Intelligent Systems: Reasoning and Recognition

James L. Crowley

ENSIMAG 2 / MoSIG M1 Second Semester 2014/2015

Lesson 9 11 March 2015

Structured Knowledge Representation

Structured Knowledge Representation

Kinds of Knowledge (reminder)

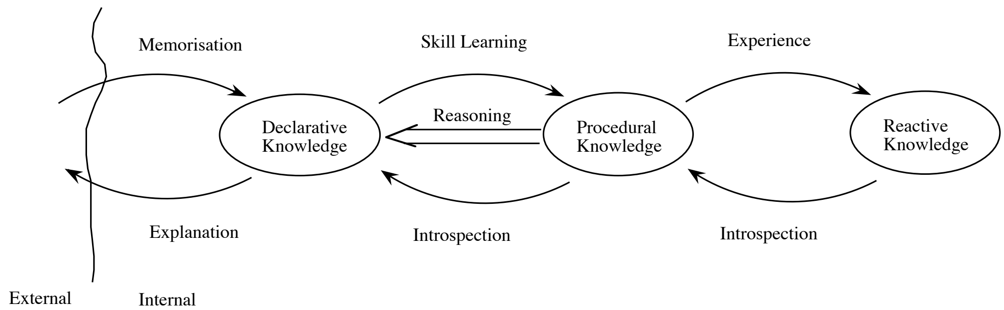
Cognitive Psychologists identify different categories of knowledge:

Reactive: stimulus - response.

Procedural knowledge: A series actions that lead to a goal. A compiled expression of knowledge, where each action triggers the next action.

Declarative knowledge: A symbolic expression of competence.

Declarative knowledge is used for communication and for abstract reasoning.



Structure Knowledge Representations are a form of Declarative representation.

Declarative representations are useful for communication and for reasoning about knowledge to generate new knowledge.

Structure Knowledge Representations have been explored for many years as a general symbolic representation for declarative knowledge.

As we have seen, a fundamental concept for organizing such structures is the concept of "relation".

Relations as N-Ary Predicates

Relations are a key concept in structured knowledge representations.

Examples include temporal relations, spatial relations, family relations, social relations, administrative organizations, military hierarchies, etc. This is not a closed list.

Relations are formalized as N-Ary Predicates.
The valence or Arity of a relation is the number of entities that it associates.

Unary or Monadic: Man(Bob)
Binary or dyadique: Brother(Charlie, Bob)
Ternary or triadique: Action(Jim, Talks-To, Bob)

Unary relations represent properties for entities
Binary relations associate entities.

Examples From Blocks world
Arity 0 : (HandEmpty)
Unary : (OnTable A)
Binary: (On A B)
Ternary: (Over A B C) ;; Block A is a bridge over B et C.

A symbol can be defined as a form of triadic relation between 3 entities: a sign, a thing and an agent. The agent interprets the sign to represent the thing.

```
(defclass symbol (is-a USER)
    (slot sign)
    (slot thing)
    (slot agent)
)
```

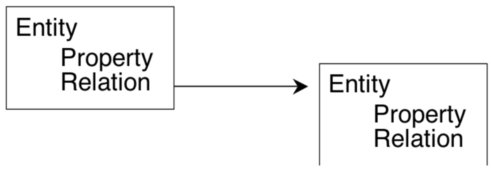
Relations can be represented "implicitly" or "explicitly".
In an implicit representation, the thing is represented as a value of a slot. We saw this with the family relations:

(defclass PERSON (slot NAME) (slot FATHER) (slot MOTHER)).

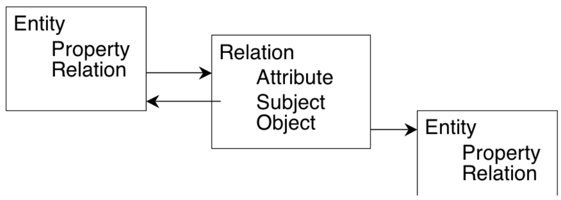
The slots FATHER and MOTHER contained pointers to an object of the class PERSON.

In an explicit representation, the ternary structure is expressed as a schema. The relation is represented as an object with pointers to the component entities.

Implicit representation:



Explicit:



Implicit representations are simpler and slightly more efficient in computing and memory.

Explicit representations allows a system to reason about relations.

This is yet another example of the power of representing program as data.

Situation Models

P. Johnson-Laird 1983 - Mental Models.

Situations models are used in cognitive psychology to express the mental models that people use to understand.

Properties: Sensed values
Entities: Anything that can be named or designated; People, things, etc.
(entites can be defined as a correlation of properties)
Relations: An N-ary predicate (N=1,2,3 …) over entities.
(relations are defined as tests on the properties of entities).
Situation: A set of relations between entities

Example: John is facing Mary. John is talking to Mary. Mary is not listening.

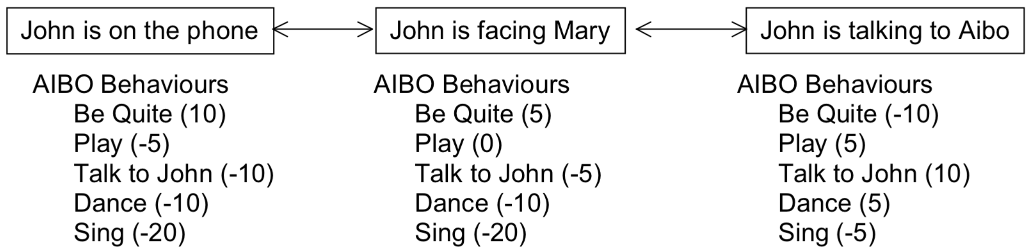
facingP(John, Mary) ^ talkingP(John,Mary)^ not listeningP(Mar, John))

Situations can be organized into a state space referred to as a situation network. Each situation (or state) corresponds to a specific configuration of relations between entities. A change in relation results in a change in situation (or state).

The situation graph, along with the set of entities and relations is called a Context.

Each situation can prescribe and proscribe behaviours.
Behaviours: List of actions and reactions that are allowed or forbidden
Attention: entities and relations for the system to observe, with methods to observe the entities
Default values: Expectations for entities, relations, and properties
Possible next situations: Adjacent neighbors in the situation graph.

Behaviours include methods for perception, and methods for interaction with the external world



Each situation indicates:

Transition probabilities for next situations

The appropriateness or inappropriateness of behaviours

Situation models are used to construct context aware systems.

A "Context" (the information that must be supplied by the system) is defined as

1) A set of entities, with their properties.

2) A set of relations between entities

3) A network of situations, such that each situation specifies

- A list of adjacent situations, possibly with transition probabilities.

-A list of system behaviours that are allowed or fobidden, possibly with preferences or probabilities for execution.

Frames

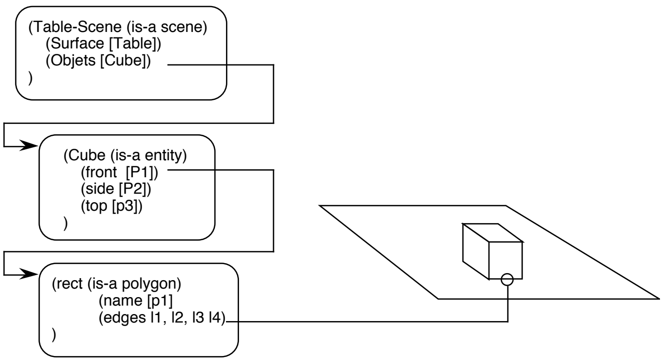
Marvin Minsky, A Framework for Representing Knowledge, in: Patrick Henry Winston (ed.), The Psychology of Computer Vision. McGraw-Hill, New York (U.S.A.), 1975.

Frame: A Structured Representation to provide context for focusing visual interpretation of scenes.

In computer vision, the problem can be stated as transforming a table of numbers into a symbolic description of the scene. This problem can be made easy when you know exactly what to look for.

M. Minsky proposed Frames as a structure to guide scene interpretation in computer vision. A frame guided interpretation in a top down manner, telling the system where to look and what to look for. Minsky's insight was that it is much easier to see if you know what to look for.

Frames provide visual context to guide scene interpretation.
A Frame tells the program what to look for and where to look for it.



A Frame is composed of a set of "slots" and "methods".

A slot is a named place-holder for a pointer. The slots point to other frames that represent entities that are described (or interpreted) by the frame. Ultimately, some slots point to raw perceptions. When a slot points to an entity it is said to play a "role" in the frame. Frames typically come with methods (procedures) for searching for the entities that can plays roles in the frame.

Frames can be formalized as a set of relations between entities having certain properties.

Scripts

Schank and Abelson Scripts, Plans, Goals and Understanding, Erlbaum, 1977.

A script is a data structure used to represent a sequence of events. Scripts are used for interpreting stories. Popular examples have been script driven systems that can interpret and extract facts from Newspaper Stories.

Scripts have been used to
1) Interpret, understand and reason about stories,
2) Understand and reason about observed events
3) Reason about observed actions
4) Plan actions to accomplish tasks.

A script is composed of
1) A scene
2) Props (objects manipulated in the script)
3) The actors (agents that can change the state of the world).
4) Events
5) Acts: A set of actions by the actors.

In each scene, one or more actors perform actions. The actors act with the props. The script can be represented as a tree or network of states, driven by events.

As with Frames, scripts drive interpretation by telling the system what to look for and where to look next. The script can predict events.

Example of a script

The classic example is the restaurant script:

Scene: A restaurant with an entrance and tables.
Actors: The diners, servers, chef and Maitre d'Hotel.
Props: The table setting, menu, table, chair.
Acts: Entry, Seating, Ordering a meal, Serving a meal, Eating the meal, requesting the check, paying, leaving.

Schema systems capture Frames, Scripts, and semantic network as networks of schema. Typically, schema represent relations between entities playing roles:

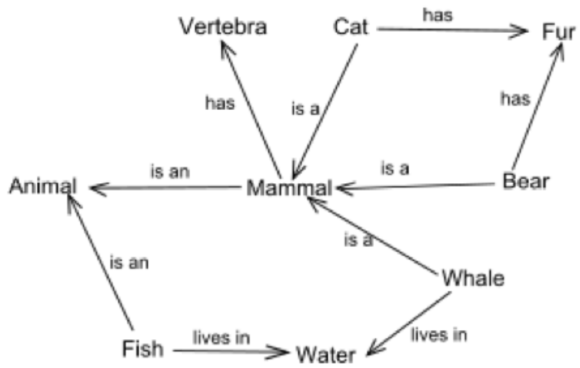
Semantic Nets

M.Quillian, (1968). Semantic Memory, in M. Minsky (ed.), Semantic Information Processing, pp 227-270, MIT Press

J. F. Sowa (1987). "Semantic Networks". in Stuart C Shapiro. Encyclopedia of Artificial Intelligence. Retrieved 2008-04-29.

Network of semantic relations between concepts.

Used as a form of knowledge representation for language understanding and translation



Semantic nets are networks of words with rich sets of relations. Because words are related in a network and not a strict hierarch, a semantic net leads to circular definitions if used to "define" words.

Ultimately definitions are grounded on perception (as observed by Kant). This is known as the Symbol Grounding problem.

Fundamental problems with Schema Systems.

Some Fundamental Problem with all schema systems.

1) Knowledge Acquisition: Learning a schema system is long, tedious, and ad hoc process. There is a temptation to overload with useless information.

2) Context Recognition (The Frame problem): Many problems are easily solved when the context and situation are known. Recognizing the correct context can be very difficult.

3) Ontology Alignment: Two schema systems describing the same context, may not use the same symbols for entities and relations. Alignment is the problem of determining equivalences.

Solution to both of these can be obtained from probabilistic techniques for learning and recognition.