Knowledge Engineering

Semester 2, 2004-05

Michael Rovatsos
mrovatso@inf.ed.ac.uk

School of Informatics

The University of Edinburgh

Lecture 4 – Version Space Example/Logic Recap
21th January 2005

Today

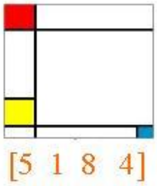
- Example: Version Space Learning
- Logic (Recap)

Updating the Version Space

- Final issue: how to update the version space?
- Assume S_i and G_i members of S-/G-sets.
  Each example can be a false positive (FP)/false negative (FN) for each of them:
  1. FP for S_i ➡ S_i too general ➡ throw S_i out (no consistent specialisations of S_i exist by definition)
  2. FN for S_i ➡ S_i too specific ➡ replace it by all its immediate generalisations
  3. FP for G_i ➡ G_i too general ➡ replace it by all its immediate specilisations
  4. FN for G_i ➡ G_i too specific ➡ throw G_i out (no consistent generalisations of G_i exist by definition)

Example: The "Mondrians" World

Idea: describe paintings by number/type of lines, number of rectangles and number oc colours ➡ which ones are "Mondrians"?



Number	Lines	Line types	Rectangles	Colours	Mondrian?
1	5	1	8	4	Yes
2	4	2	8	5	No
3	5	2	7	4	Yes
4	6	1	10	4	No
5	5	1	10	5	No

Logic & Logic-Based Reasoning

- Very brief overview of central concepts (more thorough treatment: FAI notes, AIMA ch. 7-9)
- Purpose of logic (in AI): describe knowledge in a formal way such that a computer can conduct **inference** on it
- Example:

  Socrates is human
  Every human is mortal
  ―――――――――――
  Socrates is mortal

- Inference procedure: use set of logical sentences (knowledge base) to derive new facts
- **Syntax** defines the set of admissible logical formulae; **semantics** defines the meaning of a formula

Logic & Logical Reasoning

- (At least) two uses of inference procedures:
  - Proving a sentence (**valid**/**satisfiable**/**unsatisfiable**)
  - Deducing new knowledge
- Desirable properties:
  - **Soundness** – the inference mechanism should only derive correct logical sentences
  - **Completeness** – the inference mechanism should be able to prove any correct sentence
  - **Tractability** – it should have reasonable time and space complexity
- Usually talk only about definite knowledge (though will look at methods for describing uncertain knowledge later on)

Propositional Logic

A very simple logic ...

- Atomic propositions: A, B, C are assigned truth values from $\{True, False\}$
- Logical connectives $\neg$, $\vee$, *wedge*, etc. are used to connect these to clauses
- Semantics given by truth tables

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \Rightarrow B$	$A \Leftrightarrow B$
F	F	T	F	F	T	T
F	T	T	F	T	T	F
T	F	F	F	T	F	F
T	T	F	T	T	T	T

- Easiest way to establish truth value of a sentence: truth table construction

Propositional Logic

- Alternatively use inference rules:

$$\frac{\alpha \Rightarrow \beta, \quad \alpha}{\beta}, \quad \frac{\alpha \wedge \beta}{\alpha}, \quad \frac{\alpha \vee \beta, \quad \neg\beta \vee \gamma}{\alpha \vee \gamma}, \quad \text{etc.}$$

- **Resolution**: If $l_i = \neg m_j$

$$\frac{l_1 \vee \ldots \vee l_k, \quad m_1 \vee \ldots \vee m_n}{l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee l_k \vee m_1 \vee \ldots \vee m_{j-1} \vee m_{j+1} \vee \ldots \vee m_n}$$

- *Any complete search algorithm using just the resolution rule can derive any conclusion entailed by a knowledge base in PL*

First-Order Logic

A richer logic that allows for describing properties of and relationships between objects

- Syntax:
  - Predicate symbols $P(x, y)$, $Q(f(x))$, etc.
  - Variables x, y, constants *John*, *Mary*, functions $f(x)$ (terms consist of functions an variables)
  - Usual connectives $\vee$, $\wedge$ etc.
  - Quantifiers: $\forall$, $\exists$
- Example:

$$\forall x \exists y.(x = ArmOf(John) \wedge Girl(y) \Rightarrow Longer(x, ArmOf(y)))$$

- Quantifiers allow variables to range over infinite domains
  BUT NO QUANTIFICATION OVER PREDICATES!

First-Order Logic

- A **substitution** replaces variables by terms
  $\vartheta = \{x/John, y/Mary\}$
- **Unification**: process of making two formulae identical by determining and applying suitable substitutions
- Example: $P(x, f(x))$ and $P(y, f(y))$ can be unified by applying $\{x/y\}$ to both (while $P(John)$ and $P(Mary)$ cannot be unified)
- **Interpretation**: A function that assigns a truth value to each ground (variable-free) proposition
- Inference: essentially uses similar rules to propositional logic combined with unification

First-Order Logic

Example: if $\text{UNIFY}(l_i, \neg m_j) = \vartheta$, then we can derive

$$\frac{l_1 \vee \ldots \vee l_k, \quad m_1 \vee \ldots \vee m_n}{\text{SUBST}(\vartheta, l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee l_k \vee m_1 \vee \ldots \vee m_{j-1} \vee m_{j+1} \vee \ldots \vee m_n)}$$

by applying the generalised first-order resolution rule

- First-order resolution: **refutation-complete**, i.e. if a set of sentences is unsatisfiable, resolution will always derive a contradiction
- *Cannot be used to generate all logical consequences, but can be used to establish that a given sentence is entailed by a set of sentences*