Semantic Theory
Lecture 6: Lambda Abstraction

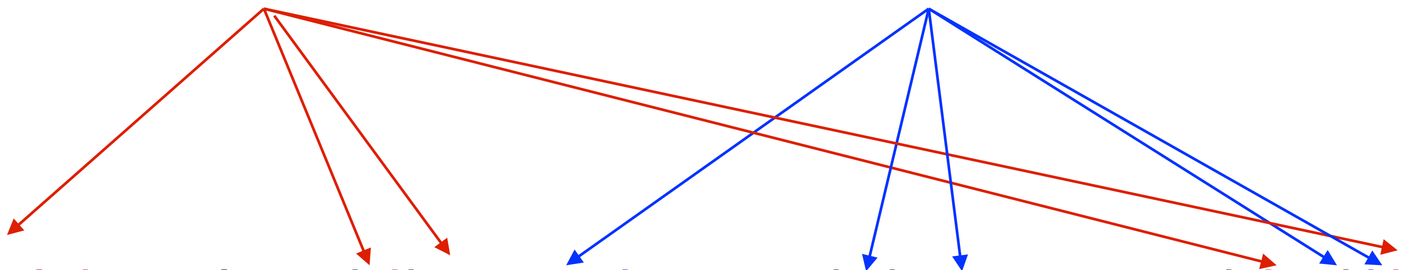
Manfred Pinkal
FR 4.7 Computational Linguistics and Phonetics

Summer 2014

UNIVERSITÄT
DES
SAARLANDES

Quantification in NL: A Challenge for Compositional Semantics

Every student presented a paper



$\forall d\ (student(d) \rightarrow \exists p\ (paper(p) \wedge present(d,p)))$

The type-theoretic analysis of quantificational NPs

- is truth-conditional and strictly compositional at the same time
- enables a unified interpretation of noun phrases of various kinds

However:

- no connection between semantic representation and FOL quantifiers (deduction calculi! theorem provers!)

How can we bring first-order logic into play again?

Lambda Abstraction: Motivation

John drinks and drives

Someone drinks and drives

Drinking is unwise

Drinking and driving is unwise

Swimming is healthy

Not smoking is healthy

Lambda Abstraction: Example

- λx[drive’(x) ∧ drink’(x)]
- … denotes the property of “being an x such that x drives and drinks”.
- λ-abstraction in the above case is applied to a type t expression, with the effect that the argument position(s) marked by variable x are “opened”, thus creating the complex predicate λx[drive’(x) ∧ drink’(x)].

drive’: ⟨e,t⟩ x:e　　　drink’: ⟨e,t⟩ x:e

drive’(x): t　　　drink’(x): t

drive’(x) ∧ drink’(x): t

j*: e　　λx[drive’(x) ∧ drink’(x)]: ⟨e,t⟩

λx[drive’(x) ∧ drink’(x)](j*): t

More Examples

John drinks and drives

$\lambda x[drink`(x) \wedge drive`(x)])(j^*)$

Someone drinks and drives

$someone`(\lambda x[drink`(x) \wedge drive`(x)])$

Drinking and driving is unwise

$\neg wise'(\lambda x.drink'(x) \wedge drive'(x))$

Not smoking is healthy

$healthy'(\lambda x.\neg smoke'(x))$

The Language of Typed Lambda Calculus

- By adding (a generalized version of) the abstraction rule to the language of Type Theory, we extend it to Typed Lambda Calculus:

  If $\alpha \in WE\tau$ and $v \in VAR_\sigma$, then $\lambda v\alpha \in WE_{\langle\sigma,\tau\rangle}$.

- Notational convention: The scope of the λ-operator is the smallest WE to its right. Wider scope must be indicated by brackets. We often use the “dot notation” λx.A indicating that the λ-operator takes widest possible scope.

Typed Lambda Calculus: Syntax

- The sets of **well-formed expressions WE_τ** for every type τ are given by:

(i) $CON_\tau \subseteq WE_\tau$ and $VAR_\tau \subseteq WE_\tau$, for every type τ

(ii) If α is in $WE_{\langle\sigma, \tau\rangle}$, β in WE_σ, then $\alpha(\beta) \in WE_\tau$.

(iii) If φ, ψ are in WE_t, then $\neg\varphi$, $(\varphi\wedge\psi)$, $(\varphi\vee\psi)$, $(\varphi\rightarrow\psi)$, $(\varphi\leftrightarrow\psi)$ are in WE_t.

(iv) If φ is in WE_t and v is a variable of arbitrary type, then $\forall v\varphi$ and $\exists v\varphi$ are in WE_t.

(v) If α, β are well-formed expressions of the same type, then $\alpha = \beta \in WE_t$.

(vi) If $\alpha \in WE\tau$ and $v \in VAR_\sigma$, then $\lambda v\alpha \in WE_{\langle\sigma,\tau\rangle}$.

Example

- $[\![\lambda x(drink'(x) \wedge drive'(x))]\!]^{M,g}$ = the f : $D_e \rightarrow D_t$ such that
  for all a $\in D_e$, f(a) = $[\![drink'(x) \wedge drive'(x)]\!]^{M,g[v/a]}$

- f(a) = 1 iff $[\![drink'(x) \wedge drive'(x)]\!]^{M,g[v/a]} = 1$
  iff $[\![drink'(x)]\!]^{M,g[v/a]} = 1$ and $[\![drive'(x)]\!]^{M,g[v/a]} = 1$
  iff $V_M(drink')([\![x]\!]^{M,g[v/a]}) = V_M(drive')([\![x]\!]^{M,g[v/a]}) = 1$
  iff $V_M(drink')(a) = V_M(drive')(a) = 1$
  iff $a \in V_M(drink') \cap V_M(drive')$

- $[\![\lambda x(drink'(x) \wedge drive'(x))(j^*)]\!]^{M,g} = 1$ iff
  $[\![\lambda x(drink'(x) \wedge drive'(x))]\!]^{M,g}([\![j^*]\!]^{M,g}) = 1$ iff
  $V_M(j^*) \in V_M(drink') \cap V_M(drive')$

Lambda Abstraction: Interpretation

- $[\![\lambda v \alpha]\!]^{M,g}$ is that function $f : D_\sigma \rightarrow D_\tau$ such that

  for all $a \in D_\sigma$, $f(a) = [\![\alpha]\!]^{M,g[v/a]}$

  (for $\alpha \in WE_\tau$, $v \in VAR_\sigma$)

Typed Lambda Calculus: Interpretation

- **Interpretation with respect to** a model structure **M = ⟨U, V⟩** and a variable assignment **g**:

(i) $[\![\alpha]\!]^{M,g} = V(\alpha)$, if α is a constant

$[\![\alpha]\!]^{M,g} = g(\alpha)$, if α is a variable

(ii) $[\![\alpha(\beta)]\!]^{M,g} = [\![\alpha]\!]^{M,g}([\![\beta]\!]^{M,g})$

(iii) $[\![\neg\varphi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 0$

$[\![\varphi \wedge \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 1$ and $[\![\psi]\!]^{M,g} = 1$

$[\![\varphi \vee \psi]\!]^{M,g} = 1$ iff $[\![\varphi]\!]^{M,g} = 1$ or $[\![\psi]\!]^{M,g} = 1$

…

(iv) $[\![\alpha = \beta]\!]^{M,g} = 1$ iff $[\![\alpha]\!]^{M,g} = [\![\beta]\!]^{M,g}$

(v) $[\![\exists v\varphi]\!]^{M,g} = 1$ iff there is a $d \in D_\tau$ such that $[\![\varphi]\!]^{M,g[v/d]} = 1$, if $v \in VAR_\tau$

$[\![\forall v\varphi]\!]^{M,g} = 1$ iff for all $d \in D_\tau$: $[\![\varphi]\!]^{M,g[v/d]} = 1$, if $v \in VAR_\tau$

(vi) $[\![\lambda v\alpha]\!]^{M,g} = f : D_\sigma \to D_\tau$, with $f(a) = [\![\alpha]\!]^{M,g[v/a]}$ for all $a \in D_\sigma$,

if $v \in VAR_\sigma$ and $\alpha \in WE_\tau$

NL Quantifier Expressions: Interpretation

- someone' $\in CON_{\langle\langle e,t\rangle,t\rangle}$, so V_M(someone') $\in D_{\langle\langle e,t\rangle,t\rangle}$
- $D_{\langle\langle e,t\rangle,t\rangle}$ is the set of functions from $D_{\langle e,t\rangle}$ to D_t , i.e.,

  the set of functions from $\mathcal{P}(U)$ to $\{0,1\}$,

  which in turn is equivalent to $\mathcal{P}(\mathcal{P}(U))$.
- Thus, V_M(someone') $\subseteq \mathcal{P}(U_M)$. More specifically:
- V_M(someone') = $\{S \subseteq U_M \mid S \neq \emptyset\}$, if U_M is a domain of persons

Example, Continued

- If the λ-expression is applied to an overt argument, we can simplify the interpretation:
  - $[\![\lambda v \alpha(\beta)]\!]^{M,g} = [\![\alpha]\!]^{M,g[v/[\![\beta]\!]^{M,g}]}$

- Example:
  - $[\![\lambda x(\text{drink'}(x) \wedge \text{drive'}(x))(j^*)]\!]^{M,g} = 1$
  - iff $[\![\text{drink'}(x) \wedge \text{drive'}(x)]\!]^{M,g[x/[\![j^*]\!]^{M,g}]} = 1$
  - iff $[\![\text{drink'}(x) \wedge \text{drive'}(x)]\!]^{M,g[x/V_M(j^*)]} = 1$
  - iff $[\![\text{drink'}(x)]\!]^{M,g[x/V_M(j^*)]} = [\![\text{drive'}(x)]\!]^{M,g[x/V_M(j^*)]} = 1$
  - iff $V_M(\text{drink'})(V_M(j^*)) = V_M(\text{drive'})(V_M(j^*)) = 1$
  - iff $V_M(j^*) \in V_M(\text{drink'}) \cap V_M(\text{drive'})$

Lambda Conversion

- $[\![\lambda v\alpha(\beta)]\!]^{M,g} = [\![\alpha]\!]^{M,g[v/[\![\beta]\!]^{M,g}]}$
- This implies that all (free) occurrences of the λ-variable v in α get the interpretation of β as value.
- Then: Why not first substitute all free occurrences of of v in α with β (notation: [β/v]α), and then interpret the simplified expression?
- Actually, the original expression and the result of the substitution are truth-conditionally equivalent (under certain conditions!):
- $[\![\lambda v\alpha(\beta)]\!]^{M,g} = [\![\alpha]\!]^{M,g[v/[\![\beta]\!]^{M,g}]} = [\![[\beta/v]\alpha]\!]^{M,g}$
- **$\lambda v\alpha(\beta) \Leftrightarrow [\beta/v]\alpha$**

Lambda Conversion: Examples

- *drinks and drives* ➔ $\lambda x[\text{drink}'(x) \wedge \text{drive}'(x)] : \langle e,t \rangle$
- *John* ➔ $j^*:e$
- *John drinks and drives* ➔ $\lambda x[\text{drink}'(x) \wedge \text{drive}'(x)](j^*) : t$

$$\Leftrightarrow [j^*/x]\ \text{drink}'(x) \wedge \text{drive}'(x)$$

$$= \text{drink}'(j^*) \wedge \text{drive}'(j^*)$$

- *John* ➔ $\lambda F.F(j^*): \langle\langle e,t \rangle,t \rangle$
- *John drinks and drives* ➔ $\lambda F[F(j^*)](\lambda x[\text{drink}'(x) \wedge \text{drive}'(x)]) : t$

$$\Leftrightarrow \lambda x[\text{drink}'(x) \wedge \text{drive}'(x)](j^*)$$

$$\Leftrightarrow \text{drink}'(j^*) \wedge \text{drive}'(j^*)$$