Cognitive Psychology - Memory Models, Knowledge Representation

We've been covering topics such as search, predicate calculus, heuristics, and other formal methods for performing intelligent tasks. Let's start from the other direction - the way that people perform intelligent tasks - and then see if we can construct models that emulate humans.

Semantic Knowledge is loosely defined as organized knowledge about the world. The famous psychologist Tulving contrasted semantic knowledge with episodic knowledge, which is temporal knowledge (e.g., "I wrote this document yesterday" is temporal, while the existence of this document and its contents is semantic).

Sentence Verification Experiment

The sentence verification experiment has led to several observations about semantic memory. The experiment itself is simple. People are presented with simple sentences, and they must consult their semantic memory to determine if the sentence is true or false.

Example sentence verification task: Answer true or false as quickly as you can:

1. A carrot is a vegetable.
2. A carrot is an organism.
3. A emu is a bird.
4. A robin is a bird.
5. A petunia is a tree.
6. A maple tree is a plant.
7. A horse is a vegetable.
8. A bird is a reptile.

What can we learn from such simple questions? The answers are not meaningful. But by timing how long it takes to answer, we can deduce some interesting observations about how memory works.

1. True-False effect. People answer true items faster than false items. In some studies, people answer the true questions about 0.17 seconds faster than the false questions. That may not seem significant, but the time difference is statistically significant across trials. In general, people handle true information better than false.
2. Category size effect. People usually reach decisions faster when an item is a member of a small category rather than a large category. For example, you can answer "A carrot is a vegetable" faster than "A carrot is a organism".
3. Typicality effect. People usually reach decisions faster when an item is a typical member of a category rather than an unusual member. For example, #3 takes longer to answer than #4.
4. Context effect. People respond faster to an item if it was preceded by a similar item (also called priming). You probably verified #4 quickly since you already had "bird" on your mind. There is also an unconscious priming effect – subliminal messages.

Typicality has also been studied extensively, and is often referred to as **prototypes**. A prototype is the most representative member of a category.

Example: Visualize a bird. Visualize a dessert. Visualize clothing.
Did you think of a robin? Apple pie? Shirt or pants?

These are all category prototypes. There are a number of interesting observations regarding prototypes. Here are only a few:

1. Prototypes are examples of a category and can substitute for the category in a sentence.
2. Categories based on prototypes are learned quickly. (Dani tribe experiment identifying colors. Their native colors are only light and dark).
3. Prototypes serve as reference points. From 1000 and 1004 fill in the blanks: ___ is basically ____.

Feature Comparison

Based on these observations, Smith and Rips proposed the **feature comparison model** of semantic memory in 1974. In this model, they proposed that concepts are stored with a list of features. The concept of bird would have features like:

Has feathers * defining feature
flies
eats worms
Has wings * defining feature
Has two legs

When comparing two concepts the following algorithm is used:

Compare all features of subject with predicate to determine similarity
If high, match = true
If low, match = false
If medium, compare defining features
    If match, then true
    If mismatch, then false

This simple technique suggests that similar items can be matched quickly. Dissimilar items should also result in a quick false. Items that are partially similar take longer, because we have to compare defining features.

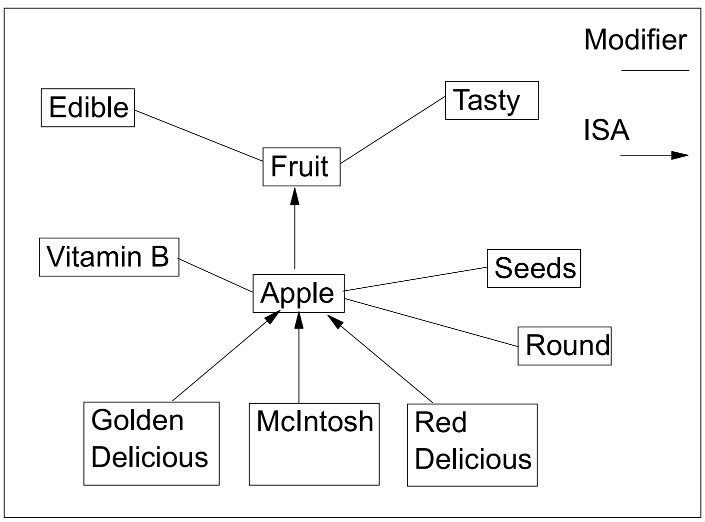
This model can explain some of the observations we made earlier. The typicality effect says that we can quickly identify a typical member of a category. Under this model, a typical item should have many features in common resulting in a quick answer. For example, "Is a robin a bird". It also helps explain false matches, like “Is a bat a bird?” But the question "Is a penguin a bird" will take longer because we have to compare defining features.

The model fails in other respects, however. For the category size effect, a small category should have many defining features (consider dog vs. animal). So we should see a longer delay for comparing an item in a small category, but the results are the opposite.

Semantic Networks

Collins and Loftus proposed the semantic network model for human memory in 1975 based on earlier models. The basic ideas from the model are still in use today. This model proposes a netlike organization of concepts in memory, with many interconnections. Each concept can be represented as a node. There are links or associations that connect a particular node with other concept nodes.

Example:



The arrowed links indicate ISA relationships. The ISA links form a hierarchy from the most general topics at the top, to the more specific at the bottom. The straight links indicate modifiers, properties, or related concepts, but not superordinate concepts.

Collins and Loftus popularized the notion of **spreading activation**. Depending on the input, various items or nodes in the network are activated. This starts a chain reaction where neighboring nodes to the activated nodes are activated with a tag indicating the source. This process repeats, until a **collision or intersection** occurs from two different sources at the same node. At this point, information is evaluated to validate any activated tags.

Example: “A Golden Delicious is a fruit.” The golden delicious node and the fruit node are activated, and a collision occurs at Apple. We then evaluate that a Golden Delicious is a fruit and produce a ‘YES’ answer. Notice that "Fruit" would have also propagated to "Edible" and "Tasty" but no collision occurs there.

Example: “A platypus is a fruit.” Activation spreads from both platypus and fruit, but there is no intersection except at a high level concept of “object” or “living entity”. Consequently a “no” answer is returned.

Note that it takes many more steps of spreading propagation in the platypus example than the apple example. This matches the "True/False" effect that we observed from the sentence verification experiment, where false questions took longer to answer.

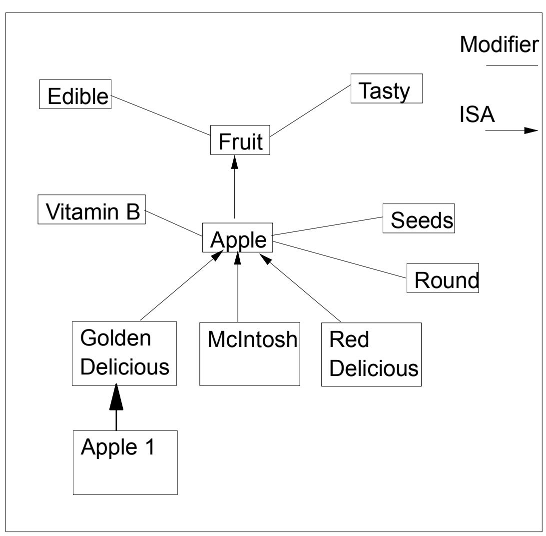
The model also might explain the category effect. To determine if a Golden Delicious is a fruit requires two links to be traversed, while to determine if it is an Apple requires only one link to be traversed. Consequently, it is faster to name items that are closer together in the network.

What about the priming/context effect? One possible modification to explain this effect is that nodes, once activated, stay activated for some period of time before decaying off. If the bird node was activated from a previous sentence, we might be able to answer another bird question more quickly without the spreading activation requirement.

Finally, the typicality/prototype effect might be answered by the strength of a link. If a link is frequently used, we might increase the strength of the link, resulting in faster travel times between nodes. Consequently we can quickly link "bird" with "robin" and we might even equate the two together.

The model doesn’t explain everything though, for example the Tip of the Tongue phenomenon.

Note also that instances, or actual instantiations of items, are typically indexed below the concept of that item:



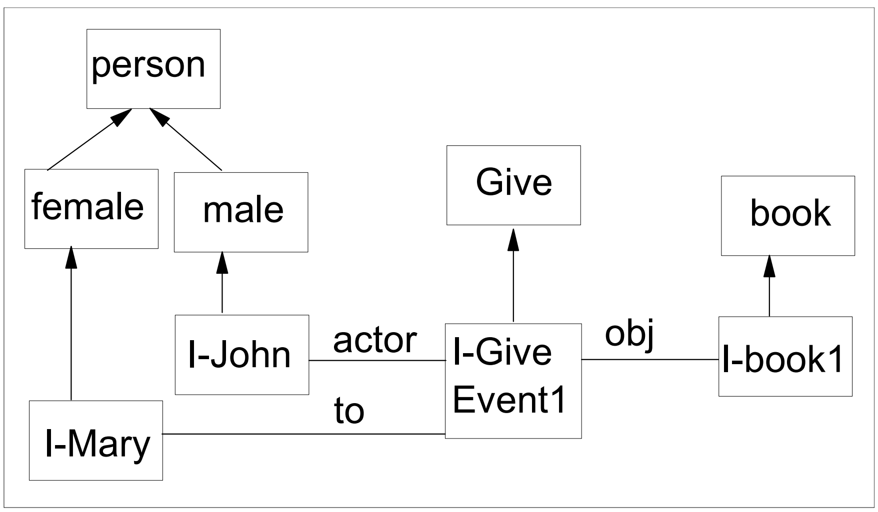
Here, actual apple "Apple 1" is an instance of a golden delicious apple.
An important consideration is that of inheritance; apples inherit all the modifiers of fruit, golden delicious all the attributes of apple, and so on, so a golden delicious is also known to be tasty and edible.

Note that if we were doing the equivalent in logic, we would be writing something like:

Golden-Delicious(X) $\rightarrow$ Apple(X).
Golden-Delicious(Apple 1).

These types of networks are widely accepted. However they seem to work best for representing a hierarchy of things. But we can even use the same technique for actions and declarative sentences. Nodes can be abstract objects, not just physical things.

Consider the sentence: "John gave Mary a book."



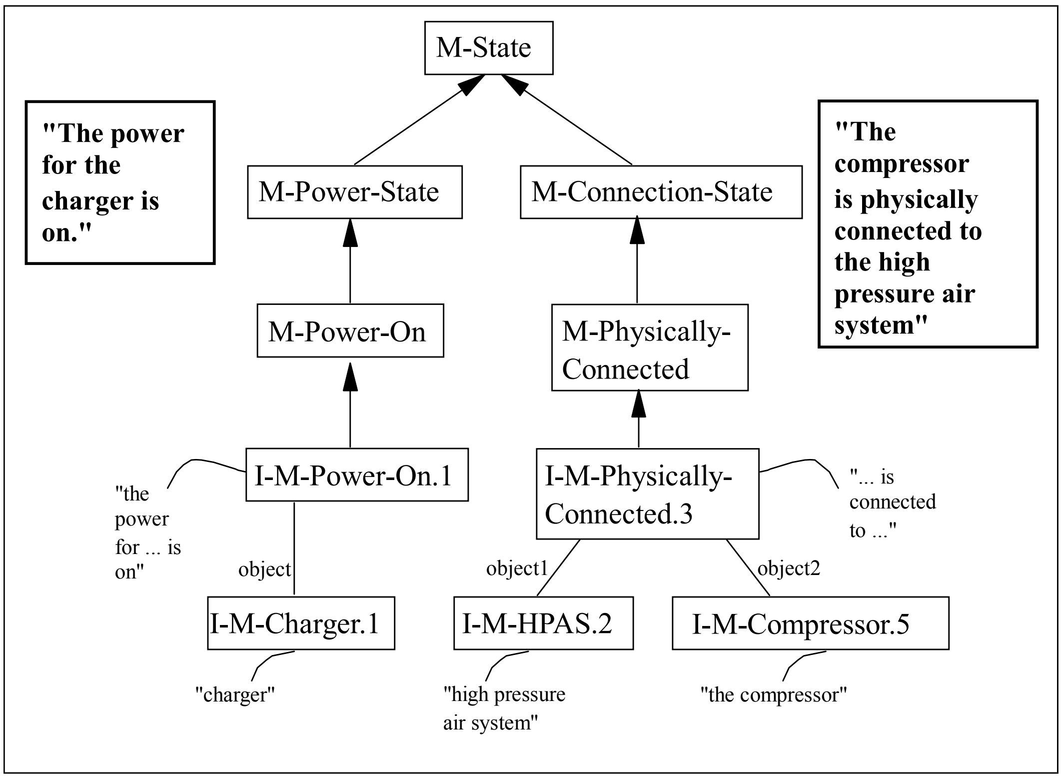
Here the give event is created with modifiers to indicate the actor, recipient, and object. Note that these are specific instances of the give event. We could represent arbitrary actions and relations to convey complex information within these networks, and is in many cases more intuitive than predicate calculus or other logic techniques. John, Mary, book1, and the Give-Event are all instances of the abstract concepts of each.

Frames

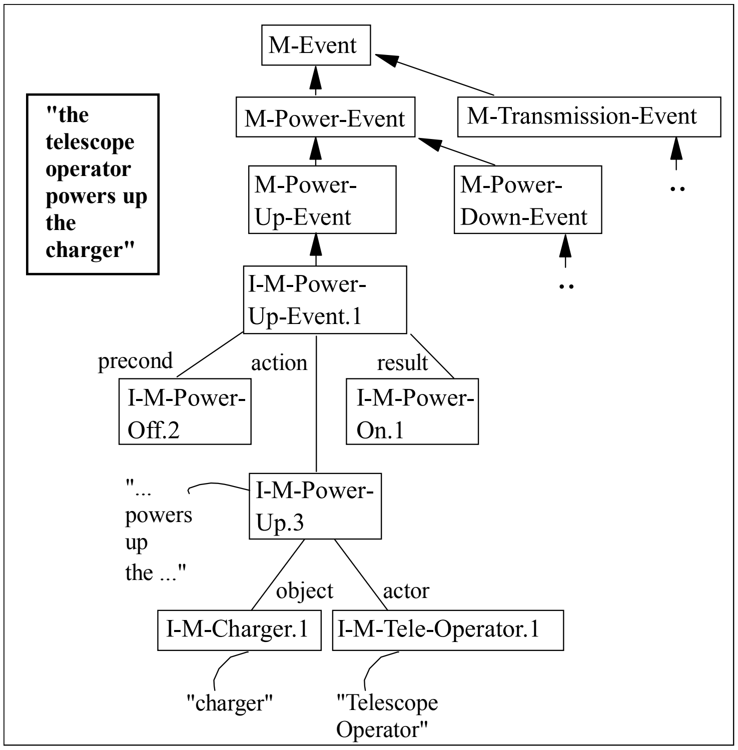
The notion of frames by Marvin Minsky took the idea of semantic networks and applied a more formal object-oriented inheritance structure (I've already been doing this to some extent in the previous example). Instead of arbitrary nodes connected together, a node is considered to be a **frame**. A frame contains attribute/value pairs, called **slots**. Each value may be some other frame. The collection of frames makes up a knowledge system. Inheritance is an integral property of most frame-based systems. Note that most of these ideas are now common in modern object-oriented programming languages like C++ and Java.

This is essentially the same as what we've been doing so far. Keep in mind that values may be other frames, resulting in a framework of knowledge that allows easy abstraction and construction of concepts based upon sub-concepts and sub-procedures.

Examples of frame-based knowledge from FANSYS: Failure Analysis System for NASA's proposed space station and also for SOFIA, a replacement for the Kuiper Airborne Observatory:



Here, two states are represented, one for the state of power (on or off), and another for what objects are connected to one another (physically or logically). In this case we actually parsed from input text, such as "charger" to map into the instance of a charger. M stands for MOP, or Memory Organization Packet, which is a type of frame. For example, the frame for "Physically Connected" has two slots. The first slot is the attribute of object1, and the value is I-M-HPAS.2, an instance of the High Pressure Air System. The second slot is the attribute of object2 and the value is the MOP for I-M-Compressor 5.



Another MOP or frame that we created was for Events. In this example, the memory knowledge for "The telescope operator powers up the charger" is represented by an instance of the Power-Up-Event. This event has three slots. The first slot has an attribute of precondition, and a value of I-M-Power.Off.2. The second slot has the attribute of action, and the value of I-M-Power-Up.3, the actual powering-up action. Finally, the third slot has the attribute of result, and the value of I-M-Power.On.1. Note that this MOP/frame is the same as the one previously defined. MOPs/Frames can then be shared wherever appropriate, resulting in shared knowledge and the construction of more complex abstractions.

These have been examples of using Frames to represent knowledge. Frames represent explicit, visual representations of hierarchical object-based knowledge. By including arbitrary relations as possible nodes or frames, then we can represent fairly general knowledge constructs and also perform reasoning. Spreading activation and hierarchy search can also be used to find specific knowledge constructs.