# 1 The aim of truth-conditional semantics

- In our study of the meanings of utterances, we will start with two research goals due to Frege:

($R_1$) TRUTH-CONDITIONAL SENTENCES: The meaning of a sentence is its truth-conditions.

| |
|---|
| To know the meaning of a sentence is to know how the world would have to be for the sentence to be true. |

($R_2$) COMPOSITIONALITY: The meaning of an expression is a function of the meanings of its parts.

Montague's formulation of compositionality (or, a simplified version):
Let *m* a function which maps syntactic objects to their meanings. Then for any syntactic rule $F$ of syntactic objects $e_i$, *m* is $F$-*compositional* iff there exists some semantic rule $G$ such that

$$m(F(e_1, e_2, \ldots, e_n)) = G(m(e_1), m(e_2), \ldots, m(e_n)).$$

*m* is compositional if for all syntactic rules $F$ it is $F$-compositional.

- Noteworthy features of Montague's formulation of compositionality:

a. "rule-by-rule": We match syntactic rules with semantic rules. The "function" in ($R_2$) can vary, depending on how the elements are syntactically structured.

b. strictly local composition ("context-free semantics" Dowty 2006): the meaning of a syntactic object is a function of the meaning of its immediate constituents "as a whole," i.e. insensitive to the meanings from which its constituents' meanings were formed.

c. syntactically-based: syntactic constituency forms the basis for semantic constituency.

- Our main modification: reduce the number of semantic rules to a minimum (like the reduction of transformations in syntax). motivated by elegance

# 2 Meanings and extensions

- By ($R_1$), *m*(sentence) = truth conditions.
  - Truth conditions = How the world would have to be for the sentence to be true = a way of partitioning the space of possible ways the world could be = **a function from worlds to $\{$1,0$\}$**.

(1) *m*(John runs)= the function $f : WORLDS \rightarrow \{0, 1\}$ such that

$$f(w) = \begin{cases} 1 \text{ if } w \text{ is a world where John' runs'} \\ 0\text{, otherwise.} \end{cases}$$

.

$\boxed{?}$ What about *utterance meaning* (a sentence in a particular situation)?

- Truth conditions + situational context = truth value.
  - That is: Utterance meaning is a truth value.

(2) [*m*(Pranav is human)](this-world)= 1 (assuming I am human).

- In what follows in this course, we will be considering the meanings of elements in situational context, which (following Frege) we will call the *extension*.

(3) For any syntactic object $\alpha$, the **extension** of $\alpha$, $[\![\alpha]\!]$, is [*m*($\alpha$)](this-world).

- The extension of a DP: an individual. — by intuition
- The extension of a VP: what is needed for compositionality — by choice

(4) $[\![[_{NP}$John$]]\!]$=John' — an individual, an element of $D_e$.

(5) $[\![[_S$ John runs$]]\!]$=1 iff John' runs' in $w@$. — a truth value, an element of $D_t = \{0, 1\}$

- Determining "what is needed for compositionality" requires a compositional function. Let us assume, for simplicity, only the one Frege thought would work:

(6) *Functional Application*:
If object $\alpha$ is composed of constituents $\beta$ and $\gamma$ and $[\![\beta]\!]$ is in the domain of $[\![\gamma]\!]$, then $[\![\alpha]\!]$=$[\![\gamma]\!]([\![\beta]\!])$.

- Assuming the VP and NP extensions compose by *Functional Application*, we have the following situation:

(7) a. $[\![[_S$ NP VP$]]\!]$=$[\![$VP$]\!]([\![$NP$]\!])$.
b. $[\![[_S$ NP VP$]]\!] \in D_t$,
c. $[\![$NP$]\!] \in D_e$,
d. Therefore $[\![$VP$]\!] \in D_{\langle e,t\rangle}$, a function from $D_e$ to $D_t$.

(8) $[\![$runs$]\!]$= the function $RUN : D_e \rightarrow D_t$ such that

$$RUN(x) = \begin{cases} \text{1 if x runs' in } w@ \\ \text{0, otherwise.} \end{cases}$$

.

- We will need a few additional rules of semantic composition:

(9) *Terminal Nodes*:
If object $\alpha$ is a terminal node, $[\![\alpha]\!]$ is $\alpha$'s lexical entry.

(10) *Non-branching Nodes*:
If object $\alpha$ is composed of constituent $\beta$, then $[\![\alpha]\!]$=$[\![\beta]\!]$.

• Now we can show that our system derives the truth-conditions that we intuitively expect.

**Claim**: $[\![ [_S \text{ John runs}] ]\!] = \begin{cases} 1 \text{ if John' runs' in } w@ \\ 0\text{, otherwise.} \end{cases}$

*Pf*:

(11) a. $[\![ [_{VP} [_V \text{ runs}]] ]\!] = [\![ [_V \text{ runs}] ]\!]$ by (*Non-branching Nodes*)
$= RUN \in D_{\langle e,t \rangle}$. by (*Terminal Nodes*)

b. $[\![ [_{NP} \text{ John}] ]\!] = \text{John'} \in D_e$. by (*Terminal Nodes*)

c. As John' $\in domain(RUN)$,
$[\![ [_S [_{NP} \text{ John}] [_{VP} [_V \text{ runs}]]] ]\!] = [\![ [_{VP} [_V \text{ runs}]] ]\!]([\![ [_{NP} \text{ John}] ]\!])$ by (*Functional Application*)
$= RUN(\text{John'})$ by (a) and (b)
$= \begin{cases} 1 \text{ if John' runs' in } w@ \\ 0\text{, otherwise.} \end{cases}$ by the definition of $RUN$ in (8)

• Noteworthy aspects of our system:

a. Binary branching: We have no rules to deal with ternary branching structures.
b. Insensitivity to linear order: Functional Application is not order-sensitive.
c. Insensitivity to syntactic category: None of our rules are dependent on syntactic category.
d. Type-driven: Semantic composition is governed by the types of the semantic objects.

# 3 Sets, Functions, and $\lambda$-terms

• We have decided to represent intransitive verbs as functions from individuals to truth values.

(12) $[\![\text{runs}]\!]$= the function $RUN : D_e \rightarrow D_t$ such that

$$RUN(x) = \begin{cases} 1 \text{ if x runs' in } w@ \\ 0\text{, otherwise.} \end{cases}$$

• This function is the *characteristic function* of the set $RUNNERS$ = $\{x$: $f(x) = 1\} \equiv \{x$: $x$ runs' in $w@\}$.

• Talking about verbs in terms of their characteristic sets (the sets they are characteristic functions of) allows us to describe certain things more neatly.

(13) $\{x$: $x$ runs' in $w@\} \subseteq \{x$: $x$ has-legs' in $w@\}$, therefore
$[\![\text{runs}]\!] \subseteq [\![\text{has legs}]\!]$

• We could do everything above with sets instead of functions, but things will get harder once we deal with meanings below the sentence level. But it's good to keep characteristic sets in mind for thinking about things intuitively.

• If we are keeping with functions, it would be nice to express them more compactly. That's what the $\lambda$-calculus is for.

(14) $[\![$runs$]\!]$ = $\lambda x : x \in D_e$. 1 iff x runs' in $w@$.

(15) $[\![$likes$]\!]$ = $\lambda x : x \in D_e.\lambda y : y \in D_e$. 1 iff $y$ likes $x$.