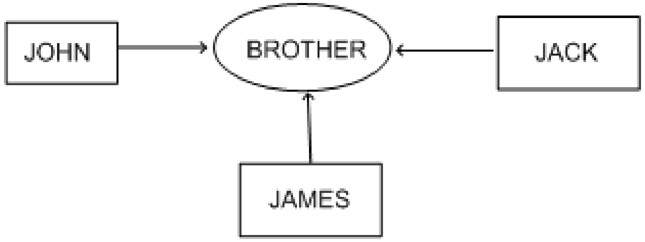
- Because the syntax is the same
  - We can guess that Julia’s age is similar to Bryan’s
- Formalism imposes restricted syntax

Semantic Networks

- Graphical representation (a graph)
  - Links indicate subset, member, relation, ...
- Equivalent to logical statements (usually FOL)
  - Easier to understand than FOL?
  - Specialised SN reasoning algorithms can be faster
- Example: natural language understanding
  - Sentences with same meaning have same graphs
  - e.g. Conceptual Dependency Theory (Schank)

Conceptual Graphs

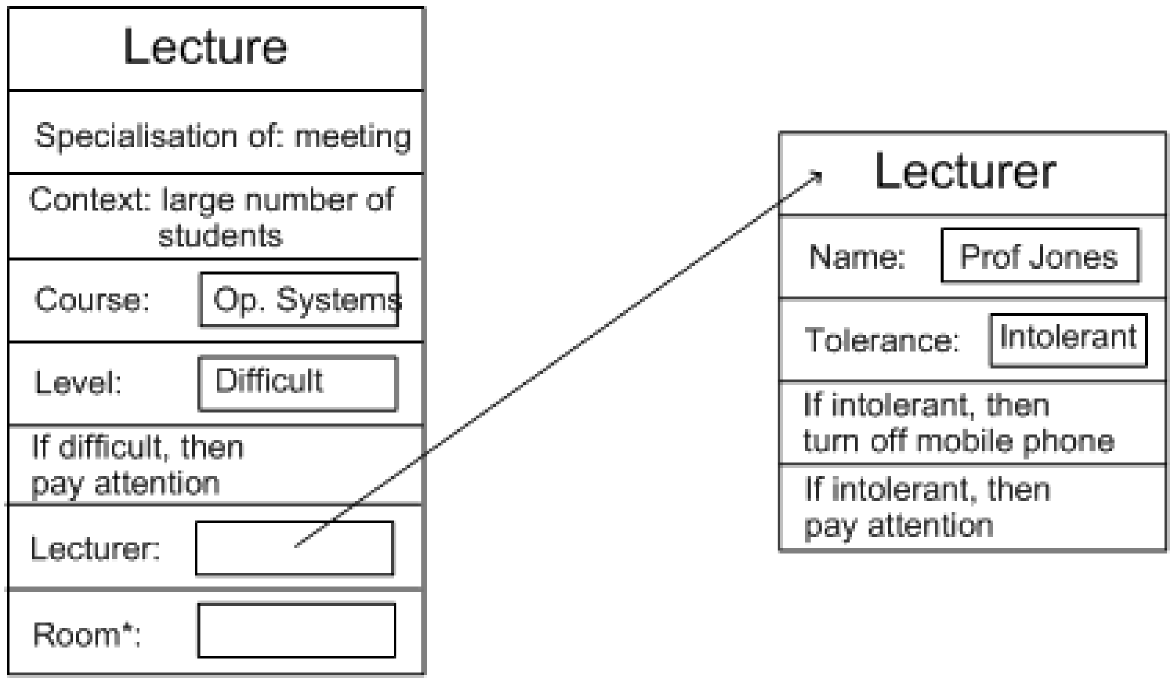
- Semantic network where each graph represents a single proposition
- Concept nodes can be
  - Concrete (visualisable) such as restaurant, my dog Spot
  - Abstract (not easily visualisable) such as anger
- Edges do not have labels
  - Instead, conceptual relation nodes
  - Easy to represent relations between multiple objects



Frame Representations

- Semantic networks where nodes have structure
  - Frame with a number of slots (age, height, ...)
  - Each slot stores specific item of information
- When agent faces a new situation
  - Slots can be filled in (value may be another frame)
  - Filling in may trigger actions
  - May trigger retrieval of other frames
- Inheritance of properties between frames
  - Very similar to objects in OOP

Example: Frame Representation



Flexibility in Frames

- Slots in a frame can contain
  - Information for choosing a frame in a situation
  - Relationships between this and other frames
  - Procedures to carry out after various slots filled
  - Default information to use where input is missing
  - Blank slots: left blank unless required for a task
  - Other frames, which gives a hierarchy

- Can also be expressed in first order logic

Representation & Logic

- AI wanted “non-logical representations”
  - Production rules
  - Semantic networks
    - Conceptual graphs, frames
- But all can be expressed in first order logic!
- Best of both worlds
  - Logical reading ensures representation well-defined
  - Representations specialised for applications
  - Can make reasoning easier, more intuitive