Semantic Technology

John F. Sowa

VivoMind Intelligence, Inc.

4 March 2008

What is Semantic Technology?

A common misconception: annotating web pages with XML tags.

Semantics is the study of meaning, natural languages are the ultimate semantic notations, and all XML tags must be related to language.

But there are many useful ways of processing language short of full understanding.

It's important to decide what level of processing is required for various purposes.

And it's also important to find ways of designing computer systems that can come much closer to the goal of understanding language.

For further discussion of these issues, see the following paper by Yorick Wilks:

> **http://www.dcs.shef.ac.uk/~yorick/papers/IEEE.SW.untrak.pdf**
> **The Semantic Web as the apotheosis of annotation, but what are its semantics?**

Syntax, Semantics, and Pragmatics

Syntax: the way words are put together to form a sentence.

Semantics: the information encoded by that sentence.

Pragmatics: the implications of that information in the context.

Full understanding requires the listener to

1. relate the new information to background knowledge,
2. determine what the speaker intended,
3. decide how to respond to that intention,
4. respond according to the listener's intentions.

Background Knowledge

Five kinds of knowledge needed for understanding language:

1. Animal-level knowledge: Feelings, parts of the body, ways of moving and interacting, and ways of satisfying desires.

2. Lexical knowledge: General knowledge about words, word senses, and typical patterns of relations among words.

3. Episodic knowledge: Specific knowledge about situations, including the people, places, things, and events involved.

4. Task-oriented knowledge: Knowledge about a task, how to do it, and the requirements for doing it.

5. Wittgenstein's language games: Knowledge about a stylized way of using language in a particular task or activity.

Multiple Word Senses

Four sentences that use the verb *support* in the same syntactic pattern:

- ***Tom supported the tomato plant with a stick.***
- ***Tom supported his daughter with $20,000 per year.***
- ***Tom supported his father with a decisive argument.***
- ***Tom supported his partner with a bid of 3 spades.***

Using Lexical Knowledge

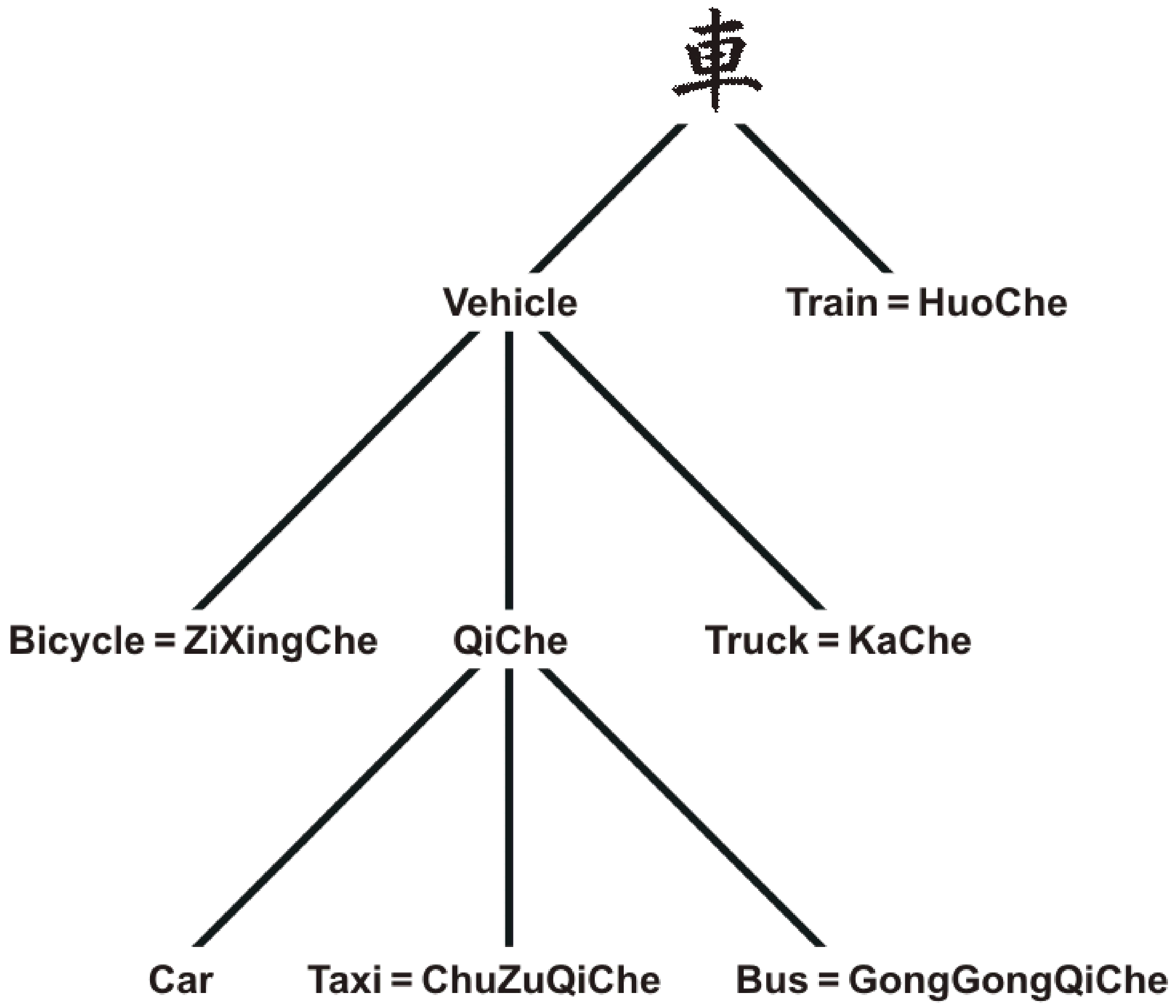
A sentence with an ambiguous word:

John went to the board.

Word sense determined by context:

- ***He kicked it to see if it was rotten.***
- ***He wrote the homework assignment for the next week.***
- ***He presented arguments for the proposed merger.***

Cross-Linguistic Variations



Task-Oriented Knowledge

A driver stops the car to ask a question:

Driver: Where is the nearest gas station?

Pedestrian: Turn right at the corner, and go about two blocks.

D: Thanks.

A short while later, the driver comes back:

D: That gas station is closed.

P: Yes. It hasn't been open for years.

D: Why didn't you tell me?

P: You didn't ask.

The pedestrian did not have or use task-dependent knowledge about cars, drivers, and gas stations.

Organizing Background Knowledge

Methods for organizing and representing knowledge can be combined in various ways:

1. **Lexical semantics:**
    - **Knowledge about words and patterns of words and relations.**
2. **Axiomatized semantics:**
    - **Formal rules or axioms intended for deduction.**
3. **Statistical approaches:**
    - **Empirical techniques derive information from large corpora.**
4. **Task-oriented semantic interpretation:**
    - **Large amount of general lexical knowledge.**
    - **But axioms specialized for each task.**

Common Logic

An ISO/IEC standard for representing logic.

A superset of most common versions of logic and logic-based notations, including RDF(S), OWL, and RuleML.

Defined by an abstract syntax and model theory.

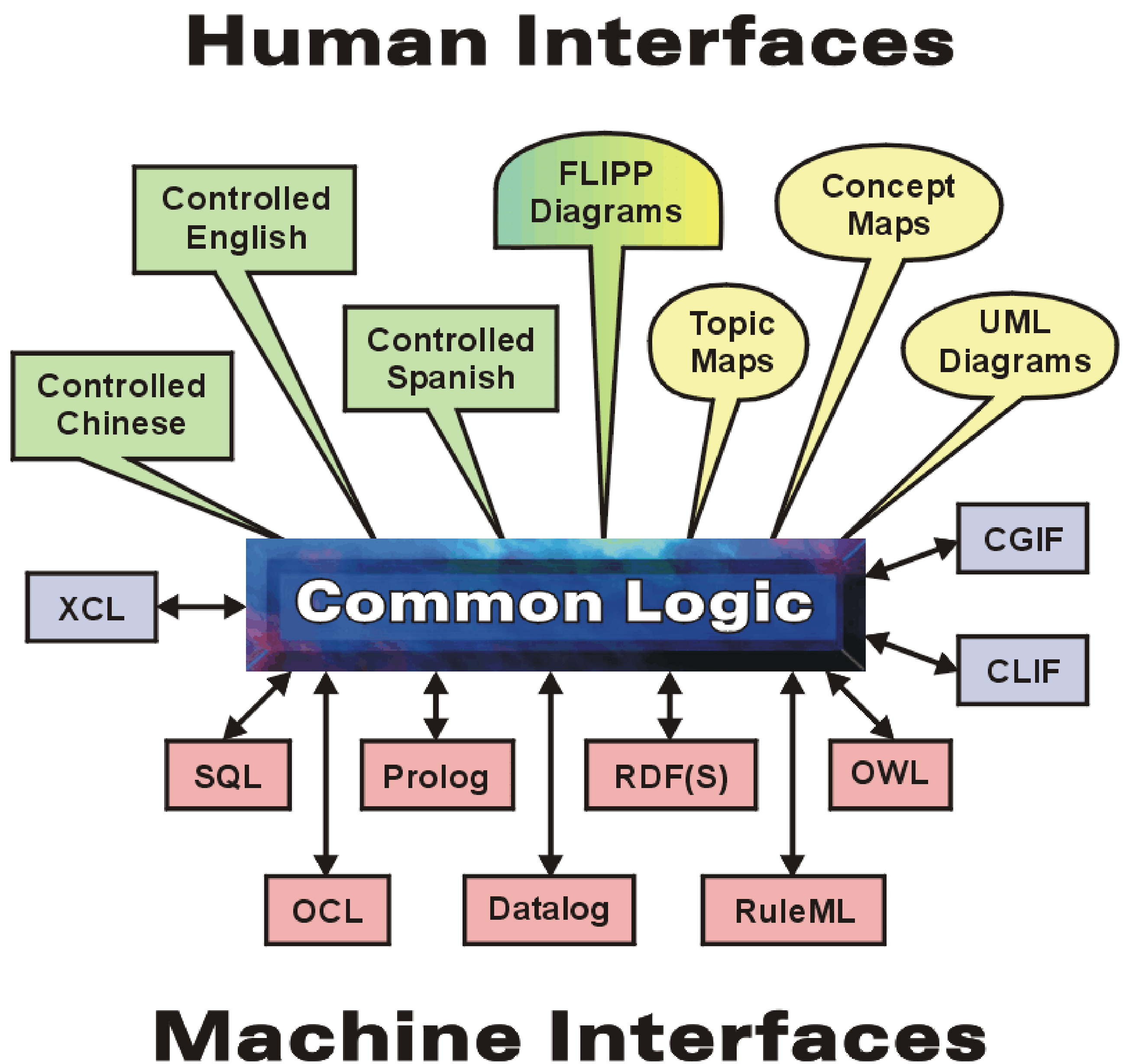
Three concrete syntaxes specified in the standard:

- CGIF — Conceptual Graph Interchange Format
- CLIF — Common Logic Interchange Format
- XCL — XML-based notation for Common Logic

Suitable for mapping to and from controlled natural languages.

One example is Common Logic Controlled English (CLCE).

Web site: http://www.common-logic.org/


Human Interfaces
Controlled English
FLIPP Diagrams
Concept Maps
Controlled Spanish
Topic Maps
UML Diagrams
Controlled Chinese
CGIF
XCL
Common Logic
CLIF
SQL
Prolog
RDF(S)
OWL
OCL
Datalog
RuleML
Machine Interfaces

Common Logic Controlled English

A dialect of Common Logic that looks like English.

CLCE uses a subset of English syntax and vocabulary.

But the CLCE grammar avoids constructions that may cause ambiguities.

CLCE replaces pronouns with short-lived names called variables.

Examples:

> For every company C,
> exactly one manager in C is the CEO of C;
> every employee of C except the CEO reports to the CEO;
> the CEO of C does not report to any employee of C.
>
> If an integer N is 5, then (N**3 = 125).

The scope of the variable C extends to the period at the end.

The rules for the scope of variables are a formally specified subset of the rules for pronoun scope in English.

CLCE allows arithmetic expressions, enclosed in parentheses, to use variables defined outside the parentheses.

CLCE Semantics

CLCE can express the full semantics of Common Logic.

A recursive definition of "reports" in terms of "directly reports":

> Every employee who directly reports to a manager
> reports to that manager.
>
> If an employee of a company C directly reports to a manager M1 in C,
> and the manager M1 reports to a manager M2 in C,
> then the employee reports to the manager M2.

Definitions link CLCE words and phrases to other languages and logicsv:

> Define "x directly reports to y" as
> (DirectlyReports x y).
>
> Define "x directly reports to y" as
> sql(SELECT EMP, MGR FROM EMPLOYEES).

The defining and defined expressions may be stated in any version of logic:

- CLCE expressions are enclosed in double quotes.
- CLIF expressions begin with "(".
- CGIF expressions begin with "[".
- XCL expressions begin with "<".

An Extension Beyond Common Logic

An extension to CL called IKL (Interoperable Knowledge Language) adds one more operator to support metalevel statements.

An IKL statement expressed in CLCE notation:

Tom believes that Mary knows that (4 = 2 + 2).

And in CLIF notation:

```
(believes Tom (that (knows Mary (that (= 4 (+ 2 2))))))
```

The operator "that" is one very short, but very powerful extension.

It enables IKL to specify various languages, axiomatize their semantics, and define transformations from one language to another.

Those definitions can be stated in CLCE and other CL dialects.

Methods of Reasoning

- **Deduction: Applying a general principle to a special case.**
- **Induction: Deriving a general principle from special cases.**
- **Abduction: Guessing that some general principle can relate a given pattern of cases.**
- **Analogy: Finding a common pattern in different cases.**

Problem with Deduction

Deduction is the method of reasoning at the foundation of most database and knowledge-based systems.

Deduction is precise, predictable, and brittle.

If everything is perfect, deduction is perfect.

If anything in the system is imperfect, deduction can magnify and propagate the imperfection to the point of a total collapse.

When multiple systems interoperate, the likelihood and the danger of imperfection escalates.

How do People Avoid the Problems?

They don’t always avoid them, but there are some safeguards:

- **People seldom carry out long chains of deductions.**
- **When the conclusion seems odd, they check the facts, ask for advice, and perform a “sanity check.”**
- **They communicate by message passing.**
- **They don’t expect every message to be completely understood.**
- **They ask questions, give explanations, negotiate, and compromise.**

Some of the greatest disasters in history were caused by people who demanded and expected computer-like perfection.

Suggested Solution

Treat computers as if they're no better than humans:

- Don't assume that all systems have identical ontologies, similar ontologies, or any explicit ontology of any kind.
- Don't require agreement on anything outside the specific task.
- Insist on sanity checks and the option to back out gracefully.
- Communicate by message passing and dialog, not direct orders.
- Develop methods of negotiation, compromise, and voting.

This approach requires reasoning methods that are more flexible and robust than strict deduction.

Technologies to Consider

Natural languages:

- The ultimate knowledge representation languages.
- Capable of representing anything in human experience.
- Highly flexible and adaptable to changing circumstances.
- But not easy to implement on digital computers.

Formal logics and ontologies:

- Precise and implementable on digital computers.
- Can be translated to natural languages.
- But inflexible, brittle, and uncompromising.

Statistical methods:

- Flexible, robust, and designed to handle uncertainty.
- But there is an open-ended variety of different methods.
- Not clear how to relate them to language, logic, and ontology.

Research issue: Find suitable combinations of the above.

Suggested Readings

John F. Sowa, "Conceptual Graphs," in F. van Harmelen, V. Lifschitz, and B. Porter, eds., *Handbook of Knowledge Representation*, Elsevier, 2008, pp. 213-237. http://www.jfsowa.com/cg/cg_hbook.pdf

John F. Sowa, a lecture on Common Logic and CLCE. http://jfsowa.com/talks/cl_sowa.pdf

John F. Sowa, "Categorization in Cognitive Computer Science," in H. Cohen & C. Lefebvre, eds., *Handbook of Categorization in Cognitive Science*, Elsevier, 2006, pp. 141-163. http://www.jfsowa.com/pubs/cogcat.htm

John F. Sowa & Arun K. Majumdar, "Analogical reasoning," in de Moor, Lex, Ganter, eds., *Conceptual Structures for Knowledge Creation and Communication*, Proceedings of ICCS 2003, LNAI 2746, Springer-Verlag, Berlin, 2003, pp. 16-36. http://www.jfsowa.com/pubs/analog.htm

John F. Sowa, "The Challenge of Knowledge Soup," in J. Ramadas & S. Chunawala, *Research Trends in Science, Technology and Mathematics Education*, Homi Bhabha Centre, Mumbai, 2005, pp. 55-90. http://www.jfsowa.com/pubs/challenge.pdf

John F. Sowa, "A Dynamic Theory of Ontology," in B. Bennett & C. Fellbaum, eds., *Formal Ontology in Information Systems*, IOS Press, Amsterdam, 2006, pp. 204-213. http://www.jfsowa.com/pubs/dynonto.htm

John F. Sowa, "Peirce's contributions to the 21st Century," in H. Schärfe, P. Hitzler, & P. Øhrstrøm, eds., *Conceptual Structures: Inspiration and Application*, LNAI 4068, Springer, Berlin, 2006, pp. 54-69. http://www.jfsowa.com/pubs/csp21st.pdf

John F. Sowa, "Worlds, Models, and Descriptions," *Studia Logica*, Special Issue Ways of Worlds II, 84:2, 2006, pp. 323-360. http://www.jfsowa.com/pubs/worlds.pdf

Yorick Wilks, "The Semantic Web as the apotheosis of annotation, but what are its semantics?" http://www.dcs.shef.ac.uk/~yorick/papers/IEEE.SW.untrak.pdf