SDRT and Continuation Semantics

(CAuLD project)

Nicholas Asher[1] Sylvain Pogodalla[2]

[1]nicholas.asher@irit.fr
IRIT CNRS, Toulouse, France
[2]sylvain.pogodalla@loria.fr
LORIA/INRIA Nancy–Grand Est, France

CAULD – Nancy
December 13–14, 2010

Outline

Linguistic Phenomena

Example

1. John walked in. ▲ He poured himself a cup of coffee.
2. John fell. ▲ Mary pushed him.
3. We bought the apartment, ▲ but we've rented it.
4. Il commence à dessiner et peindre en 1943 , ▲ fréquente les ateliers de sculpture ▲ puis de peinture de l' école des Beaux-Arts d' Oran , ▲ où il rencontre Guermaz. *(ANNODIS corpus)*
5. Julie had an excellent meal, ▲ beginning with an elegant and inventive truffes du Périgord en première cuisson comme un petit déjeuner,▲ followed by some wonderful scallops, ▲ then sweetbreads, ▲ a sumptuous cheese plate, ▲ and ending with a scrumptious dessert.

Discourse Structure and SDRT [Asher and Lascarides(2003)]

Example (Hierarchical structure of the discourse)

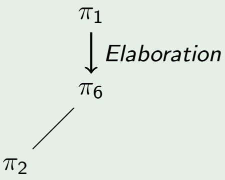
π_1

- (π_1) John had a great evening last night.

Discourse Structure and SDRT [Asher and Lascarides(2003)]

Example (Hierarchical structure of the discourse)

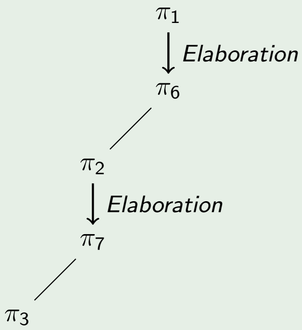
- (π_1) John had a great evening last night.
- (π_2) He had a great meal.



Discourse Structure and SDRT [Asher and Lascarides(2003)]

Example (Hierarchical structure of the discourse)

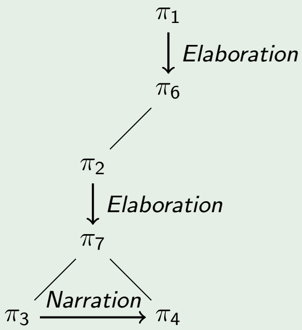
- (π_1) John had a great evening last night.
- (π_2) He had a great meal.
- (π_3) He ate salmon.



Discourse Structure and SDRT [Asher and Lascarides(2003)]

Example (Hierarchical structure of the discourse)

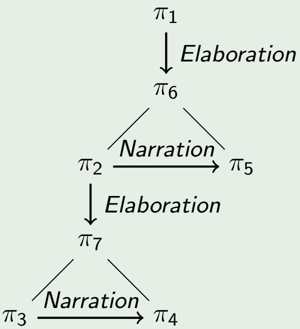
- (π_1) John had a great evening last night.
- (π_2) He had a great meal.
- (π_3) He ate salmon.
- (π_4) He devoured lots of cheese.



Discourse Structure and SDRT [Asher and Lascarides(2003)]

Example (Hierarchical structure of the discourse)

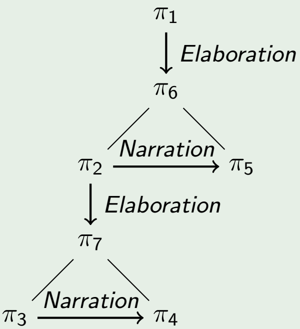
- (π_1) John had a great evening last night.
- (π_2) He had a great meal.
- (π_3) He ate salmon.
- (π_4) He devoured lots of cheese.
- (π_5) He then won a dancing competition.



Discourse Structure and SDRT [Asher and Lascarides(2003)]

Example (Hierarchical structure of the discourse)

- (π_1) John had a great evening last night.
- (π_2) He had a great meal.
- (π_3) He ate salmon.
- (π_4) He devoured lots of cheese.
- (π_5) He then won a dancing competition.



$$Elaboration(\pi_1, \pi_6, \pi_0) \wedge Elaboration(\pi_2, \pi_7, \pi_6) \wedge Narration(\pi_3, \pi_4, \pi_7) \wedge Narration(\pi_2, \pi_5, \pi_6)$$

Building SDRS's

- Segment a text into EDUs
- Compute attachment points
- Compute discourse relations between an EDU and its attachment point

Dynamic Logics in Discourse

Technical and Conceptual Issues

- Non-standard interpretation of formulas using assignment functions (cf. Sylvain's talk)
- interactions between syntax, compositional semantics and discourse very separated in [Asher and Lascarides(2003)]

Dynamic Logics in Discourse

Technical and Conceptual Issues

- Non-standard interpretation of formulas using assignment functions (cf. Sylvain's talk)
- interactions between syntax, compositional semantics and discourse very separated in [Asher and Lascarides(2003)]

Formal Semanticist or Logician?

- What are the useful data to feed the context with?
- How do discourse and sentences combine?
- What are the semantic recipes of the lexical items
- Should I design a new logic?

Dynamic Logics in Discourse

Technical and Conceptual Issues

- Non-standard interpretation of formulas using assignment functions (cf. Sylvain's talk)
- interactions between syntax, compositional semantics and discourse very separated in [Asher and Lascarides(2003)]

Formal Semanticist or Logician?

- What are the useful data to feed the context with?
- How do discourse and sentences combine?
- What are the semantic recipes of the lexical items
- Should I design a new logic? Continuation semantics

Continuation Semantics

Principles [de Groote(2006)]

$[\![s]\!]$

Continuation Semantics

Principles [de Groote(2006)]

$$[\![s]\!]$$
$$[\![np]\!] = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!]$$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{aligned}
&[\![s]\!] \\
&[\![np]\!] && = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\
&[\![n]\!] && = e \rightarrow [\![s]\!]
\end{aligned}
$$

Continuation Semantics

Principles [de Groote(2006)]

$$\begin{array}{ll}
[\![s]\!] & = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \quad \triangleq \Omega \\
[\![np]\!] & = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\
[\![n]\!] & = e \rightarrow [\![s]\!]
\end{array}$$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{aligned}
[\![s]\!] &= \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \quad \triangleq \Omega \\
[\![np]\!] &= (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\
[\![n]\!] &= e \rightarrow [\![s]\!]
\end{aligned}
$$

Example

A man is sleeping.

Continuation Semantics

Principles [de Groote(2006)]

$$\begin{aligned} [\![s]\!] &= \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \triangleq \Omega \\ [\![np]\!] &= (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\ [\![n]\!] &= e \rightarrow [\![s]\!] \end{aligned}$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

Continuation Semantics

Principles [de Groote(2006)]

$$\begin{aligned}
[\![s]\!] &= \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t \quad \triangleq \Omega \\
[\![np]\!] &= (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] \\
[\![n]\!] &= e \rightarrow [\![s]\!]
\end{aligned}$$

Example

A man is sleeping. He is snoring.

$\lambda i.\lambda k.\exists x.\ (\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{array}{lll}
[\![s]\!] & = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t & \triangleq \Omega \\
[\![np]\!] & = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] & \\
[\![n]\!] & = e \rightarrow [\![s]\!] &
\end{array}
$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i)$

Continuation Semantics

Principles [de Groote(2006)]

$$
\begin{array}{lll}
[\![s]\!] & = \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t & \triangleq \Omega \\
[\![np]\!] & = (e \rightarrow [\![s]\!]) \rightarrow [\![s]\!] & \\
[\![n]\!] & = e \rightarrow [\![s]\!] &
\end{array}
$$

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i)$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.	He is snoring.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$	$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge k\ (x :: i)]\ i$$
$$(\lambda i'.(\lambda i\ k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$
$$\rightarrow_\beta \quad \lambda i\ k.[\lambda k.\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))]$$
$$(\lambda i'.(\textbf{snoring}\ (\texttt{sel}\ i')) \wedge (k\ i'))$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i)$

$$\lambda i\ k.[\lambda i\ k.\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge k\ (x :: i)]\ i\ (\lambda i'.(\lambda i\ k.(\textbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i))\ i'\ k)$$

$$\rightarrow_\beta \quad \lambda i\ k.[\lambda k.\exists x.(\textbf{man}\ x) \wedge (\textbf{sleeping}\ x) \wedge (k\ (x :: i))]\ (\lambda i'.(\textbf{snoring}\ (\texttt{sel}\ i')) \wedge (k\ i'))$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i)$

$$\begin{array}{ll} & \lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i \\ & \qquad (\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\texttt{sel}\ i)) \wedge (k\ i))\ i'\ k) \\ \rightarrow_\beta & \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))] \\ & \qquad (\lambda i'.(\mathbf{snoring}\ (\texttt{sel}\ i')) \wedge (k\ i')) \\ \rightarrow_\beta & \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\lambda i'.(\mathbf{snoring}\ (\texttt{sel}\ i')) \wedge (k\ i'))\ (x :: i))] \end{array}$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\begin{array}{ll} & \lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i \\ & \qquad (\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k) \\ \to_\beta & \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))] \\ & \qquad (\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i')) \\ \to_\beta & \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))\ (x :: i))] \end{array}$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$
\begin{array}{ll}
 & \lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i \\
 & \qquad (\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k) \\
\rightarrow_\beta & \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))] \\
 & \qquad (\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i')) \\
\rightarrow_\beta & \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))\ (x :: i))] \\
\rightarrow_\beta & \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\mathbf{snoring}\ (\mathtt{sel}\ (x :: i)) \wedge (k\ (x :: i))))]
\end{array}
$$

CS: an Example

$$[\![T.S]\!] = \lambda i.\lambda k.[\![T]\!]\ i\ (\lambda i'.[\![S]\!]\ i'\ k)$$

Example

A man is sleeping.
$\lambda i.\lambda k.\exists x.\ (\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))$

He is snoring.
$\lambda i.\lambda k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i)$

$$\begin{array}{ll}
 & \lambda i\ k.[\lambda i\ k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge k\ (x :: i)]\ i \\
 & \qquad (\lambda i'.(\lambda i\ k.(\mathbf{snoring}\ (\mathtt{sel}\ i)) \wedge (k\ i))\ i'\ k) \\
\rightarrow_\beta & \lambda i\ k.[\lambda k.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge (k\ (x :: i))] \\
 & \qquad (\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i')) \\
\rightarrow_\beta & \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\lambda i'.(\mathbf{snoring}\ (\mathtt{sel}\ i')) \wedge (k\ i'))\ (x :: i))] \\
\rightarrow_\beta & \lambda i\ k.[\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge ((\mathbf{snoring}\ (\mathtt{sel}\ (x :: i)) \wedge (k\ (x :: i))))]
\end{array}$$

SDRT in Continuation Semantics

- A set of labels $\pi, \pi_1, \pi_2, \ldots : \ell$, representing discourse constituents
- $R(\pi_1, \pi_2, \pi) : t$, a set of relation symbols that represent discourse relations over constituents, where R is a relation symbol for a discourse relation. This formula says that the discourse relation R holds between π_1 and π_2 in constituent π.
- $\Omega \triangleq \gamma \to (\gamma \to \ell \to t) \to \ell \to t$

Option 1: more complicated sentential semantics

- Left contexts are records
- Binder rule is as before.
- Sentence semantics is more complicated

Sentential semantics

- $?_R(\pi_S, ?, ?) \wedge \pi_S : \|S\|$

That is, a sentence requires the resolution of an attachment point in some environment with some discourse relation.
In CS, this means:

$$[\![S]\!] = \lambda io.\exists \pi_s.P_S \wedge \mathtt{sel}_\rho(\mathtt{sel}_L(i), \pi_s, \mathtt{sel}_L(i)) \wedge (o\,\upsilon(i, \pi_2)) \quad (1)$$

Exceptions in SDRT

The sentential semantics rule presupposes that there are at least two labels in the left context. When this is not met, we have the exception handling clause:

$$[\![S]\!] = \lambda io.\exists\pi.\exists\pi_s.P_S \wedge \mathtt{sel}_\rho(\mathtt{sel}_L(i), \pi_s, \pi) \wedge (o\,\upsilon(i, \pi_S)) \qquad (2)$$

Need another exception when there is no label at all in the context (discourse initial segment).

Option 2: Complicate the Binder rule

$$\begin{aligned} [\![D.S]\!] &= \lambda io.\exists \pi_1.[\![D]\!](\pi_1 :: i)(\lambda i'.\exists \pi_2.[\![S]\!](i')(0) \\ &\quad \wedge \mathtt{sel}_\rho(\mathtt{sel}_L(i'), \pi_s, \mathtt{sel}_L(i')) \wedge (o\, \upsilon(i', \pi_2)) \end{aligned}$$

Avoids the need for the exception when we have a discourse initial segment.

Glueing functions

- $\mathtt{sel}_L : \gamma \rightarrow \ell$ extracts a label from the left context that is SDRT accessible
- $\mathtt{sel}_E : \gamma \rightarrow \ell \rightarrow e$ extracts a discourse referent from the set of accessible discourse referents associated with a label.
- $\mathtt{sel}_\rho : \gamma \rightarrow \ell \rightarrow \ell \rightarrow \ell \rightarrow t$. (*i.e.* a ternary relation) linking a label chosen from i the current context and returns a proposition.
- $\upsilon : \gamma \rightarrow \ell \rightarrow \gamma$.
  υ changes the left context record in virtue of $\|S\|$ and its link to the context.

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.

$$
\begin{aligned}
[\![man]\!] &= \lambda x.\lambda i o \pi.(\mathbf{M}\, x\, \pi) \wedge (o\, i\, \pi) \\
[\![a]\!] &= \lambda P.\lambda Q.\lambda i o \pi.\exists x.(P\, x\, (x :: i)\, (\lambda i' \pi'.Q\, x\, o\, i'\, \pi'))\, \pi \\
[\![walked\ in]\!] &= \lambda s.s(\lambda x.\lambda i o \pi.(\mathbf{W}\, x\, \pi) \wedge (o\, i\, \pi))
\end{aligned}
$$

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.

$\lambda io\pi.\exists x.\mathbf{M}(x,\pi) \wedge \mathbf{W}(x,\pi) \wedge (o\,(x :: i)\,\pi)$

$$
\begin{aligned}
[\![man]\!] &= \lambda x.\lambda io\pi.(\mathbf{M}\,x\,\pi) \wedge (o\,i\,\pi) \\
[\![a]\!] &= \lambda P.\lambda Q.\lambda io\pi.\exists x.(P\,x\,(x :: i)\,(\lambda i'\pi'.Q\,x\,o\,i'\,\pi'))\,\pi \\
[\![walked\ in]\!] &= \lambda s.s(\lambda x.\lambda io\pi.(\mathbf{W}\,x\,\pi) \wedge (o\,i\,\pi))
\end{aligned}
$$

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.
$\lambda io\pi.\exists x.\mathbf{M}(x,\pi) \wedge \mathbf{W}(x,\pi) \wedge (o\,(x :: i)\,\pi)$

(π_2) Then he coughed.

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.
$\lambda io\pi.\exists x.\mathbf{M}(x,\pi) \wedge \mathbf{W}(x,\pi) \wedge (o\,(x :: i)\,\pi)$

(π_2) Then he coughed.

$$[\![coughed]\!] = \lambda s.s(\lambda x.\lambda io\pi.(\mathbf{C}\,x\,\pi) \wedge (o\,i\,\pi))$$
$$[\![he]\!] = \lambda P.\lambda io\pi.P\,(\mathtt{sel}_E\,i)\,i\,o\,\pi$$
$$[\![then]\!] = \lambda s.\lambda io\pi_2.\exists\pi.s\,i\,(\lambda i'\pi'.Nar(\mathtt{sel}_L(i),\pi_2,\pi) \wedge (o\,(\pi :: i')\,\pi'))\,\pi_2$$

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.
$\lambda io\pi.\exists x.\mathbf{M}(x,\pi) \wedge \mathbf{W}(x,\pi) \wedge (o\,(x :: i)\,\pi)$

(π_2) Then he coughed.
$\lambda io\pi_2.\exists \pi.\mathbf{C}(\mathtt{sel}_E(i),\pi_2)$
$\wedge Nar(\mathtt{sel}_L(i),\pi_2,\pi) \wedge o\,(\pi + i)\,\pi_2)$

$$[\![coughed]\!] = \lambda s.s(\lambda x.\lambda io\pi.(\mathbf{C}\,x\,\pi) \wedge (o\,i\,\pi))$$
$$[\![he]\!] = \lambda P.\lambda io\pi.P\,(\mathtt{sel}_E\,i)\,i\,o\,\pi$$
$$[\![then]\!] = \lambda s.\lambda io\pi_2.\exists \pi.s\,i\,(\lambda i'\pi'.Nar(\mathtt{sel}_L(i),\pi_2,\pi) \wedge (o\,(\pi :: i')\,\pi'))\,\pi_2$$

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.
$\lambda i o \pi . \exists x . \mathbf{M}(x, \pi) \wedge \mathbf{W}(x, \pi) \wedge (o\,(x :: i)\,\pi)$

(π_2) Then he coughed.
$\lambda i o \pi_2 . \exists \pi . \mathbf{C}(\mathtt{sel}_E(i), \pi_2)$
$\wedge Nar(\mathtt{sel}_L(i), \pi_2, \pi) \wedge o\,(\pi + i)\,\pi_2)$

$$[\![S_1 . S_2]\!] = \lambda i o \pi'' . \exists \pi_1 . [\![S_1]\!]\,(\pi_1 :: i)\,(\lambda i' \pi' . \exists \pi_2 . [\![S_2]\!]\,(\pi_2 :: i')\,o\,\pi_2)\,\pi_1$$

SDRT in CS: Lexicalized Discourse Relations

Example

(π_1) A man walked in.
$\lambda i o \pi . \exists x . \mathbf{M}(x,\pi) \wedge \mathbf{W}(x,\pi) \wedge (o\,(x :: i)\,\pi)$

(π_2) Then he coughed.
$\lambda i o \pi_2 . \exists \pi . \mathbf{C}(\mathtt{sel}_E(i), \pi_2) \wedge Nar(\mathtt{sel}_L(i), \pi_2, \pi) \wedge o\,(\pi + i)\,\pi_2)$

$$\begin{aligned}
[\![S_1 . S_2]\!] &= \lambda i o \pi'' . \exists \pi_1 . [\![S_1]\!]\,(\pi_1 :: i)\,(\lambda i' \pi' . \exists \pi_2 . [\![S_2]\!]\,(\pi_2 :: i')\,o\,\pi_2)\,\pi_1 \\
&\rightarrow_\beta \lambda i o \pi'' . \exists \pi_1 . \exists x . \mathbf{M}(x,\pi_1) \wedge \mathbf{W}(x,\pi_1) \\
&\qquad \wedge (\exists \pi_2 . \exists \pi . \mathbf{C}(\mathtt{sel}_E(\pi_2 :: (x :: (\pi_1 :: i))), \pi_2) \\
&\qquad \wedge Nar(\mathtt{sel}_L((\pi_2 :: (x :: (\pi_1 :: i))), \pi_2, \pi)) \\
&\qquad \wedge o\,(\pi + (\pi_2 :: (x :: (\pi_1 :: i))))\pi_2)
\end{aligned}$$

Structuring γ

Example

(π_1) A man walked in. (π_2) He sported a hat. (π_3) Then a woman walked in. (π_4) She wore a coat.

$$\left[\begin{array}{l} \text{Labels} = \\ \text{Available Labels} = \\ \text{Discourse entities} = \\ \\ \\ \\ \text{Content} = \\ \\ \\ \end{array}\right]$$

Structuring γ

Example

(π_1) A man walked in. (π_2) He sported a hat. (π_3) Then a woman walked in. (π_4) She wore a coat.

$$\left[\begin{array}{ll}
\text{Labels} = & \pi_1, \pi_2, \pi, \pi_3, \pi', \pi_4 \\
\text{Available Labels} = & \pi', \pi_3, \pi_4 \\
\text{Discourse entities} = & (\pi_1, x), (\pi_2, x), (\pi_2, w), (\pi_4, y), (\pi_4, z), (\pi_3, y) \\
 & \exists \pi_1.\exists x.\mathbf{M}(x, \pi_1) \wedge \mathbf{W}(x, \pi_1) \wedge \\
 & \exists \pi.\exists \pi_2.\exists h.\mathbf{S}(\mathtt{sel}_E(x :: \mathtt{nil}, \pi_1), h, \pi_2) \wedge H(h) \\
 & \wedge Background(\pi_1, \pi_2, \pi) \wedge \\
\text{Content} = & \exists \pi_3.\exists y(\mathbf{Wo}(y, \pi_3) \wedge \mathbf{W}(y, \pi_3)) \\
 & \wedge Narration(\pi, \pi_3, \pi') \wedge \\
 & \exists \pi'.\exists \pi_4.\exists c.\mathbf{Wear}(\mathtt{sel}_E(y :: x :: \mathtt{nil}, \pi_3), c, \pi_4) \\
 & \wedge \mathbf{Coat}(c, \pi_4) \wedge Background(\pi_3, \pi_4, \pi')
\end{array}\right]$$

Advantages of CS

- straightforward computation of complexity and confluence—depends crucially on $\texttt{sel}_E, \texttt{sel}_L, \texttt{sel}_\rho$. Everything else is just β reduction.
- typing of labels as part of the lexicon: $\mathrm{PROP} \bullet \mathrm{EVTY}$ or $\mathrm{FACT} \bullet \mathrm{EVTY}$ makes clear clashes of veridicality that drives attachment.

Example

Bob likes sports but Sam doesn't. Or Fred doesn't.

Advantages of CS continued

- Interactions between compositional semantics and discourse made more explicit.

Example

Bob came to the party only because he had nothing better to do.

If John goes to the mountains, he normally brings his dog. He normally brings a walking stick too.

- the syntax of appositions, left dislocated adverbials
  E.g., treatment frame adverbials without underspecification $\mathtt{sel}_L$ must select a label from the continuation.

Example

In the thirties, [liquor could not be sold in most areas. Speakeasies developed throughout the US.]

Perspectives

- Interaction with lexical semantics
- Interaction with syntax (ACG)
- Computations within the `sel` operators
- Interaction between $\texttt{sel}_L$, $\texttt{sel}_E$ and $\texttt{sel}_\rho$
- What about the duplication of the content?
- What (technical) solution to prefer? Why?
- Feedback on SDRT

N. Asher and A. Lascarides.
Logics of Conversation.
Cambridge University Press, 2003.

P. de Groote.
Towards a montegovian account of dynamics.
In *SALT 16*, pages 148–155. CLC Publications, 2006.