

Presuppositions (introductory comments)

Some examples

- (1) a. The person who broke the typewriter was Sam.
- b. It was Sam who broke the typewriter.
- c. John screwed up again.
- d. John likes Mary, too.
- e. John realizes that he screwed up.

1. Identifying Presuppositions

- (2) The person who broke the typewriter was Sam.

Sentence (2) seems to entail the following sentences:

- (3) a. Someone broke the typewriter.
- b. Only one person broke the typewriter.
- c. Sam broke the typewriter.

However, different feelings are associated with the different entailments. In particular (3)a and (3)b feel somewhat different from (3)c. But, how so?

There is a clear feeling (it is said) that in “an ideal” or “proper” conversation, (3)a and (3)b would already be “taken for granted” at the point in which (2) is uttered, and that (3)c would be taken to be the noteworthy piece of the conversation.

- (4) We say that sentence, S, presupposes p if p is taken for granted in every “ideal conversation” before S is uttered.

Major methodological problem: is it clear that we have reliable intuitions about how things work in an ideal conversation, given that so many of our conversations are far from ideal (i.e., require “presupposition accommodation”)?

- (5) I am asked by someone who I have just met, “Are you going to lunch?” I reply, “No, I’ve got to pick up my sister.” Here I seem to *presuppose* that I have a sister, even though I do not assume that the addressee knows this. (Stalnaker 1974, for very useful discussion, see von Stechow 2000)

How can we justify Stalnaker’s intuition about what he seems to presuppose.

Two fortunate facts:

- a. There are a few tests that allow us to identify presuppositions (with no reliance on our “theoretical” intuitions about the nature of ideal conversations).
- b. Presuppositions have characteristic projection properties.

2. Two Tests (from Kai's class notes)

2.1. *Hey, wait a minute!* (von Fintel, crediting Shanon for the inspiration)

- (6) The person who broke the typewriter was Sam.
- Hey wait a minute, I didn't know that someone broke the typewriter.
 - #Hey wait a minute, I didn't know that Sam broke anything.

- (7) At 5 PM, John screwed up again.
- Hey wait a minute, I didn't know that he screwed up before.
 - #Hey wait a minute, I didn't know that he messed up at 5 PM.

If p is a presupposition of S , *HWM*, *I didn't know that p* is a good response to an utterance of S .

If p is an entailment of S which is not a presupposition (or entailed by a presupposition), *HWM*, *I didn't know that p* is a bad response to an utterance of S .

Possible Explanation: Although presupposition accommodation is possible. It is something that we do out of our good nature when the presupposition is not controversial. But sometimes we might want to object and natural language has a construction specialized for this purpose.

Problem: As pointed out by Ofra, we can object to aspects of the assertion other than the presupposition with *HWM*:

- (8) All of the students were at the party
Hey wait a minute, I didn't (even) know that SOME of the students were at the party.

So, if there is a test for presuppositions here, it is rather subtle...

2.2. *What's more! Furthermore...* (Percus 1998 lecture notes)

- (9) A. Sam broke a typewriter.
B. #Yes, and furthermore, the person who broke a typewriter is Sam.
B. Yes, and furthermore, Only Sam broke a typewriter.

If p is a presupposition of S and what we can infer from S is equivalent to p and S' , *yes*, and *what's more S* (or *yes, furthermore S*) is a bad response to an utterance of S' .

In other words: *what's more S* is bad if only the presupposed part of S is new in the context.

Explanation: The presupposition of every assertion must in principle be satisfied in the context. (In other words, every conversation is not that different from our idealization.)

But there is this process of presupposition accommodation. Before we evaluate the sentence we might accommodate the presupposition, and then evaluate the sentence. What goes wrong with the bad *what's more* sentences is that after accommodation, the sentence provides no new information. And a *what's more* sentence must provide new information.

3. An Empirical Claim and a Theoretical Challenge

- (10) We say that a sentence S presupposes p , if
- p is taken for granted in every utterance of S (as diagnosed by HWM, WM...)
 - p has special projection properties (to be specified).

Empirical Claim: If p is an entailment of S , p satisfied (a) iff p satisfied (b)
(Our two tests for presuppositions coincide.)

Challenge: To understand why the empirical claim might be right.

In other words, to understand:

- What type of grammatical representation for S (together with principles of language use) leads to the pragmatic fact that p is taken for granted.
- To understand how the grammatical representation of S (together with principles of language use) leads to the particular projection properties we will identify.

4. Stalnaker's Strategy

- (11) We say that a sentence S semantically presupposes p , if the denotation of S is a partial function, which is undefined in a world w , unless p is true in w .

Every conversational context could be said to be based on a set of shared presuppositions (Common Ground, CG), which could be characterized (with many idealizations) as a set of possible worlds (Context Set, CS; henceforth, simply C).

The reason we utter a sentence S is to add its content to CG. This fails unless the presuppositions of S are true in every world in C .

$$(12) \quad \text{CCP}(S_p) = \lambda C: C \subseteq p. C \cap \{w: S'(w)=1\}$$

Stalnaker asks us to imagine a procedure that for every world in C , w , needs to decide whether w stays in the updated context or not. If S' is false in w , the answer is *no*, if S' is true in w , the answer is *yes*. If S' is undefined in w , the procedure fails.

Does this help us in any way in the projection problem?

4.1. Negation, a semantic approach (from Heim and Kratzer, similar to what is assumed in Gazdar)

$$(13) \quad \text{neg}' = \lambda p_t. p = 0$$

Which is shorthand for:

$$(14) \quad \text{neg}' = \lambda p: p \in \{0,1\}. p = 0$$

which is the same as:

$$(15) \quad \text{neg}' = \lambda t. \begin{array}{l} 1 \text{ if } t = 0 \\ 0 \text{ if } t = 1 \\ \text{Undefined, otherwise} \end{array}$$

- (16) John doesn't admire the king of France.
 λw : there is a unique KoF in w . John doesn't admires the KoF

Hence negation is a "hole" (Karttunen's terminology)

- (17) A function $f_{\langle \text{st}, \text{st} \rangle}$ is:
- a hole if for all X that presupposes p (for all X_p), $f(X)$ presupposes p .
 - a filter if for all X_p , $f(X)$ presupposes q , and $q \neq p$.
 - a plug if for all X_p , $f(X)$ presupposes nothing.

Generalizes in the standard way to n place relations that take a proposition in the n^{th} position.

4.2. Conjunction, a pragmatic approach (Stalnaker)

$$(18) \quad \text{and}' = \lambda t_1. \lambda t_2. t_1 = t_2 = 1$$

Conjunction should be a hole too, but it's a filter:

- (19) Moldavia is a monarchy and the King of Moldavia is powerful.
 λw : there is a unique KoM. Moldavia is a monarchy and the KoM is powerful.

Stalnaker, the two sentences are parsed one at a time (a matter of language use):

"Once a proposition has been asserted in a conversation, then (unless or until it is challenged) the speaker can reasonably take it for granted for the rest of the conversation. In particular, when a speaker says something of the form *A and B*, he may take it for granted that *A* (or at least that his audience recognizes that he accepts that *A*) after he has said it. The proposition that *A* will be added to the background of common assumptions before the speaker asserts that *B*. Now suppose that *B* expresses a proposition that would, for some reason, be inappropriate to assert except in a context where *A*, or something entailed by *A*, is presupposed. Even if *A* is not presupposed initially, one may still assert *A* and *B* since by the time one gets to saying that *B*, the context has shifted, and it is by then

presupposed that A.”

But. How general is this?

1. Problems with other connectives where it is less obvious how a statement about the order of assertion might be useful.

(20) Either this house has no bathroom or the bathroom is well hidden.

(21) If Moldavia is a Monarchy, the king is not particularly noticeable.

3. Embedding

(22) a. I have never met an academic who was an idiot and who knew that he was an idiot. (Schlenker)

b. I never heard of a monarchy in which the king was bald.
Cf. *I never heard of a country in which the king was bald*

5. The Heim/Karttunen Strategy (Heim 1983)

Adopt Stalnaker’s basic intuition but introduce it into the grammar.

More specifically, replace denotations we are accustomed to with CCPs

$C + S$ is Heim’s notation for $CCP(S)(C)$

For atomic sentences, we assume lexical specification of presuppositions, e.g., by partial functions:

(23) $C + S\{p\}$ is defined iff $C \subseteq p$, when defined
 $C + S\{p\} = C \cap \{w: S'(w)=1\}$

For complex sentences:

(24) a. $C + \text{not } A = C - [C + A]$
b. $C + [A \ \& \ B] = [C + A] + B$
c. $C + [\text{If } A, \text{ then } B] = C + [\text{not } [A \ \& \ \text{not } B]]$
 $= C - [C + [A \ \& \ \text{not } B]] = C - [[C + A] + \text{not } B]$
 $= C - ([C + A] - [[C + A] + B])$

There’s more to which we will probably return. Still need to discuss:

- a. the CCP of quantifiers,
- b. accommodation strategies and the way they might relate to CCPs (local vs. global accommodation).

6. Still not understood (Soames 1984, Rooth cited in Heim 1990):

E.g., why not the following?

- (25) a. $C + \text{not } A = C \text{ — } [W + A]$
- b1. $C + [A \ \& \ B] = [C + B] + A$
- b2. $C + [A \ \& \ B] = [C + B] \cap [C + A]$
- c. $C + [\text{If } A, \text{ then } B] = [C \text{ — } [C + A]] \cup [C + B]$, and more...

So, we have a way of stating the projection properties of various connectives within Heim’s framework. But we can’t predict the projection properties from the “classical” meanings of the connectives.

Is it reasonable to hope that we would be able to predict projection properties from the basic meanings of the connectives?

7. Peters’ (1979) Strategy

Various people, starting, I think, with van Fraassen, have tried to deal with presupposition projection using tools from tri-valiant logical systems. Karttunen, who is responsible for all of the generalizations we are trying to capture, argued that they can’t be captured by the tri-valiant systems. Peters’ aim was to show that this is false. Basic machinery: add a third member to D_t , and revise the truth functional lexical entries appropriately.

(26) $D_t = \{1, 0, \#\}$

(27) $\text{and}(p)(q) = 1$, if $p = q = 1$
 $= 0$, if $p = 0$ or $[(p = 1) \ \& \ (q = 0)]$
 $= \#$, if $p = \#$ or $[(p = 1) \ \& \ (q = \#)]$

$p \wedge q$	q	1	0	#
	p			
	1	1	0	#
	0	0	0	0
	#	#	#	#

This, in conjunction with Stalnaker’s assertability condition, yields more or less correct results:

(28) **Stalnaker’s Assertability Condition:**

A sentence S is assertable in a context C only if $\forall w \in C [S(w) \neq \#]$

But aren’t the truth tables just as stipulative as Heim’s lexical entries?

6. Trivalent system in the post Schlenker era:

Strong Kleene:

$$(29) \quad \text{and}(p)(q) = 1, \text{ if } p = q = 1 \\ = 0, \text{ if } p = 0 \text{ or } [(q = 0)] \\ = \#, \text{ otherwise}$$

$p \wedge q$				
	q	1	0	#
p				
1		1	0	#
0		0	0	0
#		#	0	#

Intuition: imagine that the system is 2 valued, and that # stands either for 1 or 0, but that you don't know which. If you can figure out the value for the whole sentence, ignoring #, you have it. Otherwise, the value is #.

A rule (for a propositional logic): Suppose you encounter a node, X, of type t with semantic value #. Look at the first node of type t above X, Y. If you can figure out the value for Y, ignoring X, you have it. That is to say, if the truth value for Y is the same in a two valued system when you replace the value of X with 0 or 1, the value of node Y is whatever that truth value is. Otherwise, the value of Y is #.

Peter's Intuition (hinted at in Beaver and Krahmer): Imagine that the system is 2 valued, and that # stands either for 1 or 0, but that you don't know which. If you can figure out the value for the whole sentence, ignoring # *and everything that follows it*, you have it. Otherwise, the value is #.

A rule (for a propositional logic): Suppose you encounter a node, X, of type t with semantic value #. Look at the first node of type t above X, Y. If you can figure out the value for Y, based on material pronounced before X, you have it. That is to say, if the truth value for Y is the same in a two valued system when you replace the value of X *and every node* (of type t) in Y *pronounced after X* with 0 or 1, the value of node is whatever that truth value is. Otherwise, the value of Y is #.

Homework: construct lexical entries for *or* and *if then*, guided by the Strong Kleene and the Peters intuition as it was stated above.

MIT OpenCourseWare
<http://ocw.mit.edu>

24.954 Pragmatics in Linguistic Theory
Spring 2010

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.