COMPUTATIONAL LINGUISTICS:
Lecture 8: Semantics and Inference 1

Stephen.Pulman@comlab.ox.ac.uk

What do you know when you understand a sentence?

- what the world would be like if it were true **(truth conditions)**

(not the same as knowing whether it is true)

- what else has to be true if the sentence is **(entailments)**

- what is contextually implied by the utterance of that sentence **(pragmatic inferences)**

Paul Portner, 2004 What is Meaning? Fundamentals of Formal Semantics. Oxford: Blackwell.
Blakemore, D. 1992. Understanding Utterances: An Introduction to Pragmatics. Blackwell.
Cann, R. 1993. Formal Semantics. CUP.
Cruse, D.A. 2000. Meaning in Language. An Introduction to Semantics and Pragmatics. OUP.
Jaszczolt, K. 2002. Semantics and Pragmatics: Meaning in Language and Discourse. Longman.
Levinson, S.C. 1983. Pragmatics. CUP.
Saeed, J. 2003 Semantics. 2nd edition. Oxford: Blackwell.

LEVELS OF ANALYSIS: SEMANTICS (TRUTH CONDITIONS + INFERENCE)

We can distinguish different classes of word according to the inferences that they allow:

Jones is a Welsh lawyer.	$\rightarrow$ Jones is Welsh
	$\rightarrow$ Jones is a lawyer
All lawyers are musicians.	$\rightarrow$ Jones is a Welsh musician.

Jones is a former lawyer.	$\not\rightarrow$ Jones is a lawyer

---

Jones is a skilful lawyer.	$\rightarrow$ Jones is a lawyer.
	$\rightarrow$ Jones is skilful???
All lawyers are musicians.	$\rightarrow$ Jones is a musician.
	$\not\rightarrow$ Jones is a skilful musician.

Minnie is a large mouse.	
All mice are animals.	$\not\rightarrow$ Minnie is a large animal.

Structurally based inference

All men are mortal $\rightarrow$ Some men are mortal

John is taller than Bill
$\rightarrow$ Bill is less tall than John
$\not\rightarrow$ John is tall.

Bill is tall, John is taller than Bill $\rightarrow$ John is tall.

No fish can live out of water $\rightarrow$ No large fish can live out of water.

John will leave unless Mary leaves first $\equiv$ If Mary does not leave first, John will leave.

Jones quickly polished the boots $\rightarrow$ Jones polished the boots.

Jones won a Nobel prize $\rightarrow$ Jones won a prize $\rightarrow$ Jones won something.

But NB 'Jones is looking for a unicorn' which does not entail there is a unicorn.

Lexically based inference

X murdered Y $\rightarrow$ X killed Y $\rightarrow$ Y died.

X melted the chocolate $\rightarrow$ the chocolate melted.

Fido is a dog $\rightarrow$ Fido is an animal.

Presupposition

John regrets that his dog died. (His dog died)

John doesn't regret that his dog died. (His dog died).

Notice that presupposition cannot be the same as entailment.

LEVELS OF ANALYSIS: CONTEXTUAL RESOLUTION

A: Jones is a lawyer.
B: No, he isn't - he's a policeman.

He isn't = Jones is not a lawyer.

Needs to be filled in from prior context - but interaction with non-linguistic knowledge:

a) John gave Mary two vintage bottles of wine, but one of them was undrinkable. They were very disappointed/expensive.

b) James gave each boy a trumpet. They made a terrible noise.

c) Every college employs a gardener. They pay them badly.

d) Jones finished his homework before Smith did.

e) The porters refused the students admission because they feared/advocated violence.

LEVELS OF ANALYSIS: PRAGMATICS

Pragmatics = contextual influence on interpretation. What is meant can be more than what is said.

> A: Would you like some coffee?
> B: It would keep me awake.

A's question really demands the answer 'yes' or 'no'. 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:
utterance U + assumption A $\rightarrow$ conclusion C.

Both A and C are conveyed by U.

First Order Predicate Calculus

First Order Logic serves as a model for some aspects of what we would like to do for the semantics of natural languages:

- Syntax defines an infinite number of well formed sentences.
- Compositional interpretation function defines truth conditions for these.
- Either using denotational semantics or proof theory we can work out the entailments of sentences.
- But non-logical constants (for us, words) are taken as primitives: this misses out some important aspects of lexical semantics.

FIRST ORDER PREDICATE CALCULUS

Individual constants: e.g. Fred, 21, a, b, c, ...
Individual* variables: e.g. x_1, x_2,...(we will usually write 'x', 'y', 'z' for readability)
Predicates: e.g. P, Q, R, snores, likes, between, ...
Function symbols: e.g. mother-of, half-of, f, g, h, ...
Quantifiers: $\exists$, $\forall$. ($\exists$= Existential quantifier: 'there is a';
$\forall$= Universal quantifier: 'for all'
Connectives: $\wedge$, $\vee$, $\neg$, $\rightarrow$...)

Syntax: define a 'well formed formula' (wff):

wff	$\Rightarrow$ predicate(arg$_1$,....,arg$_n$)
wff	$\Rightarrow$ quantifier variable . (wff)
wff	$\Rightarrow$ $\neg$wff; wff $\wedge$ wff; wff $\vee$ wff; wff $\rightarrow$wff
argument	$\Rightarrow$ constant; variable ; function(arg$_1$,....,arg$_n$)

* $\boxed{\text{This is the only type of variable, hence 'first order'}}$

Example wffs:

Roy is Welsh = welsh(roy)

Janet likes John = likes(john,janet)

Every number has a successor = $\forall$x.number(x) $\rightarrow\exists$y. successor(x,y)

Every boy likes some girl = $\forall$x.(boy(x) $\rightarrow$ ($\exists$y.girl(y) $\wedge$ likes(x,y)))
$\exists$y.(girl(y) $\wedge$ $\forall$x.(boy(x) $\rightarrow$likes(x,y)))

It is not the case that for everyone who likes their father, their father likes them =

$\neg\forall$x.(likes(x,father-of(x)) $\rightarrow$likes(father-of(x),x))

The precise interpretation of the connectives is given via **TRUTH TABLES**:

p	$\wedge$	q
T	T	T
F	F	T
T	F	F
F	F	F

p	$\vee$	q
T	T	T
F	T	T
T	T	F
F	F	F

p	$\rightarrow$	q
T	T	T
F	T	T
T	F	F
F	T	F

$\neg$	p
T	F
F	T

The value for the whole expression is shown below the relevant connective.

Quantifier scope:

Every man loves a woman:

For every man, there is a (possibly different) woman such that he loves her

$\forall$x.(man(x) $\rightarrow\exists$y.(woman(y) $\wedge$ loves(x,y)))

There is (just one) woman who is such that every man loves her:

$\exists$y.(woman(y) $\wedge$ $\forall$x.(man(x) $\rightarrow$loves(x,y)))

Free vs bound variables:

$\underline{\forall\text{x.}(\underline{\exists\text{y.(likes(x,y))}} \wedge \text{ likes(}\underline{\text{y}}\text{,x)} \wedge \text{ likes(}\underline{\text{z}}\text{,x))}}$

A wff with no free variables is ‘closed’, otherwise ‘open’.

SEMANTICS of FOPC

Assume a non-empty domain of objects, D. These can be anything we like: our 'calculus' is a purely mechanical formal system, with logical properties that are independent of what we are talking about.

An 'interpretation' consists of such a domain and an 'assignment' or 'interpretation' function, I. The interpretation function associates:

> each individual constant with a member of D
> each 1-place predicate with a subset of D
> each 2-place predicate with a relation in $D \times D$
> ... and so on for n-place relations. We could also describe these as functions from sets to truth values, sets of pairs to truth values, etc.
> each 1-place function symbol with a function $D \rightarrow D$
> each 2-place function symbol with a function $(D \times D) \rightarrow D$
> ... and so on for n-place functions

We just assume that such relations and functions exist. In fact, this is an oversimplification when our predicates are supposed to model English words.

SEMANTICS OF FOPC

Clearly I provides a simple denotation for predicates and constants in a well formed formula. But for functions, we need a recursive definition:

Functions:
$\mathrm{I}(\mathrm{f}(\alpha_1,\ldots,\alpha_n)) = \beta$, where $\langle \mathrm{I}(\alpha_1),\ldots \mathrm{I}(\alpha_n),\beta\rangle \in \mathrm{I}(\mathrm{f})$.

Now we can define the notion of truth for wff of FOPC:

A wff of the form $\mathrm{P}(\alpha_1,\ldots,\alpha_n)$ is true iff $\langle \mathrm{I}(\alpha_1),\ldots,\mathrm{I}(\alpha_n)\rangle \in \mathrm{I}(\mathrm{P})$.

Connectives: via truth tables, as above.

We need to assume some way of interpreting variables by associating them with (arbitrarily chosen) elements in D.

Quantifiers: (simplified)
Universal: a wff of the form $\forall$x.P is true iff P is true for every choice of value for x in D.

Existential: a wff of the form $\exists$x.P is true iff P is true for at least one choice of value for x in D.

Example

John likes Mary but Mary doesn't trust him.

like(John,Mary) $\wedge$ $\neg$trust(Mary,John)

I(likes) = $\{\langle$x,y$\rangle$ | x likes y $\}$
I(trusts) = $\{\langle$x,y$\rangle$ | x trusts y $\}$
likes(i,j) is true iff $\langle$i,j$\rangle \in$ I(likes)
trusts(i,j) is true iff $\langle$i,j$\rangle \in$ I(trusts)
I(John/Mary) = the actual person John/Mary
S1 $\wedge$ S2 is true iff S1 is true and S2 is true.
$\neg$S is true iff S is false.

So: like(John,Mary) $\wedge$ $\neg$trust(Mary,John) is true iff
like(John,Mary) is true, and
$\neg$trust(Mary,John) is true.

like(John,Mary) is true iff $\langle$I(John),I(Mary)$\rangle \in$ I(likes)
$\neg$trust(Mary,John) is true iff trust(Mary,John) is false
trust(Mary,John) is false iff $\langle$I(Mary),I(John)$\rangle \notin$ I(trusts)

A Semantics for a Fragment of English

1.	S	$\rightarrow$ NP VP	T iff I(NP) $\in$ I(VP)
2.	NP	$\rightarrow$ Name	I(NP)=I(Name)
3.	VP	$\rightarrow$ V_{intr}	I(VP)=I(V)
4.	VP	$\rightarrow$ V_{trans} NP	I(VP)=$\{X \mid \langle$X,I(NP)$\} \in$ I(V)$\rangle$
5.	V_{intr}	$\rightarrow$ snores	I(Vi)=$\{X \mid X$ snores$\}$ etc.
6.	Name	$\rightarrow$ John	I(John) etc.
7.	V_{tran}	$\rightarrow$ likes	I(Vt)=$\{\langle X,Y\rangle \mid X$ likes $Y\}$
8.	S	$\rightarrow$ S and/but S	T iff both daughter S are true
9.	VP	$\rightarrow$ VP and VP	I(VP0)= $\{X \mid X \in$ I(VP1) and $\in$ I(VP2)$\}$
10.	VP	$\rightarrow$ doesn't VP	I(VP0)=$\{X \mid X$ is not in I(VP1)$\}$

Domain = $\{$john,bill,mary,sue$\}$
I(likes) = $\{\langle$john,mary$\rangle$,$\langle$sue,mary$\rangle\}$; I(snores) = $\{$john,sue$\}$; I(John) = john, etc.