Deriving Truth Conditions

Lecture 06

January 23, 2024

Announcements:

This lecture is supplemented by the following readings:

- Heim & Kratzer: Ch.1 (13–23)

Your second homework assignment is available and is due on January 25th.

1 Introduction

During our last meeting we took steps toward a theory of the set of rules that deliver on our goal to compute the truth-conditional meaning of utterances. Given our goals, what we ultimately wanted to end up with is a system along the following lines:

(1) $[\![\ [_{\text{NP}}\ \text{Sam}\ ]\ ]\!] + [\![\ [_{\text{VP}}\ \text{dances}\ ]\ ]\!] =$
$[\![\ [_{\text{S}}\ \text{Sam dances}\ ]\ ]\!] =$
truth conditions of $[_{\text{S}}$ Sam dances]

We proposed that Proper Nouns denote entities and that intransitive verbs denote functions that map entities into the set of truth values:

(2) **Proper Nouns as entities**
$[\![\ \text{NAME}\ ]\!] =$ the thing referred to with NAME

(3) **Predicates as functions**
$[\![\ \text{VERB}_{intrans}\ ]\!] = f : \{\, x : x \text{ is an entity} \,\} \to \{\, T, F \,\}$
for every $y \in \{\, x : x \text{ is an entity} \,\}$, $f(y) = T$ *iff* y VERBs

With these lexical entries, we were able to define the following rule of semantic composition to combine predicates with their arguments.

(4) **Functional Application**
If X is a node that has two daughters, Y and Z, and if $[\![\ \text{Y}\ ]\!]$ is a function whose domain contains $[\![\ \text{Z}\ ]\!]$, then $[\![\ \text{X}\ ]\!] = [\![\ \text{Y}\ ]\!]([\![\ \text{Z}\ ]\!])$.

This small set of rules and representations allows us to compositionally derive the extension—i.e., the **truth value**—for any sentence of the form in (5), which composes and intransitive verb with a proper noun.

(5) Sam dances.

While this represents major progress, it doesn't quite represent our desire for a system that compositionally derive the **truth conditions** of an utterance. In other words, we want a formal system that derives a correct statement of the following form:

(6) **Truth-conditional Statement of *Sam dances***
Sam dances is *T iff* Sam dances

As we will see immediately below, our system for deriving extensions can be seen as part of a larger system that does exactly this.

Basically, if we, as addressees, accept that what we're hearing has the value *True*, our compositional semantics can use its knowledge of the extensions of the component parts to compute how the world must be in order for the sentence to be *True*.

We will also fill in some technical details regarding the relationship between our semantic system and properly articulated syntactic structures. This will require that we introduce two additional semantic rules of composition:

(7) **Non-Branching Nodes (NN)**
If X is a non-branching node that has Y as its daughter, then $[\![\text{X}]\!] = [\![\text{Y}]\!]$

(8) **Terminal Nodes Rule (TN)**
If X is a terminal node, then $[\![\text{X}]\!]$ is specified in the lexicon.

2 From Extensions to Truth Conditions

2.1 Deriving Extensions: The Remix

We saw in our previous meeting how a compositional extensional semantics can, given the facts of the world, compute the truth value of a sentence of the shape in (9).

(9) $$\left[\!\!\left[\begin{array}{c} \text{S} \\ \diagup \quad \diagdown \\ \textbf{Kuniko} \quad \textbf{sings} \end{array} \right]\!\!\right] = T$$

Given our understanding of the meaning of proper names and intransitive verbs, the components parts of this sentence will have the lexical entries listed in (10):

(10) **Lexical entries**

a. $[\![\text{ Kuniko }]\!] = \text{Kuniko}$

b. $[\![\text{ sings }]\!] = f_{sings} : \{\, x : x \text{ is an entity} \,\} \rightarrow \{\, T, F \,\}$
for every $y \in \{\, x : x \text{ is an entity} \,\}$, $f(y) = T$ *iff* y sings

The rule that we have for composing the component parts of a sentence is Functional Application, presented in (11):

(11) **Functional Application (FA)**
If X is a node that has two daughters, Y and Z, and if $[\![\text{ Y }]\!]$ is a function whose domain contains $[\![\text{ Z }]\!]$, then $[\![\text{ X }]\!] = [\![\text{ Y }]\!]([\![\text{ Z }]\!])$.

It is possible to schematically represent the rule of Functional Application as in (12):

(12)
$$\left[\!\!\left[\begin{array}{c} \text{X} \\ \wedge \\ \text{Z} \quad \text{Y} \end{array}\right]\!\!\right] = [\![\text{ X }]\!] = [\![\text{ Y }]\!]([\![\text{ Z }]\!]) = f_Y(z)$$

A demonstration of how the system computes the extension of (9) is presented below:

(13) **Semantic Derivation**

i.	$[\![\text{ S }]\!]$	=	(by Functional Application)
ii.	$[\![\text{ sings }]\!]([\![\text{ Kuniko }]\!])$	=	(by Lexicon)
iii.	$[\![\text{ sings }]\!](\text{Kuniko})$	=	(by Lexicon)
iv.	$f_{sings}(\text{Kuniko})$	=	(by definition of f_{sings} and facts of the world)
v.	T		

2.2 Deriving Truth Conditions

With this much in place, we have a proof-of-concept of a basic system that can derive the extension—i.e., the **truth value**—of a sentence from the extension of its component parts.

As we noted, however, this isn't the kind of formal semantic system that we desired. When we understand the meaning of sentence, we know it's **truth conditions**, even if we don't know it's actual truth value.

Thus, we would like to have a system that computes truth-conditional statements of the kind in (14):

(14) **Truth-conditional Statement of *Kuniko sings***
Kuniko sings is T *iff* Kuniko sings

As it happens, the system that we have developed allows us to do exactly this.

Our system for computing the extension of a sentence can be employed in a compositional system that presupposes the truth of utterances and computes the conditions of the world that must hold in order for the sentence to be true.

We can demonstrate the calculation of the truth conditions of an utterance using the same proof-schema that we employed above. We will refer to this process as a **semantic proof** of the truth conditions of a sentence.

To do this, we start from the truth-conditional statement that wish to derive:

(15) **Truth-conditional Statement of *Kuniko sings***
Kuniko sings is T *iff* Kuniko sings

We can take note of the lexical entries of the words that compose the utterance:

(16) **Lexical entries**

a. $[\![\text{Kuniko}]\!] = \text{Kuniko}$

b. $[\![\text{sings}]\!] = f_{sings} : \{ x : x \text{ is an entity} \} \rightarrow \{ T, F \}$
for every $y \in \{ x : x \text{ is an entity} \}$, $f(y) = T$ *iff* y sings

We can also remind ourselves of the rule we have for semantic composition:

(17) **Functional Application (FA)**
If X is a node that has two daughters, Y and Z, and if $[\![\text{Y}]\!]$ is a function whose domain contains $[\![\text{Z}]\!]$, then $[\![\text{X}]\!] = [\![\text{Y}]\!]([\![\text{Z}]\!])$.

We then attempt to deduce our truth-conditional statement of the meaning of the expression from our semantic rule and lexical entries.

(18) **Calculation of the Truth Conditions of *Kuniko sings***

i.	"Kuniko sings"	is T *iff*	(by syntactic assumptions)
ii.	" [S **Kuniko** **sings**] "	is T *iff*	(by notation)
iii.	$[\![\text{S}]\!]$	$= T$ *iff*	(by Functional Application)
iv.	$[\![\text{sings}]\!]([\![\text{Kuniko}]\!])$	$= T$ *iff*	(by Lexicon)
v.	$[\![\text{sings}]\!](\text{Kuniko})$	$= T$ *iff*	(by Lexicon)
vi.	$f_{sings}(\text{Kuniko})$	$= T$ *iff*	(by definition of f_{sings})
vii.	Kuniko sings		

It is worth pointing out that this proof exploits an important logical truth:

(19) **Transitivity of *iff***
A iff B and *B iff C* entails *A iff C*.

When we put everything in above together, we conclude (18i) *iff* (18vii). In truth-conditional terms:

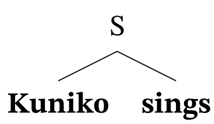
(20) **Truth-conditional Statement of *Kuniko sings***
Kuniko sings is *T iff* Kuniko sings

Thus, exactly as desired, we have a computational system that allows us to compute the truth conditions of sentences from the meanings of the component parts and their mode of composition.

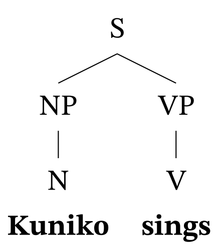
3 Syntax-Semantics Interface

3.1 Syntactic Structures

Despite the progress we have made modeling the semantic component of human language competence, we have made this progress while assuming extremely simplified syntactic representations, like in (21).

(21)


A significant amount of syntactic theorizing suggests that structures like (22) are a much more realistic representation for the sentences we have been investigating.

(22)


To the extent that this structure represent how the component parts are composed, we should aim to compute the truth conditions of the utterance from these representations.

Unfortunately, we don't currently have the ability to compute the meaning of such structures. Consider the semantic proof on the following page.

(23) **Attempted Calculation of the Truth Conditions of *Kuniko sings***

i.	"Kuniko sings"	is *T iff*	(by syntactic assumptions)
ii.	" [S [NP [N **Kuniko**]] [VP [V **sings**]]] "	is *T iff*	(by notation)
iii.	⟦ S ⟧	= *T iff*	(by Functional Application)
iv.	⟦ **??** ⟧(⟦ **??** ⟧)		

We would like to compute ⟦ S ⟧ using the rule of Functional Application, like we did above. The problem is that we can't; we don't know the value of either ⟦ NP ⟧ of ⟦ VP ⟧.

Because these are phrases and not lexical items, we can't look them up in the lexicon. Instead, we have to compute their meanings. However, they are not nodes that have the form specified by Functional Application so we also can't compute them.

(24) **Functional Application (FA)**

If X is a node that has two daughters, Y and Z, and if ⟦ Y ⟧ is a function whose domain contains ⟦ Z ⟧, then ⟦ X ⟧ = ⟦ Y ⟧(⟦ Z ⟧).

(25) **Schematic representation of Functional Application**

$$\left[\!\!\left[\begin{array}{c} \mathrm{X} \\ \diagup\!\diagdown \\ \mathrm{Z} \quad \mathrm{Y} \end{array} \right]\!\!\right] = [\![\mathrm{X}]\!] = [\![\mathrm{Y}]\!]([\![\mathrm{Z}]\!]) = f_Y(z)$$

3.2 More Rules of Composition

In order to compute the meaning of structures that more accurately represent syntactic reality, we will need to introduce a rule for computing NP and VP nodes.

The following **Non-Branching Nodes Rule** can serve this purpose.

(26) **Non-Branching Nodes (NN) Rule**

If X is a non-branching node that has Y as its daughter, then ⟦ X ⟧ = ⟦ Y ⟧

(27) **Schematic representation of Non-Branching Nodes**

$$\left[\!\!\left[\begin{array}{c} \mathrm{X} \\ | \\ \mathrm{Y} \end{array} \right]\!\!\right] = [\![\mathrm{Y}]\!]$$

We will also need a rule to compute the meaning of terminal N and V nodes.

The **Terminal Nodes Rule** shown below serves this purpose.

(28) **Terminal Nodes (TN) Rule**
If X is a terminal node, then $\llbracket X \rrbracket$ is specified in the lexicon.

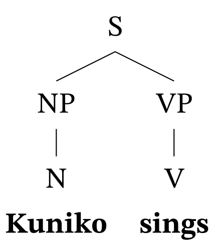
(29) **Schematic representation of Non-Branching Nodes**
$\llbracket X \rrbracket$ = Lexical Entry

These rules will allow us to perform **subproofs** within the larger **semantic proof** of the truth conditions of a sentence. Let us see how.

3.3 Compositionality

We can start by making clear our syntactic assumptions regarding the structure of the sentence:

(30)



The lexical entries for the components parts of the sentence that are stored in the lexicon include:

(31) **Lexical entries**

a. $\llbracket \text{Kuniko} \rrbracket = \text{Kuniko}$

b. $\llbracket \text{sings} \rrbracket = f_{sings} : \{\, x : x \text{ is an entity} \,\} \rightarrow \{\, T, F \,\}$
for every $y \in \{\, x : x \text{ is an entity} \,\}$, $f(y) = T$ *iff* y sings

We now have the set of semantic rules listed below:

(32) **Functional Application (FA)**
If X is a node that has two daughters, Y and Z, and if $\llbracket Y \rrbracket$ is a function whose domain contains $\llbracket Z \rrbracket$, then $\llbracket X \rrbracket = \llbracket Y \rrbracket(\llbracket Z \rrbracket)$.

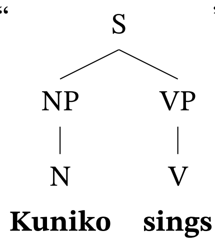
(33) **Non-Branching Nodes (NN) Rule**
If X is a non-branching node that has Y as its daughter, then $\llbracket X \rrbracket = \llbracket Y \rrbracket$

(34) **Terminal Nodes (TN) Rule**
If X is a terminal node, then $\llbracket X \rrbracket$ is specified in the lexicon.

We can now see how the proof of the truth conditions of can proceed with subproofs of the major constituents of the sentence.

(35) **Truth-conditional Statement of *Kuniko sings***
Kuniko sings is T *iff* Kuniko sings

(36) **Calculation of the Truth Conditions of *Kuniko sings***

i.	"Kuniko sings"	is T *iff*	(by syntactic assumptions)
ii.	"  "	is T *iff*	(by notation)
iii.	$[\![\text{S}]\!]$	$= T$	
iv.	*Calculation of* $[\![\text{NP}]\!]$		
	a. $[\![\text{NP}]\!]$	=	(by NN)
	b. $[\![\text{Kuniko}]\!]$	=	(by TN)
	c. Kuniko		
v.	*Calculation of* $[\![\text{VP}]\!]$		
	a. $[\![\text{VP}]\!]$	=	(by NN)
	b. $[\![\text{sings}]\!]$	=	(by TN)
	c. f_{sings}		
vi.	$[\![\text{S}]\!]$	$= T$ *iff*	(by FA, iv., v.)
vii.	$[\![\text{VP}]\!]([\![\text{NP}]\!])$	$= T$ *iff*	(by iv.)
viii.	$[\![\text{VP}]\!](\text{Kuniko})$	$= T$ *iff*	(by v.)
ix.	$f_{sings}(\text{Kuniko})$	$= T$ *iff*	(by definition of f_{sings})
x.	Kuniko sings		

It is, again, worth taking the time to point out some logical truths that we are exploiting in this proof:

(37) **Transitivity of *iff***
A *iff* B and B *iff* C entails A *iff* C.

(38) **Substitution of 'Likes'**
If $x = y$ and $y = z$, then $x = z$.

4 The Semantics of Lexical Items

In our last meeting and in what we have done above, we developed a theory of the semantic value of intransitive verbs that identifies its extension as a function from entities to truth values:

(39) **Predicates as functions**

$$[\![\text{ VERB}_{intrans}]\!] = f : \{ x : x \text{ is an entity} \} \to \{ T, F \}$$
$$\text{for every } y \in \{ x : x \text{ is an entity} \}, f(y) = T \textit{ iff } y \text{ VERBs}$$

It will be instructive for us to spend some time reflecting on how we came to this conclusion and how we *didn't*.

We *didn't* look the verbs up in a dictionary. For however helpful such definitions of words are, they don't necessarily reveal the compositional potential of that lexical item.

And we *didn't* build the meaning of verbs on the basis of our conceptual understanding of their meanings. That is, we didn't reflect on how "dance" means to move the body (and/or its parts) to match the rhythm of music, often with predefined movements, etc.

Instead, we engaged in a specific investigative methodology aimed at appropriately deriving the truth condition of the entire utterance.

(40) **Determining the Meaning of Lexical Items**

i.) We identify the truth conditions of sentences in which a particular lexical item appears.

ii.) We consider how the lexical item systematically contributes to the truth conditions of those sentences (i.e., a form-meaning correspondence).

iii.) We consider the "known" extensions of the other lexical items in these sentences.

iv.) We consider the available rules of composition in our system (FA, NN, TN).

v.) Then, we develop a lexical entry that correctly derives the truth condition of these sentences.

We will see how this kind of approach works especially well for function words like *and*, *the*, and *every*, which can be quite difficult to intuitively define.

Admittedly, this approach does not contribute much to the meaning of content words like *dances*, *sings*, and *desk*. Indeed, it makes it possible to largely ignore much of their meaning, which is arguably a weakness of our approach.

5 Practice

Exercise. Please compute the the truth conditions of the sentence in (41).

(41) Felix snores.

To do this, you will need to provide:

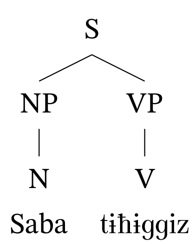
- the syntactic structure of the sentence,
- the lexical entries for the terminal nodes, and
- a proof of the truth-conditional statement: "Felix snores" is T *iff* Felix snores.

Exercise. Consider the following sentence of Tigrinya:

(42) saba tɨħɨggɨz
Saba helps
'Saba helps.'

Assume that this Tigrinya sentence has the syntactic representation in (43):

(43)



Assume also that Tigrinya has the following two lexical entries:

(44) **Lexical entries**

a. $[\![\text{Saba}]\!] = \text{Saba}$

b. $[\![\text{tɨħɨggiz}]\!] = f : \{ x : x \text{ is an entity} \} \to \{ T, F \}$
for every $y \in \{ x : x \text{ is an entity} \}$, $f(y) = T$ *iff* y helps

Please provide a proof of the truth conditional statement: "Saba tɨħɨggiz" is T *iff* Saba helps.