Semantics Bootcamp Part III: More of English
Elizabeth Coppock
NASSLLI 2012, Austin, Texas

1 Fun with Functional Application

Composition rules

For branching nodes:

1. Functional Application (FA)
   If $\alpha$ is a branching node and $\{\beta, \gamma\}$ the set of its daughters, then, for any assignment $a$, if $[[\beta]]$ is a function whose domain contains $[[\gamma]]$, then $[[\alpha]] = [[\beta]]([[\gamma]])$.

For non-branching nodes:

2. Non-branching Nodes (NN)
   If $\alpha$ is a non-branching node and $\beta$ its daughter, then, for any assignment $a$, $[[\alpha]] = [[\beta]]$.

3. Terminal Nodes (TN)
   If $\alpha$ is a terminal node occupied by a lexical item, then $[[\alpha]]$ is specified in the lexicon.

1.1 Rick Perry is conservative

4. $[[\text{Rick Perry}]] = \text{Rick Perry}$

5. $[[\text{conservative}]] = \lambda x \in D_e. x$ is conservative

What to do with $\text{is}$? How about just an identity function:

6. $[[\text{is}]] = \lambda f \in D_{(e, t)} . f$

7. $S:\ t$
   $NP: e$
   $V:\ \{e, t\} \rightarrow \text{conservative}$
   $\text{Rick Perry}\ V: \{\{e, t\}, \{e, t\}\} \rightarrow \text{A:}\ \{e, t\}$
   $\special\{\text{FA}\} = 1 $ iff $\text{Rick Perry is conservative}$

$[[\{\{\text{is}\}, \{\text{conservative}\}\}]]$

8. $[[\text{in}]] = \lambda y \in D_e . \lambda x \in D_e . x$ is in $y$

9. $[[\text{Texas}]] = \text{Texas}$

*Notice that I wrote “$= 1$ iff Rick Perry is conservative” instead of “$= 1$ iff Rick Perry is conservative”. Normally, the result of $\beta$-reduction is written by replacing the $\lambda$-bound variable by the argument to which the function is being applied, but this case is different. That is because for Heim & Kratzer, “[\$\alpha: \phi . \gamma\]$” is read as either (i) or (ii), “whichever makes sense”:

(i) “the function which maps every $\alpha$ such that $\phi$ to $\gamma$”
(ii) “the function which maps every $\alpha$ such that $\phi$ to 1, if $\gamma$, and to 0 otherwise”

When the value description is a sentence, it makes sense to read it in way (ii).
1.3 Rick Perry is proud of Texas

(10) \(\|\text{proud}\| = \lambda y \in D_e \cdot \lambda x \in D_e \cdot x \text{ is proud of } y\)

(11) \(\|\text{of}\| = \lambda x \in D_e \cdot x\)

1.4 Rick Perry is a Republican

(12) \(\|\text{Republican}\| = \lambda x \in D_e \cdot x \text{ is a Republican}\)

2 Predicate Modification

2.1 Rick Perry is a conservative Republican

Hey, let’s make the indefinite article vacuous too:

(13) \(\|a\| = \lambda f \in D_{(e,t)} \cdot f\)

(14)

(15) S: t

NP: e
V: \{\langle e, t \rangle, \langle e, t \rangle\}
A: \{\langle e, e \rangle\}
N: e

Rick Perry
is
proud

PP: e

A: \{\langle e, e \rangle\}
N: e

Texas

D: \{\langle e, t \rangle, \langle e, t \rangle\}
N': \{e, t\}

a
Republican

conservative

S: e

NP: e
V: \{\langle e, t \rangle, \langle e, t \rangle\}
PP: \{e, t\}

Rick Perry
in

N: e

Texas
Our lexical entry for \textit{conservative} from above is type \((e, t)\):

(16) \[ \llbracket \text{conservative}_1 \rrbracket = \lambda x \in D_e . \ x \text{ is conservative} \]

If we want to use Functional Application here, we need \textit{conservative} to be a function of type \((\langle e, t \rangle, \langle e, t \rangle)\).

(17) \[ \llbracket \text{conservative}_2 \rrbracket = \lambda f \in D_{\langle e, t \rangle} . \lambda x \in D_e . \ f(x) = 1 \text{ and } x \text{ is conservative} \]

Now, \[ \llbracket \llbracket \text{conservative}_1 \rrbracket \llbracket \text{Republican} \rrbracket \rrbracket \]
\[ = \llbracket \lambda x \in D_e . \ x \text{ is conservative} \rrbracket \]
\[ = \llbracket \text{conservative}_2 \rrbracket \llbracket \text{Republican} \rrbracket \rrbracket \]
\[ = \lambda f \in D_{\langle e, t \rangle} . \lambda x \in D_e . \ f(x) = 1 \text{ and } x \text{ is a Republican} \]
\[ = \lambda x \in D_e . \ x \text{ is a Republican and } x \text{ is conservative} \]

This is Montague’s strategy. All adjectives are \(\langle\langle e, t \rangle, \langle e, t \rangle\rangle\) for him, and in predicate position, they combine with a silent noun. Despite the ungrammaticality of *Rick Perry is conservative Republican.

Alternative strategy: Use another composition rule.

(18) \textbf{Predicate Modification (PM)}

If \(\alpha\) is a branching node, \(\{\beta, \gamma\}\) is the set of \(\alpha\)’s daughters, and \([\llbracket \beta \rrbracket]\) and \([\llbracket \gamma \rrbracket]\) are both in \(D_{\langle e, t \rangle}\), then \([\llbracket \alpha \rrbracket] = \lambda x \in D_e . \llbracket \beta \rrbracket(x) = \llbracket \gamma \rrbracket(x) = 1\)

Now, \[ \llbracket \llbracket \text{conservative}_1 \rrbracket \llbracket \text{Republican} \rrbracket \rrbracket \]
\[ = \lambda x \in D_e . \llbracket \text{conservative}_1 \rrbracket \llbracket \text{Republican} \rrbracket \llbracket \rrbracket(x) = 1 \]
\[ = \lambda x \in D_e . \llbracket \text{conservative}_2 \rrbracket \llbracket \text{Republican} \rrbracket \llbracket \rrbracket(x) = 1 \]
\[ = \lambda x \in D_e . \ x \text{ is conservative and } x \text{ is a Republican} \]

\section{The definite article}

What if we have the governor of Texas instead of Rick Perry?

\subsection{The negative square root of 4}

Regarding 	extit{the negative square root of 4}, Frege says, “We have here a case in which out of a concept-expression [i.e., an expression whose meaning is of type \((e, t)\)] a compound proper name is formed [that is to say, the entire expression is of type \(e\)] with the help of the definite article in the singular, which is at any rate permissible when one and only one object falls under the concept.”

“Permissible”: \(\text{the}\) denotes a function of type \((\langle e, t \rangle, e)\) that is only \textit{defined} for input predicates that characterize one single entity. In other words, \(\text{the}\) presupposes existence and uniqueness. Heim and Kratzer implement this as a restriction on the domain using the colon/dot notation:

(20) \[ \llbracket \text{the} \rrbracket = \lambda f \in D_{\langle e, t \rangle} : \text{there is exactly one } x \text{ such that } f(x) = 1 \text{. the unique } y \text{ such that } f(y) = 1 \]
So the is not a function from \( D_{(e,t)} \) to \( D_e \); it is a partial function from \( D_{(e,t)} \) to \( D_e \). But we can still give it the type \( (e, (e, t)) \) if we interpret this to allow partial functions.

To flesh out Frege’s analysis of this example further, Heim and Kratzer suggest that \textit{square root} is a “transitive noun”, with a meaning of type \( (e, (e, t)) \), and that “of” is vacuous. \([\text{square root}]\) applies to 4 via Functional Application, and the result of that composes with \([\text{negative}]\) under predicate modification.

\(21) \quad \llbracket \text{negative} \rrbracket = \lambda x \in D_e . x \text{ is negative}\)

\(22) \quad \llbracket \text{square root} \rrbracket = \lambda y \in D_e . \lambda x \in D_e . x \text{ is the square root of } y\)

\(23) \quad \llbracket \text{of} \rrbracket = \lambda x \in D_e . x\)

\(24) \quad \llbracket \text{four} \rrbracket = 4\)

So the constituents will have denotations of the following types:

3.2 Top-down evaluation

To compute the value “top-down”, we put the whole tree in one big old pair of denotation brackets, and use composition rules to break down the tree. Here I am putting the name of the rule I used as a subscript on the equals sign, because I don’t have enough room to put them off to the right.
Now we are done breaking down the tree. No more composition rules.

\[
\text{the} \left( \lambda x. [\text{negative}] (x) = [\text{square root}] (\text{of}) (4) (x) = 1 \right)
\]

\[
\text{the} \left( \lambda x. [\text{negative}] (x) = [\text{square root}] (4) (x) = 1 \right)
\]

\[
\text{the} \left( \lambda x. [\text{negative}] (x) = 1 \text{ and } x \text{ is a square root of } 4 \right)
\]

\[
\text{the} \left( \lambda x. x \text{ is negative and } x \text{ is a square root of } 4 \right)
\]

\[
\{ \lambda f \in D_{\{e,t\}} : \text{there is exactly one } x \text{ such that } f(x) = 1 \} \text{ the unique } y \text{ such that } f(y) = 1 | \lambda x. x \text{ is negative and } x \text{ is a square root of } 4
\]

\[
= \text{the unique } y \text{ such that } y \text{ is negative and } x \text{ is a square root of } 4
\]

\[
= -2
\]
3.3 Bottom-up style

We need to compute a semantic value for every subtree/node. For convenience, we can group the nodes according to the string of words that they dominate, and start from the most deeply embedded part of the tree.

Note: I reserve the right to abbreviate e.g. \([\text{sr} \ \text{of} \ \text{four}]\) as \([\text{sr} \ \text{of} \ \text{four}]\), omitting all but the outermost brackets.

**Bottom-up derivation**

- \([[\text{four}}]] = 4 \quad \text{TN}
- \([[\text{sr} \ [\text{four}]]] = [[\text{four}}]] = [[\text{four}}]] = 4 \quad \text{NN}
- \([[\text{of} \ \text{of}]] = [[\text{of}]] = \lambda x . x \quad \text{TN}, \text{NN}

- \([[\text{sr} \ \text{of} \ \text{four}}]] = [[\text{sr} \ \text{of} \ \text{four}}]] = 4 \quad \text{FA}

**square root**

- \([[\text{square root}}]] = [[\text{square root}}]] = \lambda y . \lambda x . y \ \text{is a square root of} \ x \quad \text{NN}, \text{TN}

**square root of four**

- \([[\text{square root of four}}]] = [[\text{square root of four}}]] = \lambda y . \lambda x . x \ \text{is a square root of} \ y(4) \quad \text{FA}
- \([[\text{square root of four}}]] = \lambda x . x \ \text{is a square root of} \ 4 \quad \text{FA}
- \([[\text{negative square root of four}}]] = \lambda y . \lambda x . x \ \text{is a square root of} \ 4 \quad \text{FA}
- \([[\text{negative square root of four}}]] = \lambda y . \lambda x . x \ \text{is a square root of} \ 4 \quad \text{FA}

**the**

- \([[\text{the}}]] = [[\text{the}}]] = \lambda f \in D_{(e,t)} : \text{there is exactly one} \ x \ \text{s.t.} \ f(x) = 1 . \ \text{the} \ y \ \text{such that} \ f(y) = 1 \quad \text{TN}, \text{NN}
- \([[\text{the}}]] = \lambda f \in D_{(e,t)} : \text{there is exactly one} \ x \ \text{s.t.} \ f(x) = 1 . \ \text{the} \ y \ \text{such that} \ f(y) = 1 \quad \text{TN}, \text{NN}

**the negative square root of four**

- \([[\text{the negative square root of four}}]] = \lambda x . \lambda y . \lambda x . x \ \text{is a square root of} \ 4 \quad \text{PM}
- \([[\text{the negative square root of four}}]] = \lambda x . \lambda y . \lambda x . x \ \text{is a square root of} \ 4 \quad \text{PM}
- \([[\text{the negative square root of four}}]] = \lambda x . x \ \text{is a square root of} \ 4 \quad \text{PM}
- \([[\text{the negative square root of four}}]] = \lambda x . x \ \text{is a square root of} \ 4 \quad \text{PM}

Note that we used the same composition rules as we did using the top-down style!

This style is a little bit simpler and cleaner. But it doesn’t work when you have to manipulate variable assignments... which we will cover tomorrow!