

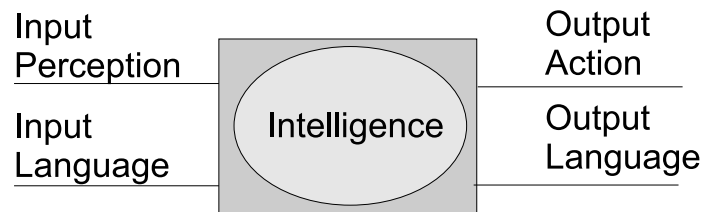
CS 8803B: Artificial Intelligence

Class notes for 11/08/02: Input

Notes taken by: Andreas Lachenmann

Input and Output as a Part of Intelligence

Intelligence can be represented like this:



Data from other agents or the world is the input. This data can be an image, for example. In this class we have so far mainly looked at the center part of this figure. However, input and output is part of intelligence, too. In the next couple of lectures we will focus on the different forms of input.

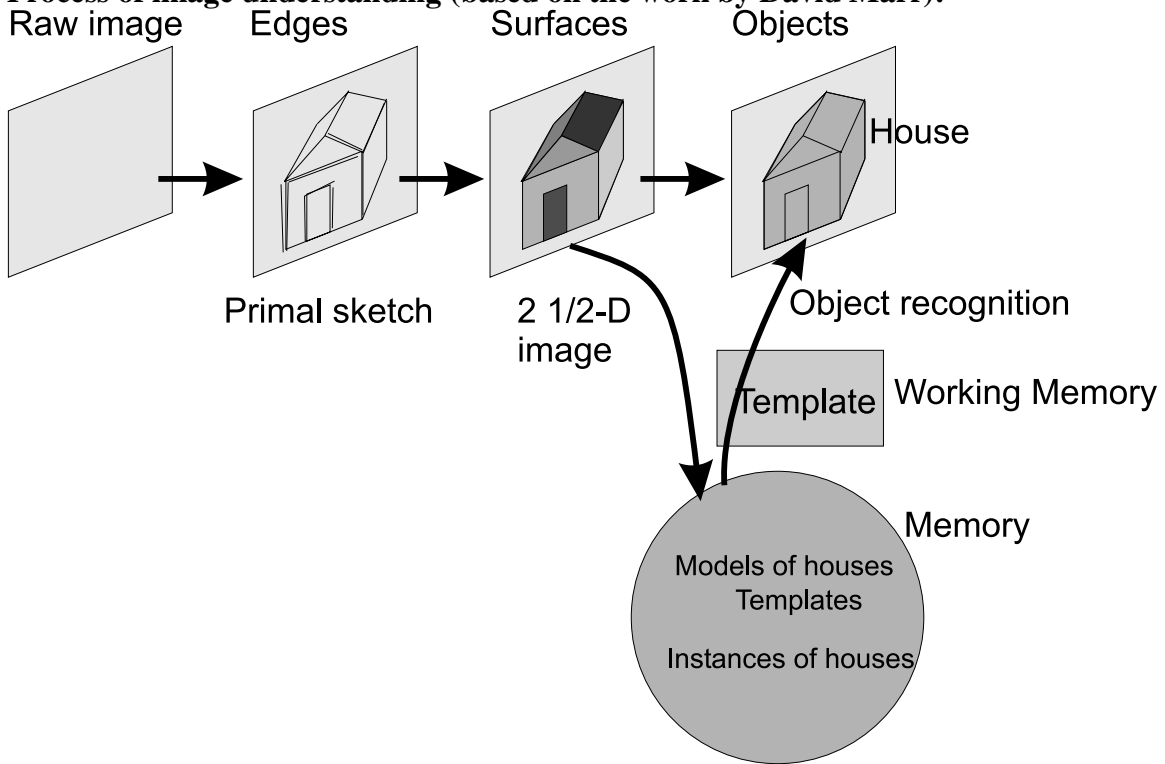
A textbook normally describes vision and language separately. Here we will look at both at the same time because they are very similar to each other.

In vision the input could be a movie/video, a single image, or a line drawing. It is not that clear what the output is. It should be the “meaning” of the image. The output could be an object shown in the image or an action or language. In language the input is a text/paragraph, a single sentence, or a word. Here again the output is difficult to define. It should be the “meaning” expressed in an action or in language.

Single sentences in language correspond to single images in vision. Paragraphs and texts are like movies: both consist of a sequence, the former of sentences and the latter of images. In both cases the interpretation of the current item depends on the preceding one. Another similarity is that there is a problem of defining the right output because the task of determining the meaning is difficult.

Vision

Process of image understanding (based on the work by David Marr):



The raw image is abstracted first into its edges. Normally, there is some noise in the image; many more edges are recognized than the ones actually forming the surfaces of the object. Then the surfaces are identified. Based on the information in the memory an object (the house) is recognized.

The abstractions are necessary because it would be too complex to compare all the pixels of the raw image. Therefore, this process aggregates and abstracts until the comparison is easy because the abstraction is sufficiently small.

Language

There are different approaches which address the problem of understanding language:

- Lexical Analysis
- Syntactic Analysis
- Semantic Analysis

Lexical Analysis:

Consider the sentence “*He ate a frog.*” Lexical analysis looks up the type of each word in a dictionary, the lexicon:

He	Pronoun
ate	Verb
a	Article
frog.	Noun

Syntactic Analysis:

The input of this analysis is a sentence. The output is a parse tree which is generated by using a grammar.

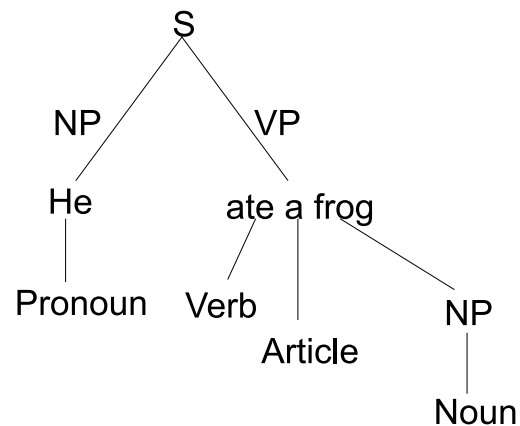
In the grammar “|” denotes an alternative and “[]” an optional part.

Grammar:

$S \rightarrow NP VP$

$NP \rightarrow \text{noun} \mid \text{pronoun}$

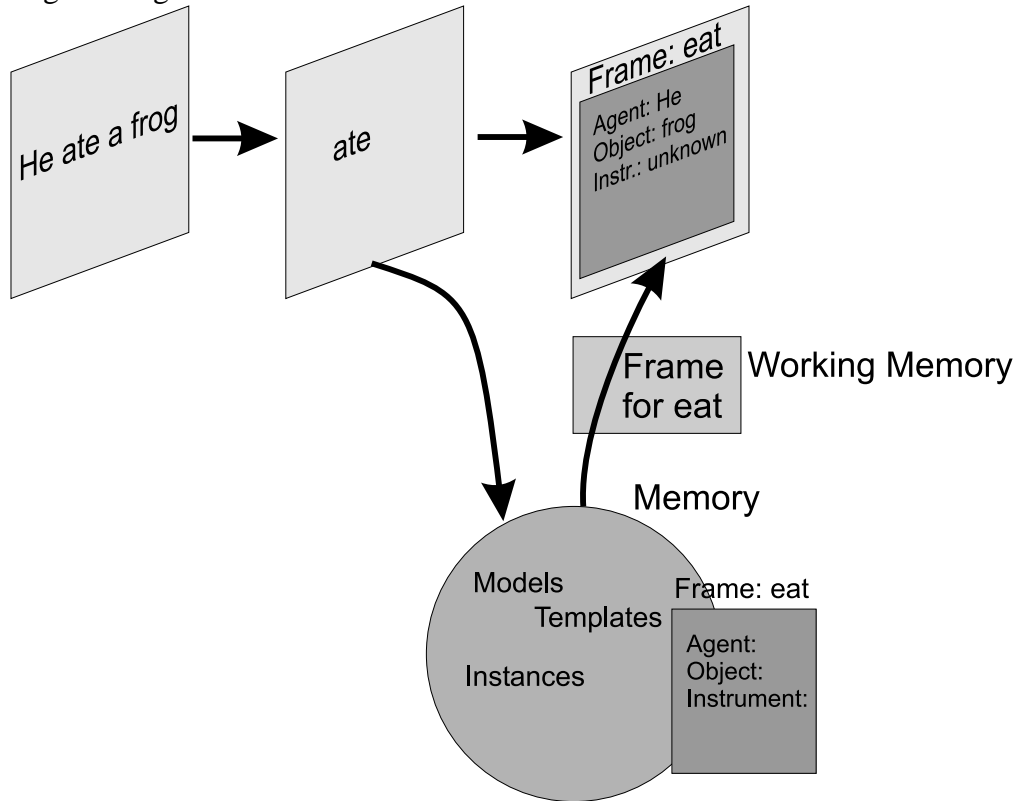
$VP \rightarrow \text{verb} [\text{article}] NP$



Semantic Analysis:

The output of this analysis are frames, scripts, or semantic networks. People in vision are interested in semantic analysis most of the time.

The process of analyzing a sentence can be illustrated similarly to the process of analyzing an image in vision:



Here again several abstractions are built. The frame which is retrieved from memory contains some information on how to organize the input around the verb. In understanding a sentence you are not interested in an object but in an activity. In vision you could be interested in an activity, too – an image of children playing soccer, for example.

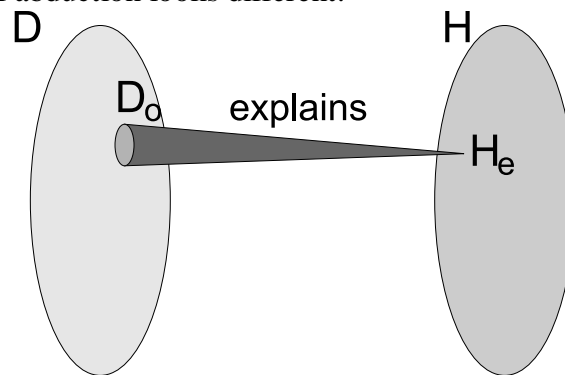
The input is data and the output is a cause or an object which explains the data.

Abduction

The reason for the similarities is that a similar pattern is going on. Last time when we looked at abduction it was presented like this:

$$\frac{\text{If } P \text{ then } Q}{Q} \\ P?$$

A set theoretic view of abduction looks different:



D is the data set of all possible symptoms, for example, in medical diagnosis. The set H consists of all possible hypotheses. In the example this would be the diseases. D_o contains the observed data (the symptoms of a patient). H_e is the set of hypotheses (possible diseases) which explain these. D_o is given; H_e contains all possible causes for it.

In vision you start with the effect, the image, and ask: “What caused the image? – A house.” In this case D is the set of all possible raw images. You look for a small number of hypotheses – if possible just 1 – which explain the image.

In language you are interested in questions like “What did the author want to convey? What is the meaning of the sentence?” Here D is the set of all possible sentences. You look for a cause or explanation that explains the sentence.

When building an intelligent agent, would it make sense to have a common architecture for vision and language? Or even for all kinds of perception? The answer is no because in vision there is the huge problem of noise that does not exist in language. Noise means that the image is blurry, the location of objects is unclear, etc. In language there are sometimes problems with ambiguity:

In

“John and Mike went to a restaurant. He ate a frog.”

and even in

“John went to the restaurant where Mike works. He ate a frog.”

it is unclear who “He” refers to. This is the reference problem. In vision there exists a similar problem: Is it really the same object that is shown in several images?

Analysis of the Meaning of a Sentence

There are several possibilities in which order the different kinds of language analysis could be executed. For example, syntactic analysis could be regarded more important than semantic analysis and therefore be executed before the other one, or vice-versa. If the semantic analysis is processed first, it is in control and lexical and syntactic enter later. The input of this analysis can be a single sentence; the output can be the meaning as it is captured in a frame.

Consider the sentence "*I go to the beach.*" It is processed from left to right. First, the verb is searched. Each word that is not the verb is kept in memory for later processing. In some sentences the verb can be placed very late ("*To the beach I go.*"). Those sentences too are not processed until the verb is found.

When the verb is found, the following frame is pulled out from memory. It is coupled with a set of rules/exceptions (R1, ..., R11):

go

Agent:	R1: Whatever comes before "go", put it in the agent slot.	I
Co-Agent:		
Beneficiary:		
Place:	R9: Find concept or word after "to". R10: Whatever comes after "to", is the place.	the beach
Time:		
Instrument:	R11: Find the preposition "with".	

The sentence is processed both bottom-up (from the data) and top-down (from memory). In bottom-up analysis each word is processed from left to right and transferred to the working memory. Top-down analysis means that all frame rules are checked if they are activated. This step accesses the data in the working memory.

So far not syntactic or lexical analysis has been conducted because the rule set of this example is very crude. This will be covered in the next lecture.