Presupposition Accommodation as Exception Handling

Philippe de Groote, Ekaterina Lebedeva

INRIA Nancy - Calligramme
CAuLD Workshop “Logical Methods for Discourse”

2010, December 13th

Introduction (and References)

- Montague Semantics [Montague (1970a, 1970b, 1973)]
- Dynamic Logic [de Groote (2006)]
- DRT [Kamp, 1981]
- “Presupposition as anaphora” [van der Sandt (1992)]

Natural language expressions can be categorized by types. The type of any expression can be constructed from two basic types:

ι	type of individuals
o	type of propositions

$$[\![NP]\!] = (\iota \rightarrow o) \rightarrow o$$

Each lexical item is assigned a λ-term.

$$[\![John]\!] = \lambda P.P\mathbf{j}$$

The meaning of a whole sentence is obtained by composing the λ-terms via functional application.

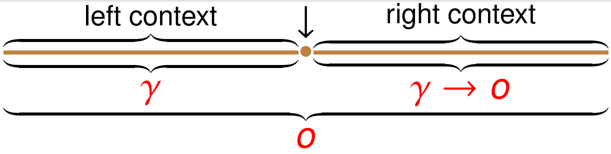
Montague's theory is limited to single sentences.

General goal

Express lexical semantics of natural language

- compositional (Montague's homomorphism requirement)
- in a logical language (we use simply typed λ-calculus) having model-theoretic interpretation
- dynamic

Dynamic logic (de Groote, 2006)



$$\mathbb{S} : \gamma \to (\gamma \to o) \to o \qquad \doteq \Omega$$

$e : \gamma$	the left context, the environment,
	a representation of what has been processed
$\phi : \gamma \to o$	the right context,
	what is still to be processed
	the continuation of the λ – term
sel	a choice function for anaphora resolution

Geurts "Good news about the Description theory of names", 1997

- there are no fundamental semantic differences between names and definite NPs
- names and definite NPs are presuppositional expressions

Geurt's paper gives linguistic and philosophical insight but no technical details.

Proper Names

How it used to be:

$$[\![NP]\!] = (\iota \to o) \to o$$

$$[\![John]\!] = \lambda P.P\mathbf{j}$$

What is changing:

$$o \rightsquigarrow \Omega \qquad\qquad \iota \rightsquigarrow (\gamma \to \iota)$$

How it is now:

$$[\![NP]\!] = ((\gamma \to \iota) \to \Omega) \to \Omega$$

$$\text{named “John”} : \iota \to o$$

$$\begin{aligned} sel &: (\iota \to o) \to \gamma \to \iota \\ sel\ (\text{named “John”}) &: \gamma \to \iota \end{aligned}$$

$$[\![John]\!] = \lambda P.P(sel(\text{named “John”}))$$

Pronouns

When he woke up, John felt better.

$$[\![he]\!] = \lambda P.P(sel\ (\lambda x.\mathbf{human}(x) \wedge \mathbf{masculine}(x)))$$

Possessives

John's car is red.

$$[\![John's\ car]\!] = \lambda P.P(\lambda e.sel(\lambda x.\mathbf{car}x \wedge \mathbf{poss}\ x\ sel(\text{named "John"})e)e)$$

$$[\![John]\!] = \lambda P.P(sel(\text{named "John"}))$$

$$[\![car]\!] = \lambda \bar{x}.\lambda e\phi.\mathbf{car}(\bar{x}e) \wedge \phi e$$

$$[\![{}'s]\!] = \lambda YX.\lambda P.P(SEL(\lambda \bar{x}.((X\bar{x}) \curlywedge Y([\![poss]\!]\bar{x}))))$$

$$\curlywedge :\ \Omega \rightarrow (\Omega \rightarrow \Omega)$$

$$A \curlywedge B = \lambda e\phi.Ae(\lambda e.Be\phi)$$

$$[\![poss]\!] = \lambda \bar{x}\bar{y}.\lambda e\phi.\mathbf{poss}(\bar{x}e)(\bar{y}e) \wedge \phi e$$

Possessives

$$[\![{}'s]\!] = \lambda YX.\lambda P.P(SEL(\lambda\bar{x}.((X\bar{x}) \wedge Y([\![poss]\!]\bar{x}))))$$

$$sel : (\quad \iota \quad \to o) \to \gamma \to \quad \iota$$
$$SEL : ((\gamma \to \iota) \to \Omega) \to \gamma \to (\gamma \to \iota)$$

$$SEL = \lambda Pe.sel(\lambda x.P(\lambda e.x)e(\lambda e.\top))e$$

$$[\![{}'s]\!][\![John]\!][\![car]\!] \to_{\beta}^{*}$$
$$\lambda P.P(\lambda e.sel(\lambda x.\mathbf{car}x \wedge \mathbf{poss}\ x\ sel(\text{named “John”})e)e)$$

Formal Details

Sentence: $\mathbb{S} : \gamma \to (\gamma \to o) \to o \quad [\doteq \Omega]$

Discourse: $\mathbb{D} : (\gamma \to o) \to o$

Type of environment: $\gamma = \texttt{list of } (\iota \times o)$

List constructor: $:: : (\iota \times o) \to \gamma \to \gamma$

Selection function: $sel : (\iota \to o) \to \gamma \to \iota$

$$sel\ Q\ [\,] = \texttt{Raise (Exception Q)}$$
$$sel\ Q\ ((x, \Sigma) :: tl) = \texttt{if } (\Sigma \vdash Qx) \texttt{ then } x \texttt{ else } (sel\ Q\ tl)$$

Discourse Update

$$\mathbb{D} \circlearrowleft \mathbb{S} \quad = \lambda\phi.\, \mathbb{D}(\lambda e.\mathbb{S} e \phi)$$

```
handle (Exception Q) with
```

$$\lambda\phi.\mathbb{D}(\lambda e.\exists x.(Qx) \wedge \phi((x, Qx) :: e)) \circlearrowleft \mathbb{S}$$

Example

John loves Mary

$$[\![John]\!] = \lambda P.P(sel(\text{named "John"}))$$
$$[\![Mary]\!] = \lambda P.P(sel(\text{named "Mary"}))$$
$$[\![love]\!] = \lambda YX.X(\lambda\bar{x}.Y(\lambda\bar{y}.(\lambda e\phi.\mathbf{love}(\bar{x}e)(\bar{y}e) \wedge \phi e)))$$

$$\mathbb{S} = \lambda e\phi.\,(\mathbf{love}\,(sel\,(\text{named "John"})\,e)\,(sel\,(\text{named "Mary"})\,e))\ \wedge\ (\phi\,e)$$

$$\mathbb{D} \circlearrowleft \mathbb{S} \quad = \lambda\phi.\, \mathbb{D}(\lambda e.\mathbb{S} e \phi)$$

`handle (Exception Q) with`

$$\lambda\phi.\mathbb{D}(\lambda e.\exists x.(Qx) \wedge \phi((x, Qx) :: e)) \circlearrowleft \mathbb{S}$$

$$\mathbb{S} = \lambda e\phi.\, (\mathbf{love}\, (sel\, (\text{named “John”})\, e)\, (sel\, (\text{named “Mary”})\, e))\ \wedge\ (\phi\, e)$$

$$\mathbb{D}_0 = \lambda\phi.\phi\texttt{nil}$$

$$\mathbb{D}_0 \circlearrowleft \mathbb{S} = \lambda\phi.\mathbb{D}_0(\lambda e.\mathbb{S} e\phi)$$

⇝ exception on property (named “John”)

$$(\lambda\phi.\exists j.(\text{named “John”}j) \wedge \phi((j, \text{named “John”}j) :: \texttt{nil})) \circlearrowleft \mathbb{S}$$

⇝ exception on property (named “Mary”)

$$e_{mj} \equiv ((m, \text{named “Mary”}m) :: (j, \text{named “John”}j) :: \texttt{nil})$$

$$(\lambda\phi.\exists j.(\text{named “John”}j) \wedge \exists m.(\text{named “Mary”}m) \wedge \phi e_{mj}) \circlearrowleft \mathbb{S} \to^*_\beta$$

$$\lambda\phi.\exists j.(\text{named “John”}j) \wedge \exists m.(\text{named “Mary”}m) \wedge \\ \quad \mathbf{love}\ (sel\ (\text{named “John”})e_{mj})(sel(\text{named “Mary”})e_{mj}) \wedge \phi e_{mj} \to^*_\beta$$

$$\lambda\phi.\exists j.(\text{named “John”}j) \wedge \exists m.(\text{named “Mary”}m) \wedge \mathbf{love}\ (j, m) \wedge \phi e_{mj}$$

- $f(\texttt{Exception Q}) \rightsquigarrow (\texttt{Exception Q})$
- $(\texttt{Exception Q})(t) \rightsquigarrow (\texttt{Exception Q})$

$$\exists x.Q(x) = \exists(\lambda x.Q(x))$$

- $\lambda x.(\texttt{Exception Q})$ where $x \notin FV(\texttt{Q}) \rightsquigarrow (\texttt{Exception Q})$
- $\lambda x.(\texttt{Exception Q})$ where $x \in FV(\texttt{Q}) \rightsquigarrow$??? $(\texttt{Exception Q})$

Example

John loves his car

$$[\![John]\!] = \lambda P.P(sel(\text{named "John"}))$$
$$[\![love]\!] = \lambda YX.X(\lambda\bar{x}.Y(\lambda\bar{y}.(\lambda e\phi.\mathbf{love}(\bar{x}e)(\bar{y}e) \wedge \phi e)))$$

$$[\![{}'s]\!][\![he]\!][\![car]\!] \rightarrow^*_\beta$$
$$\lambda Q.Q(\lambda e.sel(\lambda x.\mathbf{car}x \wedge \mathbf{poss}\ x\ sel(\lambda x.\mathbf{masc}x)e)e)$$

$$\mathbb{S} = [\![love]\!][\![{}'s]\!][\![he]\!][\![car]\!][\![John]\!] \rightarrow^*_\beta$$
$$\lambda e\phi.\mathbf{love}(sel(\text{named "John"})e)$$
$$(sel(\lambda x.\mathbf{car}x \wedge \mathbf{poss}\ x\ (sel(\lambda x.\mathbf{masc}x)e)e) \wedge \phi e$$

$$\mathbb{D} \circlearrowleft \mathbb{S} = \lambda\phi.\, \mathbb{D}(\lambda e.\mathbb{S} e\phi)$$

$$\texttt{handle (Exception Q) with}$$

$$\lambda\phi.\mathbb{D}(\lambda e.\exists x.(Qx) \wedge \phi((x, Qx) :: e)) \circlearrowleft \mathbb{S}$$

$$\mathbb{D}_0 = \lambda\phi.\phi\texttt{nil}$$

$$\mathbb{S} = \lambda e\phi.\textbf{love}(sel(\text{named “John”})e) \\ (sel(\lambda x.\textbf{car}x \wedge \textbf{poss}\ x\ (sel(\lambda x.\textbf{masc}x)e)e) \wedge \phi e$$

$$\mathbb{D}_1 = \lambda\phi.\exists j.(\text{named “John”}j) \wedge \phi((j, \text{named “John”}j) :: \texttt{nil})$$

$$\mathbb{D}_1 \circlearrowleft \mathbb{S} = \lambda\phi\exists j.(\text{named “John”}j) \wedge \textbf{love}\ j\ (\texttt{Exception Q})$$

$$Q = \lambda x.\textbf{car}x \wedge \textbf{poss}\ x\ j$$

$$\mathbb{D}_1 \circlearrowleft \mathbb{S} \; = \; \lambda\phi.\exists j.(\text{named “John”} j) \wedge \textbf{love}\ j\ (\texttt{Exception Q})$$

$$\lambda\phi.\exists j.(\text{named “John”} j) \wedge (\texttt{Exception Q})$$

$$\lambda\phi.\exists j.(\texttt{Exception Q})$$

$$\lambda\phi.\exists(\lambda j.\texttt{Exception}(\lambda x.\textbf{car} x \wedge \textbf{poss}\ x\ j))$$

$$\texttt{Exception}(\lambda x.\textbf{car} x \wedge \textbf{poss}\ x\ j)$$

$$\texttt{handle}\ (\texttt{Exception}\ (\lambda x.\textbf{car} x \wedge \textbf{poss}\ x\ j))$$

$$\to^*_\beta \lambda\phi.\exists j.(\text{named “John”} j) \wedge \exists c.(\textbf{car} c \wedge \textbf{poss}\ c\ j) \wedge \textbf{love}\ j\ c \wedge \phi((c, (\textbf{car} c \wedge \textbf{poss}\ c\ j)) :: (j, \text{named “John”} j) :: \texttt{nil})$$

Conclusion

- Modeling the presupposition phenomenon within Montague's Theory of formal semantics
- Used exception raising and handling mechanisms
- Showed the approach on presuppositions triggered by definite descriptions (global accommodation)
- The approach can be extended for other presupposition triggers
- Future work:
  Develop the representation of environment, accounts for local/intermediate accommodation, . . .

The End

Thank you!

Questions?