Ch. 14: Meaning representations

SEMANTIC ANALYSIS

Why bother parsing sentences (i.e., finding a formal representation of their structure)?

- To know whether a given sentence can be derived from a given grammar.

That's theoretically interesting, but does it have a practical application.

- To get practical applications, we need to be able to derive a meaning for the sentence from the output of the parser.

We need some way to represent that meaning. There are many such ways--we call them *meaning representations*.

Semantic analysis = any algorithm for deriving a meaning rep. from a syntactic rep. (tree structure).

Semantic analyzer = a program that implements such an algorithm.

PRAGMATICS & WORLD KNOWLEDGE

To get a useful meaning, we need not only the literal meaning of the sentence but its meaning in context = its *pragmatic meaning*.

To get pragmatic meaning, we also need *world knowledge* = non-linguistic knowledge about the world that human beings have but computers (i.e., programs) don't.

See examples on p. 502: grading essays, discussing restaurants, installing software from a manual.

Of these, only essay grading is currently done by machine, and it's done with a numerical algorithm (!), not using the methods described in this chapter.

Applications people are actually working on today: telephone bills (AT&T's "How may I help you?" application), airplane reservations.

EXAMPLES OF MEANING REPS.

Figure 14.1 = 4 kinds of meaning reps.:

- FOPC (first-order predicate calculus)
- Semantic network
- Conceptual dependency (CD) diagram
- Frame

Each of them shows 4 pieces of information: how?

Which can be directly implemented on a computer? Why?

What is the difference between the CD rep. and the semantic network?

In the FOPC rep., why do we need a separate predicate for Car and not for Speaker? I.e., why don't we write:

$\exists x,y,z$. having(x) $\wedge$ haver(z,x) $\wedge$ hadthing(y,x) $\wedge$ car(y) $\wedge$ speaker(z)

Better names for these predicates:

$\exists x,y$. event(x) $\wedge$ possessor(speaker,x) $\wedge$ object(y,x) $\wedge$ is-car(y)

or

$\exists x,y$. event(x) $\wedge$ possessor(speaker,x) $\wedge$ object(y,x) $\wedge$ is-a(y, car)

AMBIGUITY VS. UNDERSPECIFICATION VS. VAGUENESS

Consider some everyday sentences:

I want to eat near the office.
I want to eat Italian food.
etc.

What is ambiguity?

What is underspecification?

What is vagueness?

Gray areas between these terms.

Which terms apply to which sentences?

CANONICAL FORM

Does X have vegetarian dishes?
Do they serve vegetarian food at X?
Is X a vegetarian restaurant?
Visiting professor Y is a vegetarian. Can I take her to X?

What is a canonical form?

Why is it useful?

How many canonical forms are needed to represent the sentences above?

How much detail should go into the canonical form?

Does having a canonical form enable a computer (i.e., a program) to answer the above questions? NO!

Many reasons why not:
- Need for a knowledge base
- The world knowledge problem
- Ambiguity, underspecification and vagueness
- Context (especially in dialogue)

More about the world knowledge problem:
- Where are we going to get the knowledge from
- How are we going to represent it