Artificial Intelligence

Topic 15

Practical Aspects of Logic in AI

Reading: Russell and Norvig, Chapter 10

The problem with logic...
The problem with using logic in AI is:

- logic’s useful to reason about arbitrary scenarios
- to reason about arbitrary scenarios you need to be able to express arbitrary scenarios
- logics that can express arbitrary scenarios have very difficult (impossible) decision problems
- to reason with a logic you need to solve its decision problem...

The more useful a logic becomes, the harder to use it becomes!

Modern research in logic in AI is a constant trade-off between expressivity and complexity.

Outline

- The problem with logic...
- Temporal and Epistemic Logic
- Ontologies
- Description Logics
- The Semantic Web
- Very Large Knowledge Bases

Temporal and Epistemic Logic

One successful approach to applying logic in AI has been to tailor the logic to a fixed domain.

Two common domains are Time (temporal logic) and Knowledge (epistemic logic).

Temporal and epistemic logic extend propositional logic (so they throw away all the unwanted expressivity and complexity of first order logic).

Temporal logic adds operators for “at the next moment of time”, and “at some future moment of time”.

For each agent \( a \), epistemic logic adds an operator for “agent \( a \) knows”
**Temporal Logic**

Sentences of temporal logic are built from atomic propositions ($P$), logical connectives ($\land$ and $\neg$), and temporal operators next ($X$) and future ($F$).

"It is always the case that if $P$ is true, then $Q$ will be true at the next state" is expressed by:

$$\neg F \neg (P \rightarrow XQ)$$

Models of temporal logic can be as simple as a sequence of sets of propositions, $S_0, S_1, S_2, \ldots$ where $S_i$ is the set of propositions true at time $i$.

Temporal logics have been very useful for reasoning about the properties of programmes, and are used in the automatic verification of software and hardware.

**Epistemic Logic**

Sentences of epistemic logic are built from atomic propositions ($P$), logical connectives ($\land$ and $\neg$), and for each agent $a$, an operator $a$-knows ($K_a$).

"Alice does not know that Bob knows it’s her birthday" is expressed by:

$$\neg K_{Alice}K_{Bob}AlicesBirthday$$

Models for epistemic logic consist of possible worlds, where each agent considers a set of possible worlds. Each possible world could be an accurate description of the true world, so an agent knows something is true if it is true in every world that the agent considers possible.

Epistemic logic is often used for reasoning about knowledge bases, AI and multi-agent systems, rather than reasoning in knowledge bases, AI and multi-agent systems.

**Ontologies**

An alternative to making logics very domain specific is to make them very flexible, but with a restricted syntax.

*Ontology* means the study of existence, the things that are. In the field of philosophy this is a branch or metaphysics, but in information sciences the study of ontologies has come to mean the way we can describe concepts.

Representing abstract concepts is referred to as ontological engineering or knowledge engineering.

**Examples Concepts**

Concepts are descriptions, and objects may satisfying concepts. For example "Basketball" is a concept, corresponding to large, orange, rubber balls. An object (my son's ball) may satisfy the concept of "Basketball".

Ontological Engineering involves describing the relationships between concepts: Every basketball is a ball; every basketball bounces; some things that bounce are not basketballs.

There are even meta-concepts to define properties of concepts. For example *individuation* is the property that objects satisfying a concept are individuals (like basketballs) rather than conglomerates (like butter).
Description Logics

First order logic can be used very effectively for describing ontologies, but the complexity is often prohibitive.

Description Logic is a fragment of first-order logic specifically for describing ontologies.

Applications of DL

Description logic is a fragment of first-order logic that can represent simple properties of complex and general ontologies. It should be clear that it is a very natural representation for Knowledge bases, and conducive to reasoning techniques such as resolution and backwards chaining.

Two specific examples we will look at are:

1. The semantic web: The semantic web is an ambitious ongoing project which is attempting to associate meaning to concepts with webpages.
2. Medical ontologies: To help standardize medicine and pharmacology, several databases have been established to standardize all medical terminology.

T-Boxes and A-Boxes

Description logic consists of T-Boxes for describing relationships between concepts (or terms) and A-Boxes for describing whether an object belongs to a concept.

T-Boxes are built from unary and binary predicates:

\[
\text{Woman} \equiv \text{Person} \sqcap \text{Female}, \quad \text{Mother} \sqsubseteq \exists \text{hasChild} \cdot \text{Person}
\]

A-Boxes assign concepts to objects:

\[
\text{Mother(Diane)}, \quad \text{Person( Isaac)}
\]

The logical operators used are \(\land, \sqcup, \lnot\), but the quantifiers, \(\forall\) and \(\exists\) are very restricted in their use.

The Semantic Web

The semantic web is an extension of the current html standard for web-pages. It uses XML to incorporate meta-data into web-pages and allows for web-search and information synthesis based on more than just word frequencies.

The main components are:

1. XML: the base layer as a standard for passing information
2. RDF: a simple language for expressing data models
3. OWL: an ontology language for describing relationships between classes, based on description logic.
4. SPARQL: a protocol and query language.

It is hope that semantic web resources will enable intelligent agents to synthesise information and execute plans in the web domain. For example, imagine being able to deploy an agent to book you the cheapest tickets to London for two weeks during July or August.
Medical Ontologies

SNoMed-CT is the Systemized Nomenclature of Medicine-Clinical Terms. It is a systematic definition of over 1 million medical terms and concepts, and is organized for automated searching and deduction.

Concepts in the database are either primitive (e.g. virus) or defined using Description Logic (e.g. juvenile diabetes is the intersection of the disease diabetes and state of being a child). In addition to storing medical conditions, it also contains information on drugs, biology, topography, occupations etc to allow for comprehensive knowledge base.

SNoMed-CT requires that concept definitions form a Directed Acyclic Graph.

SnoMed Example

Eg. SNOMED-CT enables the following set of associations for tuberculosis pneumonia:

- A kind of lung infection
- A kind of pneumonia
- Caused by mycobacterium tuberculosis
- Site of infection: the lung
- A morphology inflammation

The End

The End.

Good luck in the exam....