Knowledge representation
Semantic networks and frames

1. Introduction: What is knowledge?

The science that studies various issues about knowledge is called **epistemology**. The name is derived from the Greek *episteme* ("knowledge") and *logos" ("reason"), and accordingly the field is sometimes referred to as the **theory of knowledge**.

The main problems considered in epistemology are:

- What is knowledge?
- Is genuine knowledge attainable at all?
- What are the limits of knowledge?
- From what faculties of the mind does knowledge originate?
- Which method should be used to obtain valid knowledge?
- How do you justify a priori statements?
- Where is the boundary between the subjective and objective factors?
- What is the nature of truth?

Artificial intelligence has also contributed to the theory of knowledge. Here is the view on knowledge adopted in Artificial intelligence, as described in **Principia Cybernetica Web**:

> "Knowledge consists of models that attempt to represent the environment in such a way as to maximally simplify problem-solving. It is assumed that no model can ever hope to capture all relevant information, and even if such a complete model would exist, it would be too complicated to use in any practical way. Therefore we must accept the parallel existence of different models, even though they may seem contradictory. The model which is to be chosen depends on the problems that are to be solved. The basic criterion is that the model should produce correct (or approximate) predictions (which may be tested) or problem-solutions, and be as simple as possible."

Thus, what we refer to as knowledge in AI is our model of the world.

**Basic assumption in Artificial Intelligence:**

**Intelligent behavior can be achieved through the manipulation of symbol structures** (representing bits of knowledge). This is based on the physical symbol system hypothesis, proposed by Newell and Simon in "Computer Science as Empirical Inquiry: Symbols and Search," Communications of the ACM, March 1976, pp 113-126.]

What is a physical symbol system?

> "A physical symbol system consists of a set of entities, called symbols, which are physical patterns that can occur as components of another type of entity called an expression (or symbol structure). Thus, a symbol structure is composed of a number of instances (or tokens) of symbols related in some physical way (such as one token being next to another). At any instant of time the system will contain a collection of"
these symbol structures. Besides these structures, the system also contains a collection of processes that operate on expressions to produce other expressions: processes of creation, modification, reproduction and destruction.

The Physical Symbol System Hypothesis:

A physical symbol system has the necessary and sufficient means for general intelligent action. By necessary we mean that any system that exhibits intelligence will prove upon analysis to be a physical symbol system. By sufficient we mean that any physical symbol system of sufficient size can be organized further to exhibit general intelligence. By general intelligent action we wish to indicate the same scope of intelligence as we see in human action.

Main questions:

How to represent knowledge as symbol structures

How to use that knowledge to solve problems.

Features:

• A knowledge representation (KR) is most fundamentally a substitute for the thing itself, used to enable an entity to determine consequences by thinking rather than acting, i.e., by reasoning about the world rather than taking action in it.

• It is a set of ontological commitments, i.e., an answer to the question: In what terms should we think about the world?

Knowledge representation is a multidisciplinary subject that applies theories and techniques from three other fields:

1. **Logic** provides the formal structure and rules of inference.

2. **Ontology** defines the kinds of things that exist in the application domain.

3. **Computation** supports the applications that distinguish knowledge representation from pure philosophy.

Without logic, a knowledge representation is vague, with no criteria for determining whether statements are redundant or contradictory.

Without ontology, the terms and symbols are ill-defined, confused, and confusing.

Without computable models, the logic and ontology cannot be implemented in computer programs.

Knowledge representation is the application of logic and ontology to the task of constructing computable models for some domain.
knowledge representation languages:

high level representation formalisms, that can in principle be implemented using a whole range of programming languages.

crucial thing about knowledge representation languages - should support *inference*.

inference - any way to get new expressions from old.

We can't represent explicitly everything that the system might ever need to know - some things should be left implicit, to be deduced by the system as and when needed in problem solving.

Features of a good knowledge representation language:

- **Representational adequacy**: It should allow you to express the knowledge you wish to represent in the language. For example, suppose you want to represent the fact that "Richard knows how old he is". This turns out to be difficult to express in some languages.

- **Inferential adequacy**: It should allow new knowledge to be inferred from a basic set of facts.

- **Inferential efficiency**: Inferences should be made efficiently.

- **Clear syntax and semantics**: It should be clear, and have a well defined *syntax* and *semantics*. We want to know what the allowable expressions are in the language, and what they mean. Otherwise we won't be sure if our inferences are correct, or what the results mean. For example, if we have a fact gray(elephant) we want to know whether it means all elephants are gray, some particular one is gray, or what.

- **Naturalness**: The language should be reasonably natural and easy to use.

**Reminder: A KR is not a data structure**

Part of what makes a language representational is that it carries meaning i.e., there is a correspondence between its constructs and things in the external world.

A semantic net, for example, is a representation, while a graph is a data structure. They are different kinds of entities, even though one is invariably used to implement the other, precisely because the net has (should have) a semantics. That semantics will be manifested in part because it constrains the network topology: a network purporting to describe family memberships as we know them cannot have a cycle in its parent links, while graphs (i.e., data structures) are of course under no such constraint and may have arbitrary cycles.

While every representation must be *implemented* in the machine by some data structure, the *representational property is in the correspondence to something in the world* and in the constraint that correspondence imposes.
2. Approaches to Knowledge Representation in AI

2.1. Declarative vs procedural knowledge:

Declarative: represent what you know through static collection of facts, along with inference rules

Procedural: represent knowledge as a collection of procedures that tell us how to use it

Declarative representations to be discussed:

- First-order logic
- Semantic nets
- Frames
- Production systems

Frames and production systems may incorporate some procedural knowledge

What is most important in each of these representations is their representational power, their semantics and their inference process, rather than the specific syntax.

2.2. First Order Predicate Logic

has a well defined syntax and semantics,

is concerned with truth preserving inference.

Problems: time, beliefs and uncertainty are difficult to represent.

2.3. Semantic nets and frames:

Represent factual knowledge about classes of objects and their properties

Not formal systems.

Basic inference mechanism: inheritance of properties

Problems: quantifiers, representing disjunction and negation.

2.4. Production rules

If -then relation:

\textit{If high\_temperature then prescribe aspirin}

Production systems:

a set of if-then rules, - typically state that if certain conditions hold, then some action should be taken.

a working memory - represents the facts (as semantic nets or frames) that are currently believed to hold.
3. Semantic Nets

Originally developed in the early 1960s by M. Ross Quillian to represent the meaning of English words, as a psychological model of human memory.

Semantic nets are used in AI to describe relationships between concepts, objects, or events.

A semantic net is represented as a graph, where the nodes in the graph represent concepts, and the arcs represent binary relationships between concepts.

- Composed of Nodes and Arcs (Links)
- Nodes represent objects, attributes, and values
- Links represent relationships between nodes
- Semantic labels attached to links must have semantics attached to be a semantic network otherwise just a graph

This network represents the fact that mammals and reptiles are animals, that mammals have heads, an elephant is a mammal, and Clyde is a particular elephant.
Types of relations: depending on the application. (e.g. has_parts, likes, etc)
Important relations: subclass / member

Is a / is instance of

Inheritance of information: more specific (sub)classes inherit (get) properties from more general (super)classes through is_a / is_instance_of links
Example: Inferring facts not explicitly represented: Clyde has a head.

representational and inferential adequacy -
problems with representing quantifiers, (such as "every dog in town has bitten the constable")
may have 2 arguments only
cannot represent disjunction and negation

inferential efficiency: considered to be good.

Used for: ontologies, relational networks
Example:
http://www.troubleshooters.com/tpromag/199907/_model.htm
Using Semantic Nets to Model Troubleshooting's Knowledge

Frames

Frames are packets of information about types of entities and their instances: all the information relevant to a particular concept is stored in a single complex entity, called a frame.
Frames support inheritance.

Frames and semantic nets: Frames can be viewed as a structural representation of semantic nets.
Examples:

Mammal
  subclass: Animal
  warm_blooded: yes

Elephant
  subclass: Mammal
  * color: grey
  * size: large

Clyde
  instance: Elephant
  colour: pink
  owner: Fred

Nellie:
  instance: Elephant
  size: small

Components of a frame entity:

Name - correspond to a node in a semantic net
attributes or slots filled with particular values
  E.G. in the frame for Clyde, instance is the name of a slot, and elephant is the value of
  the slot.

Names of slots correspond to the links in semantic nets
Values of slots correspond to nodes.
Hence each slot can be another frame.

Example:

Size:
  instance: Slot
  single_valued: yes
  range: Size-set

Owner:
  instance: Slot
  single_valued: no
  range: Person

The attribute value Fred (and even large and grey etc) could be represented as a frame, e.g.,:

Fred:
  instance: Person
  occupation: Elephant-breeder
Frames have greater representational power than semantic nets

- Necessary attributes
- Typical attributes
  ```
  "*" used to indicate attributes that are only true of a typical member of the class, and not necessarily every member.
  ```
- Type constraints and default values of slots, overriding values.
- Slots and Procedures: a slot may have a procedure to compute its value of the slot, if needed e.g. object area, given the size

Inheritance

If a slot is not defined for a given frame, we look at the parent-class slot with the same name

Simple if: single parent-class, problems with several parent classes (e.g., Clyde is both an elephant and a circus-animal)

Which parent to inherit from first?

Various mechanisms for making this choice - choosing the most specific parent class to inherit from.

Problems with frames: same as with semantic nets

Useful for:

- classifying new instances of familiar entities (objects/events/places/tasks)
- anticipating the attributes of such instances
- inferring the presence and properties of their parts or participants