A proof theoretical account of polarity items and monotonic inference.

Raffaella Bernardi

UiL OTS, University of Utrecht

e-mail: Raffaella.Bernardi@let.uu.nl

Url: http://www.let.uu.nl/~Raffaella.Bernardi/personal

Contents

1. Monotonicity Calculus

$f \circ g$	$=$	h	$h : A^z \longrightarrow C$
$\uparrow$Mon $\circ$ $\uparrow$Mon	$=$	$\uparrow$Mon	$h : A^{sg(+,+)} \longrightarrow C$
$\downarrow$Mon $\circ$ $\downarrow$Mon	$=$	$\uparrow$Mon	$h : A^{sg(-,-)} \longrightarrow C$
$\uparrow$Mon $\circ$ $\downarrow$Mon	$=$	$\downarrow$Mon	$h : A^{sg(+,-)} \longrightarrow C$
$\downarrow$Mon $\circ$ $\uparrow$Mon	$=$	$\downarrow$Mon	$h : A^{sg(-,+)} \longrightarrow C$

2. Polarity and Monotone Positions

Definition [Polarity of Occurrences]

Given a lambda term N and a subterm M of N. A specified occurrence of M in N, is called **positive** (**negative**) according to the following clausules:

i. M is positive in M.
ii. M is positive (negative) in PQ iff M is positive (negative) in P.
iii. M is positive (negative) in PQ iff M is positive (negative) in Q, and P denotes an upward monotone function.
iv. M is negative (positive) in PQ iff M is positive (negative) in Q, and P denotes a downward monotone function.
v. M is positive (negative) in $\lambda X.P$ iff M is positive (negative) in P and $X \notin FV(M)$.

Definition[Monotone position]

Let N'_α be a lambda term like N_α except for containing an occurrence of M'_β where N_α contains M_β, *i.* N_α is **upward monotone in** M_β iff for all models and assignments $[\![M]\!]^f_\mathcal{M} \leq_\beta [\![M']\!]^f_\mathcal{M}$ entails $[\![N]\!]^f_\mathcal{M} \leq_\beta [\![N']\!]^f_\mathcal{M}$;

ii. N_α is **downward monotone in** M_β iff for all models and assignments $[\![M]\!]^f_\mathcal{M} \leq_\beta [\![M']\!]^f_\mathcal{M}$ entails $[\![N']\!]^f_\mathcal{M} \leq_\beta [\![N]\!]^f_\mathcal{M}$.

3. Partial Order

Taking advantage of the fact that the denotation of all expressions of natural language can at end be reduced to sets, we can extend our model with a partial order defined recursively by means of types. Let $\mathcal{M} = \langle D, \leq, I \rangle$, be our model, where $\leq$ is recursively defined as follows:

If $\beta, \gamma \in Dom_e$, then	$[\![\beta]\!] \leq_e [\![\gamma]\!]$	iff	$[\![\beta]\!] = [\![\gamma]\!]$
If $\beta, \gamma \in Dom_t$, then	$[\![\beta]\!] \leq_t [\![\gamma]\!]$	iff	$[\![\beta]\!] = 0$ or $[\![\gamma]\!] = 1$
If $\beta, \gamma \in Dom_{(a,b)}$, then	$[\![\beta]\!] \leq_{(a,b)} [\![\gamma]\!]$	iff	$\forall \alpha \in Dom_a, [\![\beta(\alpha)]\!] \leq_b [\![\gamma(\alpha)]\!]$

4. Semantics

$$
\begin{array}{lcl}
[\![\text{no N}]\!] & = & \{X \subseteq E | [\![\text{N}]\!] \cap X = \emptyset\} \\
[\![\text{some N}]\!] & = & \{X \subseteq E | [\![\text{N}]\!] \cap X \neq \emptyset\} \\
[\![\text{every N}]\!] & = & \{X \subseteq E | [\![N]\!] \subseteq X\}
\end{array}
$$

Determining the truth-value of an expression is reduced to simple set theoretical operations e.g. inclusion, memebership, intersection. For example, checking whether in a given model $\mathcal{M}$ the sentence “Every student walks” is true, means to determine whether $[\![\text{every student (walks)}]\!] = 1$. This is done by means a simple calculation:

$$
\begin{array}{lll}
[\![\text{every student (walks)}]\!] = 1 & \text{iff} & [\![\text{every student}]\!]([\![\text{walks}]\!]) = 1 \\
 & \text{iff} & [\![\text{walks}]\!] \in [\![\text{every student}]\!] \\
 & \text{iff} & [\![\text{student}]\!] \subseteq [\![\text{walks}]\!] \\
 & \text{iff} & \forall x \in [\![\text{student}]\!]\ x \in [\![\text{walks}]\!].
\end{array}
$$