What is Natural Language Semantics?

Dealing with and reasoning about the meaning and use of natural languages (as opposed to computer languages.)

Two kinds of approaches:

- Formal/logical - inspired by work in logic and philosophy of language: try to treat natural languages like English as if they were formal ones

- Empirical/distributional - originally based on some early linguistic theories, but now incorporates ideas from information retrieval, neural networks, etc.
A formal approach to semantics

Don’t ask ‘what is meaning?’. This isn’t a question that it is possible to answer usefully. Instead ask ‘What do you know when you understand a sentence?’

- (usually) what ‘proposition’ it expresses (the same sentence can express different propositions on different occasions)
- what the world would be like if the proposition expressed were true (truth conditions) (not the same as knowing whether it is true)
- what other propositions also then have to be true if this one is (entailments)
- what is contextually implied by the utterance of that sentence (pragmatic inferences)
Semantics = truth conditions + inference

We can distinguish different semantic classes of word according to the inferences that they allow: for example, adjectives, although all very similar syntactically, fall into several different subclasses semantically.

**Intersective:**

Jones is a Welsh lawyer. → Jones is Welsh
→ Jones is a lawyer

All lawyers are musicians. → Jones is a Welsh musician.

**Privative:**

Jones is a former lawyer. \(\not\rightarrow\) Jones is a lawyer

This is a fake gun \(\not\rightarrow\) This is a gun

**Gradeable:**

Jones is a skilful lawyer. → Jones is a lawyer.
→ Jones is skilful???

All lawyers are musicians. → Jones is a musician.
\(\not\rightarrow\) Jones is a skilful musician.

Minnie is a large mouse. → Minnie is a mouse
→ Minnie is an animal
\(\not\rightarrow\) Minnie is a large animal.
### Structurally based inference

#### Quantifiers

<table>
<thead>
<tr>
<th>Statement</th>
<th>Implication</th>
</tr>
</thead>
<tbody>
<tr>
<td>All men are mortal</td>
<td>Some men are mortal</td>
</tr>
<tr>
<td>Not all dogs bark</td>
<td>At least one dog does not bark</td>
</tr>
<tr>
<td><strong>‘Downward’ monotone:</strong></td>
<td></td>
</tr>
<tr>
<td>No fish can breathe air</td>
<td>No large fish can breathe air</td>
</tr>
<tr>
<td><strong>‘Upward’ monotone:</strong></td>
<td></td>
</tr>
<tr>
<td>Some large fish migrate</td>
<td>Some fish migrate</td>
</tr>
</tbody>
</table>

#### Comparatives

<table>
<thead>
<tr>
<th>Statement</th>
<th>Implication</th>
</tr>
</thead>
<tbody>
<tr>
<td>John is taller than Bill</td>
<td>Bill is less tall than John</td>
</tr>
<tr>
<td></td>
<td>Bill is not as tall as John</td>
</tr>
<tr>
<td></td>
<td>John is tall (without further information)</td>
</tr>
<tr>
<td></td>
<td>John is tall.</td>
</tr>
<tr>
<td>Bill is tall, and John is taller than Bill</td>
<td>John is tall.</td>
</tr>
</tbody>
</table>
Semantically related structures

John will leave unless Mary leaves first
≡ If Mary does not leave first, John will leave.
Joe is too short to reach the shelf
≡ Joe is not tall enough to reach the shelf

Existential (non)-commitment

Jones won a Nobel prize
→ Someone won a prize
→ Jones won something
→ Someone won something.

Jim wants to marry a Norwegian girl
✓ there is a Norwegian girl Jim wants to marry

Fred is looking for a unicorn
✓ there are unicorns
## Lexically based inference

### Lexical relations

<table>
<thead>
<tr>
<th>X murdered Y</th>
<th>→ X killed Y</th>
<th>→ Y died.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X melted the chocolate</td>
<td>→ the chocolate melted.</td>
<td></td>
</tr>
<tr>
<td>Fido is a dog</td>
<td>→ Fido is an animal.</td>
<td></td>
</tr>
<tr>
<td>Fido is sad</td>
<td>→ Fido is not happy</td>
<td></td>
</tr>
</tbody>
</table>

### Presupposition

The King of France is (is not) bald

~~> There is a King of France

John regrets (doesn’t regret) that his dog died

~~> His dog died

When did you start/stop defrauding your boss?

~~> You have defrauded your boss

Notice that presupposition cannot be the same as entailment.
Contextual reference resolution

A: Jones is a lawyer.
B: No, he isn’t. He’s a policeman.

He = Jones
He isn’t = Jones is not a lawyer.

Ellipsis and pronoun meanings need to be filled in from prior context - but much interaction with non-linguistic knowledge:

- John gave Mary two vintage bottles of wine, but one of them was undrinkable. They were very disappointed/expensive.
- James gave each boy a trumpet. They made a terrible noise.
- Every college employs a gardener. They pay them badly.
- Jones finished his homework before Smith did.
- The porters refused the students admission because they feared/advocated violence.
Conversational implicatures

H. P. Grice’s ‘Cooperative principle’:  
“Make your contribution such as is required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which you are engaged”

Cooperative conversations follow 4 maxims:

<table>
<thead>
<tr>
<th>Maxims of Quantity:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ”Make your contribution as informative as required.”</td>
</tr>
<tr>
<td>2. ”Don’t make your contribution more informative than is required.”</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Maxims of Quality:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ”Don’t say what you believe to be false.”</td>
</tr>
<tr>
<td>2. ”Don’t say what you lack adequate evidence for.”</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Maxim of Relation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>”Be relevant.”</td>
</tr>
</tbody>
</table>
Maxims of Manner:

1. "Avoid obscurity of expression."
2. "Avoid ambiguity."
3. "Be brief (avoid unnecessary prolixity)."
4. "Be orderly."

Of course, you can fail to follow one or more of these maxims, and in two ways:

- **quietly** → effect is to mislead or lie
- **obviously** → effect is to generate a ‘conversational implicature’

Implicatures, unlike entailments, are **cancellable**

Some constructs have ‘conventional’ implicatures associated with them, which are often difficult to cancel.
Conversational implicatures

- A. I’m out of petrol. B. There’s a garage round the corner. (Assumption of relevance)
- A. How’s the new job? B. That’s confidential. (Opting out)
- A. Vienna is in Italy, isn’t it? B. And I’m the Prime Minister. (Ostentatious flouting of quality)
- A. What is Smith like as a student? B. His handwriting is excellent. (Ostentious flouting of quantity).
- A. The singer produced a series of sounds closely resembling ‘My Way’. (Obscurity)
- A. Where’s the pie? B. The dog is looking very happy. (Relation)
Contextual influence on interpretation. What is meant can be more than what is said.

\[ A: \text{Would you like some coffee?} \]
\[ B: \text{It would keep me awake.} \]

A’s question really demands the answer ‘yes’ or ‘no’, so violates a maxim. Given a context, we can deduce an answer, but the answer will depend on what the context is.

Context 1: B is tired. A reasons that B probably wants to sleep, so does not want anything that would prevent this: therefore ‘No’.

Context 2: B wants to work late revising for an exam. A reasons that B does not want to fall asleep, so would like anything that would prevent this: therefore ‘Yes’.

Can be seen as a kind of abductive reasoning:
\[ \text{utterance } U + \text{ assumption } A \rightarrow \text{ conclusion } C. \]
Both A and C are conveyed by U.
Recognising Textual Entailment

RTE is a challenging task: to determine whether or not a hypothesis follows from a text:

Israel’s Prime Minister, Ariel Sharon, visited Prague. →

Ariel Sharon is Israel’s Prime Minister. (YES)

Reagan attended a ceremony in Washington to commemorate the landings in Normandy. →

Washington is located in Normandy. (NO)

The Republic of Yemen is an Arab, Islamic and independent sovereign state whose integrity is inviolable, and no part of which may be ceded. →

The national language of Yemen is Arabic. (YES)

The notion of entailment here is not a strictly logical one, but ‘T entails H if, typically, a human reading T would infer that H is most likely true’. The inference may involve non-linguistic knowledge.
What South-American country has the largest population?
What’s the largest city in Florida?
What is Jupiter’s atmosphere made of?
Who first discovered radiocarbon dating?
When did Bobby Kennedy die?

First Google search is Wikipedia:

*Robert Francis "Bobby" Kennedy (November 20, 1925 June 6, 1968), commonly known by his initials RFK, was an American politician from Massachusetts. He served as a Senator for New York from 1965 until his assassination in 1968.*

For an early QA demo: http://start.csail.mit.edu/index.php
We saw that an important part of meaning can be captured by the notions of truth conditions and entailments.

So a semantic theory should tell us, for each sentence of a language (or the proposition expressed by it in a particular context) what its truth conditions are and what is entailed by it.

Although first order predicate calculus does not have the context-dependent property of natural languages, in other respects it is is a miniature model of what we want:

- The syntax of FOPC characterises an infinite set of well-formed sentences.
- To each of these it assigns, recursively, truth conditions via the interpretation function for FOPC.
- Via its proof theory (or its semantics) it characterises the notion of valid inference for these sentences.
John likes Mary but Mary doesn't trust him. Translate as:

\[
\text{like}(\text{John}, \text{Mary}) \land \neg\text{trust}(\text{Mary}, \text{John})
\]

Assume the usual kind of interpretation function ($\text{Int}$) for FOPC, associating expressions with set theoretic constructs:

**$\text{Int}$ for constants**

\[
\begin{align*}
\text{Int}(\text{John}/\text{Mary}) & = \text{the actual person John/Mary} \\
\text{Int}(\text{likes}) & = \{ \langle x, y \rangle \mid x \text{ likes } y \} \\
\text{Int}(\text{snores}) & = \{ x \mid x \text{ snores } \} \\
\text{etc.}
\end{align*}
\]

**$\text{Int}$ for wff**

\[
\begin{align*}
\text{Int}(\text{likes}(i,j)) & = \text{true iff } \langle \text{Int}(i), \text{Int}(j) \rangle \in \text{Int}(\text{likes}), \text{ etc.} \\
\text{Int}(S1 \land S2) & = \text{true iff } \text{Int}(S1) = \text{true and } \text{Int}(S2) = \text{true.} \\
\text{Int}(\neg S) & = \text{true iff } \text{Int}(S) = \text{false.}
\end{align*}
\]
So: like(John, Mary) \land \neg\text{trust}(Mary, John) is true iff
(A) like(John, Mary) is true,
and
(B) \neg\text{trust}(Mary, John) is true.

(A) like(John, Mary) is true iff \langle Int(John), Int(Mary) \rangle \in Int(likes)

(B) \neg\text{trust}(Mary, John) is true iff trust(Mary, John) is false

trust(Mary, John) is false iff \langle Int(Mary), Int(John) \rangle \notin Int(trusts)

The semantic interpretation function tells us what the world must be like if the sentence is true. If we had a particular domain fixed we could evaluate this expression to see whether it actually is true in that domain.
FOPC-style semantics for a Fragment of English

1. \( S \rightarrow NP \ VP \quad T \text{ iff } Int(NP) \in Int(VP) \)
2. \( NP \rightarrow \text{Name} \quad Int(NP)=Int(\text{Name}) \)
3. \( VP \rightarrow V_{intr} \quad Int(VP)=Int(V) \)
4. \( VP \rightarrow V_{trans} \ NP \quad Int(VP)=\{X \mid \langle X, Int(NP) \rangle \in Int(V) \} \)
5. \( V_{intr} \rightarrow \text{snores} \quad Int(Vi)=\{X \mid X \text{ snores} \} \text{ etc.} \)
6. \( \text{Name} \rightarrow \text{John} \quad Int(\text{John}) \text{ etc.} \)
7. \( V_{tran} \rightarrow \text{likes} \quad Int(Vt)=\{\langle X,Y \rangle \mid X \text{ likes } Y \} \)
8. \( S \rightarrow S \text{ and } S \quad T \text{ iff both daughter } S \text{ are true} \)
9. \( VP \rightarrow VP \text{ and } VP \quad Int(VP_0)=\{X \mid X \in Int(VP_1) \text{ and } \in Int(VP_2) \} \)
10. \( VP \rightarrow \text{doesn't } VP \quad Int(VP_0)=\{X \mid X \text{ is not in } Int(VP_1) \} \)

Example domain = \{john, bill, mary, sue\}
\( Int(\text{likes}) = \{\langle \text{john, mary} \rangle, \langle \text{sue, mary} \rangle \} \); \( Int(\text{snores}) = \{\text{john, sue}\} \);
\( Int(\text{John}) = \text{john, etc.} \).
But giving the semantics this way is very complicated. In practice it is easier to give the semantics indirectly via translation into logical form:

Sentence $\rightarrow$ Parse Tree $\rightarrow$ Logical Form $\rightarrow$ Truth Conditions

It has been argued that there are linguistic grounds for such a level of representation anyway:

*A dog barked. It was hungry.*

*Not every dog didn’t bark. *It was hungry.*

But ‘A dog barked’ and ‘Not every dog didn’t bark’ are logically equivalent and therefore have the same truth conditions. So truth conditions alone cannot account for this difference in availability of antecedents for pronouns. And it’s not straightforward to see how to account for this difference without some level of representation more abstract than syntax.
A simple example grammar. The part after the colon describes how the meaning of the mother constituent is built up as a function of the meanings of the daughters.

1. \( S \rightarrow NP \ VP \) : \( VP(NP) \)
2. \( NP \rightarrow \text{Name} \) : \( \text{Name} \)
3. \( VP \rightarrow V_{\text{intr}} \) : \( V_{\text{intr}} \)
4. \( VP \rightarrow V_{\text{trans}} \ NP \) : \( V_{\text{trans}}(NP) \)
5. \( V_{\text{intr}} \rightarrow \text{snores} \) : \( \lambda x.\text{snore}(x), \text{etc} \)
6. \( \text{Name} \rightarrow \text{John} \) : \( \text{john, etc} \)
7. \( V_{\text{trans}} \rightarrow \text{likes} \) : \( \lambda y.\lambda x.\text{like}(x,y) \)
8. \( S \rightarrow S \text{ and } S \) : \( S \wedge S \)
9. \( VP \rightarrow VP \text{ and } VP \) : \( \lambda x.\text{VP}_1(x) \wedge \text{VP}_2(x) \)
10. \( VP \rightarrow \text{doesn’t } VP \) : \( \lambda x.\neg(\text{VP}(x)) \)
Semantics is compositional, driven by syntax

Although there are some purely semantic ambiguities: e.g.

\[\forall x. \text{student}(x) \rightarrow \exists y. \text{lecture}(y) \land \text{go}(x,y)\]
\[\exists y. \text{lecture}(y) \land \forall x. \text{student}(x) \rightarrow \text{go}(x,y)\]

- we don't want to say that these sentences are syntactically ambiguous.
S: snore(john)
NP | VP: \( \lambda x.\text{snore}(x) \)
  | John
  | \textit{John}
  | snores
  | \( \lambda x.\text{snore}(x) \)

S: like(john, jack)
NP | VP: \( \lambda x.\text{like}(x, jack) \)
  | John
  | \textit{John}
  | likes
  | \( \lambda y.\lambda x.\text{like}(x, y) \)
  | Jack
  | \textit{Jack}
  | \textit{Jack}
S: snore(john) ∧ ¬(like(john,jill))

NP

VP: \( \lambda v. \text{snore}(v) \land \neg(\text{like}(v,jill)) \)

V_{\text{intrans}}

snores

\( \lambda x. \text{snore}(x) \)

John

john

and

VP: \( \lambda w. \neg(\text{like}(w,jill)) \)

doesn’t

VP: \( \lambda z. \text{like}(z,jill) \)

V_{\text{trans}}

like

\( \lambda y. \lambda z. \text{like}(z,y) \)

NP

Jill

jill
One benefit of having a level of logical form is that we know how to do inference with (first order) logical forms. So far, all our final logical forms have been first order, although there are some higher order steps along the way. (Proof procedures for FOPC, not for HOL).

However, there are many context dependent phenomena in language that seem to suggest a two stage process of interpretation, involving reasoning from context:
Sentence $\rightarrow$ Quasi-logical form $\rightarrow$ Resolved logical form

For example:

A cat arrived. QLF: arrive(a(cat))

Resolve quantifier scope, create discourse entity:
RLF: $\exists x.\text{cat}(x) \land x = \text{cat23} \land \text{arrive}(x)$

It caught every mouse. QLF: catch(it,every(mouse))

Resolve pronoun reference, scope quantifier:
RLF: $\forall y.\text{mouse}(y) \rightarrow \text{catch}\text{(cat23,y)}$
The notion of QLF allows us to deal with scope ambiguities:

\[
\text{every student went to a lecture} \quad \text{go}(\forall \text{student}, \exists \text{lecture}) \Rightarrow \\
\forall x. \text{student}(x) \rightarrow \exists y. \text{lecture}(y) \wedge \text{go}(x, y) \\
\exists y. \text{lecture}(y) \wedge \forall x. \text{student}(x) \rightarrow \text{go}(x, y)
\]

But this raises the question of how much inference we can do without fully resolving interpretations:

Some student in every class will win a prize.
   \rightarrow At least one student will win a prize
Two students ate five pizzas
   \rightarrow Some students ate pizza
We can now translate parsed sentences into logical forms. Note that although we use higher order logic, by the time we have a translation for a whole sentence - in our examples at least - it is in the first order subset of this logic. This is useful, because we can do automated inference (“theorem proving”) for first order logic, but not (generally) for higher order logic.

So we can solve logical problems like these:

<table>
<thead>
<tr>
<th>Syllogisms</th>
<th>Logical Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>All men are mortal</td>
<td>$\forall x. \text{man}(x) \rightarrow \text{mortal}(x)$</td>
</tr>
<tr>
<td>Socrates is a man</td>
<td>$\text{man}(\text{Socrates})$</td>
</tr>
<tr>
<td>\therefore Socrates is mortal</td>
<td>$\text{mortal}(\text{Socrates})$</td>
</tr>
</tbody>
</table>

Or build a question answering system:

<table>
<thead>
<tr>
<th>(Having read the previous ‘text’)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Is Socrates mortal?</td>
<td>Yes</td>
</tr>
<tr>
<td>Who is mortal?</td>
<td>Socrates</td>
</tr>
</tbody>
</table>
Although in principle the logical approach could get us a long way towards the goal of language ‘understanding’, in practice there are many challenges:

- there are many constructions that we do not know how to deal with properly: e.g. some of the earlier adjective examples (comparatives, superlatives, etc.)
- even when we do, the resulting analysis is often higher-order, so not amenable to theorem proving
- and analyses have to be done by hand, requiring linguistic and logical expertise - little role so far for machine learning
- some efforts towards this: Groeningen Meaning Bank: http://gmb.let.rug.nl/
Cann, R. 1993. Formal Semantics. CUP.
Bob Carpenter 1998 Type Logical Semantics, MIT Press. (first couple of chapters)

Section IV of Jurafsky and Martin (2nd edition) covers some of these topics from a computational point of view.