Week 2 - Satisfaction Theory I

Anti-Presupposition
Compare:

(1)   a.   The Chandler House cat is asleep in the library.
              Presupposition: There exists a Chandler House cat.
    b.   A Chandler House cat is asleep in the library.

The idea of an **anti-presupposition** is that if a a speaker does not select the alternative with a stronger presupposition, there is an implicature that they are denying that presupposition. In other words, if one was to utter (1-b), the speaker is intentionally denying the existence presupposition in (1-a).

This Week
This week we will begin discussing the **Satisfaction Theory for presuppositions**. Before jumping into it, we will need to understand the foundational concepts behind the theory. Specifically, what is file change semantics? What motivates this framework and what are its components? Then we will discuss how this overall framework accounts for presupposition projection under certain operators. We will discuss more complex operators and issues with the theory in the following week.

Static Semantics vs. Dynamic Semantics

> *'You know the meaning of a sentence if you know the change it brings about in the information state of anyone who accepts the news conveyed by it.'*
> (Veltman 1996)

The fundamental difference between **static semantics** and **dynamic semantics** is how the meaning of sentences are denoted; In 'traditional' static semantics, sentences are denoted as truth conditions e.g., 'Montagunian grammar' (Montague (1973); Partee (1975)) and early intensional logics (Kripke 1963 a.o.). On the other hand, dynamic semantic treats the denotation of sentences to be functions from contexts to contexts (**Context Change Potentials (CCP)**). The formulation of dynamic semantics came about as a way to account for anaphoric and context-dependent meaning, especially in relation to tracking discourse referents.[1]

To expand on this, imagine multi-sentence discourse with indefinites:

(2)   I bought a pet lizard. Then, I killed it.

---

[1]File Change Semantics is a kind of dynamic theory. Over time, other dynamic theories have also emerged e.g., Discourse Representation Theory (Kamp & Reyle 1993) and Dynamic Propositional Logic (Groenendijk & Martin 1991).

Using static semantics, and assuming that indefinites are existential quantifiers we might end up with something like this (Russell 1905):

(3) $\exists x[\text{Lizard}(x) \wedge \text{bought(i,x)}] \wedge \text{Killed}(v_3)$

Here, the v_3 variable is unbound and outside the scope of the existential quantifier.

(4) $\exists x[\text{Lizard}(x) \wedge \text{bought(i,x)} \wedge \text{Killed}(x)]$

If you forced the quantifier to take scope over multi-sentence discourses you run into an array of other issues, for example a sentence pair such as the following would suggest that there exists an x such that x is a man and x fell over the edge and x didn't fall over the edge and x jumped, which is clearly contradictory.

(5)   a.   A man fell over the edge.   (Example from C&C)
        b.   He didn't fall; he jumped.

Dynamic semantics is advantageous in the sense that it allows us to update contexts sentence-by-sentence in a discourse. We can introduce discourse referents in one context update and access the updated content to find the antecedent of an indefinite. Although all dynamic theories are spiritually the same thing, there are different theories based on notation and specific features. We will be focusing on **File Change Semantics**, which was first proposed in Heim (1982).

What is 'context'?
Dynamic semantics extensive use of the notion of contexts is an extension of the Stalnakerian pragmatics notion of context, which models conversational contexts as a set of possible worlds, a formalisation of the common ground (Stalnaker 1973; Stalnaker 1974; Stalnaker 1974). These possible worlds are compatible with the mutual beliefs[2] of the discourse participants. Assertions 'update' the context set by restricting the set of worlds.

(6) $c[\alpha] = c \cap \{w | [\![\phi]\!](w) = 1\}$

(7) $W = \{w_\theta, w_s, w_r, w_{sr}\}$ Example from notes by Patrick D. Elliot

it's raining $\Rightarrow \lambda w$. $\textbf{raining}_w$

$\{w | \textbf{raining}_w\} = \{w_r, w_{rs}\}$

$c[\text{it's raining}] = W \cap \{w_r, w_{rs}\} = \{w_r, w_{rs}\}$

This idea of context works with static truth-conditional semantics, as we can define truth conditions in terms of context set restriction:

(8) $[\![\text{It is raining is }]\!]^w = \begin{cases} 1 \text{ in } w & \text{if } \{w\} \, [\text{it is raining}] = \{w\} \\ 0 \text{ in } w & \text{if } \{w\} \, [\text{it is raining}] = \theta \end{cases}$

---

[2]Stalnaker calls these mutual beliefs 'pragmatic presuppositions'

In File Change Semantics, the meaning of a sentence is a function from files to files, otherwise known as Context Change Potentials (CCPs):

(9)   ⟦Saki is furious⟧ $= \lambda c. \cap \{w|$ Saki is furious in $w\}$

In this example above, c is an input context, the function intersects the input context with the set of worlds in which Saki is furious, with the resulting output being a new context.

Heim's original idea was that discourse references can be stored as 'file cards' (Heim 1982). An indefinite introduces a new file card and subsequent anaphoric reference updates the file card.

Satisfaction Theory
Satisfaction theory is a component of File Change Semantics that deals with the projection problem for presuppositions (Heim 1983; Beaver 2001 a.o.). A sentence p_x (where x is ppresupposition) requires that x be true throughout the context set c, in order for an update of c to be defined:

(10)   ⟦Mika fed the Chandler House cat⟧ $= \lambda c : c \subseteq \{w'|$there is a Chandler House cat in $w'\}$
        $.c \cap \{w|$Mika fed the Chandler House cat in $w\}$

In other words, c satisfies the presupposition of '...The Chandler House cat' iff for all $w \in c$, there is a Chandler House cat in c.

We know that the update of (10) is undefined if there is no Chandler House cat – so we can actually model this context change as a partial function that only allows certain input contexts within the domain of felicitous updates.

(11)   a. $c \in \text{dom}([$Mika fed the Chandler House cat$])$ iff for all $w \in c$, there is a Chandler House cat in w.

        b. $c[$Mika fed the Chandler House cat$] = \{w \in c|$Mike fed the Chandler House cat in $w\}$

Negation
Now that we are dealing with non-atomic sentences, it is worth mentioning that context updates in sentences embedded under operators can involve different **sub-contexts**.

(12)   **Global context**: The context before the entire sentence is updated.
        Local context: The context before the part inside the operator is updated.

First, we turn our attention to negation, which is derived in File Change Semantics as follows:

(13) $c[\neg\phi] = c - (c[\phi])$

You essentially update c with ϕ and then subtract the output of worlds from c. The fact that that you first have to compute the context without negation demonstrates a sub-context. The local context is the input for the embedded clause $c[\phi]$ and the global context is the input for $c[\neg\phi]$.

Since computing not $\neg\phi$ requires computing ϕ first, and ϕ's presupposition must hold in c (the input to negation), the negated sentence ends up requiring the same presupposition. This is all good, as it demonstrates why presuppositions project through negation:

(14)   (it is not the case that) Andrew stopped responding to my emails.
        Still presupposes: Andrew used to respond to my emails.

Conjunction

Now, lets turn to conjunction, which is formalised as the following:

(15) $c[\phi \wedge \psi] = c[\phi][\psi]$

c is the global context for the entire conjunction, the input being the context set prior to any computation. From this, we can compute $c[\phi]$ which then becomes in the input for the second local context $[\psi]$:

(16)   a. $c \in \text{dom}([\phi \text{ and } \psi] \text{ iff } c \in \text{dom}([\phi]) \text{ and } c[\phi] \in \text{dom}([\psi])$.
        b.   Whenever defined, $c[\phi \text{ and } \psi] = c[\phi][\psi]$

Given this rule, lets suppose a conjunction $\phi \wedge \psi$ whereby ϕ entails the presupposition of ψ:

(17)   Richard drinks and Yasu drinks too.

In the example above, the first conjunct (the fact that Richard drinks) entails the presupposition of the second conjunct (that someone other than Yasu drinks). The idea is that after updating $[\phi]$ with c, the local context now already satisfies the presupposition of $[\psi]$. Therefore, ψ's presupposition does not need to project to the global context – it is filtered locally.

(18)   ⟦Richard drinks⟧ $\wedge$ ⟦Yasu drinks too.⟧
        $= \lambda c.\ [\lambda c' : c' \subseteq \{w'|\text{someone other than Yasu drinks in } w'\}\ .\ c' \cap \{w'|\text{Yasu drinks in } w'\}]$
        $(c \cap \{w|\text{Richard drinks in } w\})$

Conditionals

For conditionals we have the following analysis:

(19) $c[\text{if } \phi, \text{ then } \psi] = c - (c[\phi] - c[\phi][\psi])$

Here we also have **sequential updates** that explain project patterns in conditionals. First, we compute c$[\phi]$, which is the global context with the antecedent. Then we update the local context of the consequent with the antecedent output $(c[\phi][\psi])$.

Since the presuppositions of ϕ are checked in the global context c, a presupposition in the **antecedent** always projects. Whether or not it projects in the consequent is based on if the presupposition is also satisfied in the antecedent for the local context. This neatly explains previously established presupposition projection patterns.

(20)   If Gump distracts Joseph **again**, I will complain to HR.
→ The presupposition that Gump has distracted Joseph before always projects in the antecedent.

(21)   Cathy has a dog, and **her dog** is very cute.
→ The presupposition that Cathy owns a dog is entailed by the antecedent and thus filtered.

Disjunction
One way to look at disjunction is symmetric disjunction:

(22) $c[\phi \vee \psi] = c[\phi] \cup c[\psi]$

In this approach, you update c with ϕ and ψ separately, and take the union of the two resulting contexts. As both contexts take the global context as an input, they check for presuppositions globally, thus projecting in either disjunct. This supports projection data:

(23)   Either Gump is annoying Lily again or is at the gym.

(24) Either Gump is at the gym or annoying Lily again.

Global Accommodation
We will focus on global accommodation for this week. Consider an expression such as the one I talked about last week:

(25)   Gump is attending his son's birthday party.

The hearer does not necessarily need to know that Gump has a son, they could just adapt the common ground by incorporating this belief.

Satisfaction theory deals with this by introducing a mechanism which lets you first update the context set with the information that Gump has a son and then updating the resulting context with the at-issue meaning of (25).

(26)   [Gump is attending his son's birthday party.]
c - (c[Gump has a son][Gump is attending his birthday party])

Acknowledgements
Handout inspired by all the readings (References to be added), Handouts by Patrik D. Elliot, Martin Hackl, Yasutada Sudo, Chris Potts and the C&C textbook.

List of References

Beaver, David (2001). *Presuppsition and Assertion in Dynamic Semantics*. Stanford: CSLI.

Groenendijk, Jeroen & Stokhof Martin (1991). "Dynamic Predicate Logic". *Linguistics and Philosophy* 14 (1): 39–100.

Heim, Irene (1982). "The Semantics of Definite and Indefinite Noun Phrases". PhD thesis. University of Massachusetts, Amherst.

Heim, Irene (1983). "On the projection problem for presuppositions". *In WCFL 2*: 114–125.

Kamp, Hans & Uwe Reyle (1993). *From Discourse to Logic*. Springer Nature.

Kripke, Saul (1963). "Semantical Considerations on Modal Logic". *Acta Philosophica Fennica* 16: 83–94.

Montague, Richard (1973). "The proper treatment of quantification in ordinary English". In: *Approaches to Natural Language*. Ed. by Patrick Suppe, Julius Moravcsik & Jakko Hinikka. Dordrecht, 221–242.

Partee, Barbara (1975). "Montague Grammar and Transformational Grammar". *Linguistic Inquiry* 6 (2): 203–300.

Russell, Betrand (1905). "On Denoting". *Mind* 14 (56): 479–493.

Stalnaker, Robert (1973). "Presuppositions". *Journal of Philosophical Logic* 2 (4): 447–457.

—— (1974). "Pragmatic presuppositions". In: *Semantics and Philosophy*. Ed. by Milton K. Munitz & Peter Unger, 141–117.

Veltman, Frank (1996). "Defaults in update semantics". *Journal of Philosophical Logic* 25 (3): 221–261.