Discourse Structure in Dialogue

Lecture 4: Underspecification

Julian J. Schlöder

Review of the Semantics

- The Big Picture:
- SDRT logical form is built on two (very similar) languages: microstructure and macrostructure.
- Microstructure describes the logical form of clauses.
- Macrostructure describes how clauses form complex narratives.
- An SDRS assigns labels to both microstructure *and* macrostructure formulae.

Microstructure Vocabulary

Variables ($x, y, \ldots, e_1, e_2, \ldots$); Name symbols (John, Max, ...); Predicate symbols (`eat`, `overlap`, ...); logical connectives ($=, >, \Rightarrow, \neg, \Diamond$).

Microstructure Formulas (DRSs)

A DRS is a tuple $\langle U, Cond \rangle$ where U is a set of variables, and $Cond$ is a set of conditions.

- For a name N and a variable x, $N(x)$ is a condition.
- For a predicate P and variables $x_1, \ldots, x_n$, $P(x_1, \ldots, x_n)$ is a condition.
- For variables x and y, $x = y$ is a condition.
- If C_1 and C_2 are DRSs, $C_1 > C_2$, $C_1 \Rightarrow C_2$, $\neg C_1$ and $\Diamond C_1$ are conditions.

(add more as needed!)

Macrostructure Vocabulary

DRSs; discourse relation symbols (Elaboration, Narration, ...); label variables (π, λ, ...); logical connectives ($\neg$, $>$, $\Rightarrow$, $\wedge$, $\Diamond$).

Macrostructure Formulas

- Any DRS *K* is a macrostructure formula.
  (DRSs are like the atoms of the macrostructure)
- For a discourse relation *R* and label variables α, β, $R(\alpha, \beta)$ is a macrostructure formula.
- If *P* and *Q* are macrostructure formulae, then so are $P \wedge Q$, $\neg P$, $\Diamond P$, $P > Q$, $P \Rightarrow Q$.

Segmented Discourse Representation Structure

An SDRS is a triple $(\Pi, \mathcal{F}, L)$ where Π is a set of label variables, $L \in \Pi$ and $\mathcal{F}$ is a function from Π to the macrostructure formulae such that for any $\pi \in \Pi$, either:

- $\mathcal{F}(\pi) = K$ for some DRS K (microstructure).
- $\mathcal{F}(\pi)$ is a conjunction of formulas of the form $R(\alpha, \beta)$ (where $\alpha, \beta \in \Pi$).

An SDRS is well-formed, if:

- It has an outscoping-maximal label.
- Outscoping has no circles

Microstructure Evaluation (revised and improved)

Microstructure Semantics

Let $M = (W, *, I)$ be a qf commonsense entailment model. Define by simultaneous recursion for any $w \in W$:

1. $f[\![\langle U, Cons\rangle]\!]_{M,w}g$ iff $f[\langle U, Cons\rangle]_{M,w}g$ and $M, w, g \models_{micro} C$ for all $C \in Cons$.
2. $M, w, f \models_{micro} R(x_1, \cdots, x_n)$ iff $M, w, f \models R(f(x_1), \cdots, f(x_n))$.
3. $M, w, f \models_{micro} \neg K$ iff there is no g with $f[\![K]\!]_{M,w}g$.
4. $M, w, f \models_{micro} \Diamond K$ iff there is a $v \in W$ and a g with $f[\![K]\!]_{M,v}g$
5. $M, w, f \models_{micro} K_1 \Rightarrow K_2$ iff for every g with $f[\![K_1]\!]_{M,w}g$ there is a h with $g[\![K_2]\!]_{M,w}h$.

Microstructure Evaluation (revised and improved)

Microstructure Semantics

Let $M = (W, *, I)$ be a qf commonsense entailment model. Define by simultaneous recursion for any $w \in W$:

1. $f[\![\langle U, Cons\rangle]\!]_{M,w}g$ iff $f[\langle U, Cons\rangle]_{M,w}g$ and $M, w, g \models_{micro} C$ for all $C \in Cons$.
2. $M, w, f \models_{micro} R(x_1, \cdots, x_n)$ iff $M, w, f \models R(f(x_1), \cdots, f(x_n))$.
3. $M, w, f \models_{micro} \neg K$ iff there is no g with $f[\![K]\!]_{M,w}g$.
4. $M, w, f \models_{micro} \Diamond K$ iff there is a $v \in W$ and a g with $f[\![K]\!]_{M,v}g$
5. $M, w, f \models_{micro} K_1 \Rightarrow K_2$ iff for every g with $f[\![K_1]\!]_{M,w}g$ there is a h with $g[\![K_2]\!]_{M,w}h$.
6. For M, g let $N^{M,g}(K)$ be the set of all worlds v such that there is a h with $g[\![K]\!]_{M,v}h$. Then: $M, w, f \models_{micro} K_1 > K_2$ iff:
   for any $v \in *(w, N^{M,f}(K_1))$ and g such that $f[\![K]\!]_{M,v}g$, there is a h such that $g[\![K_2]\!]_{M,v}h$

Truth and Update

- Let P be a macrostructure formula, w be world worlds, f, g be variable assignments, M be a model, and $S = (\Pi, \mathcal{F}, L)$ be an SDRS such that all labels that appear in P are members of Π.
- We wish to define what it means that $f[\![P]\!]^S_{M,w}g$.
- So, say, you start with a *set* of possible worlds W and an assignment f,
  - $\rightarrow$ Typically ("null context"): W is all possible worlds, and f is empty
- and you want to *narrow* this set down to the worlds that support the information in an SDRS $S = (\Pi, \mathcal{F}, L)$
- Then you compute which world-assignment pairs are not ruled out by the content of S's root label π_0:

$$\{(w, g) \mid f[\![\mathcal{F}(\pi_0)]\!]^S_{M,w}g)\}$$

Macrostructure Evaluation (revised and improved)

I'm giving you a different (but equivalent) way of going about this (this version is better suited to study examples):

- We recursively translate a macrostructure formula P into a *microstructure* K such that update with K represents the information in P (not as hard as it sounds!).

- A bit of notation:

- For two DRSs $K_1 = \langle U_1, C_1 \rangle$, $K_2 = \langle U_2, C_2 \rangle$, define $K_1 + K_2 = \langle U_1 \cup U_2, C_1 \cup C_2 \rangle$.

Macrostructure-to-Microstructure

Given an SDRS $S = (\Pi, \mathcal{F}, L)$, translate a macro formula P to a DRS $[\![P]\!]^S$ (say, *P interpreted in the narrative structure S*).

1. If $P = K$ for a DRS K, then $[\![P]\!]^S = K$.

2a. If $P = Q_1 \wedge Q_2$, then $[\![P]\!]^S = [\![Q_1]\!]^S + [\![Q_2]\!]^S$.

2b. If $P = \neg Q$, then $[\![P]\!]^S = \boxed{\begin{array}{c} \\ \hline \neg[\![Q]\!]^S \end{array}}$

2c. If $P = \Diamond Q$, then $[\![P]\!]^S = \boxed{\begin{array}{c} \\ \hline \Diamond[\![Q]\!]^S \end{array}}$

2d. If $P = Q_1 > Q_2$, then $[\![P]\!]^S = \boxed{\begin{array}{c} \\ \hline [\![Q_1]\!]^S > [\![Q_2]\!]^S \end{array}}$.

2e. If $P = Q_1 \Rightarrow Q_2$, then $[\![P]\!]^S = \boxed{\begin{array}{c} \\ \hline [\![Q_1]\!]^S \Rightarrow [\![Q_2]\!]^S \end{array}}$.

3. If $P = R(\alpha, \beta)$ for a veridical discourse relation R, then $[\![P]\!]^S = [\![\mathcal{F}(\alpha) \wedge \mathcal{F}(\beta) \wedge Info_R(\mathcal{F}(\alpha), \mathcal{F}(\beta))]\!]^S$

   where $Info_R$ is the specific semantic contribution provided by the relation R (a meaning postulate).

Info's

- Narration is veridical and adds the information that events are reported in order:

$$\mathit{Info}_{\text{Narration}}(\pi, \mathcal{F}(\alpha), \mathcal{F}(\beta)) = \boxed{\begin{array}{l} e_\pi \\ \hline \texttt{part-of}(e_\alpha, e_\pi) \\ \texttt{part-of}(e_\beta, e_\pi) \\ \texttt{end}(e_\alpha) \approx \texttt{start}(e_\beta) \end{array}}$$

- Elaboration is veridical and adds that the second content defeasibly entails the first, but not vice versa, and that the evens overlap:

$$\mathit{Info}_{\text{Elab}}(\mathcal{F}(\alpha), \mathcal{F}(\beta)) = \mathcal{F}(\beta) > \mathcal{F}(\alpha) \wedge \neg(\mathcal{F}(\alpha) > \mathcal{F}(\beta)) \wedge \boxed{\begin{array}{l} \\ \hline \texttt{part-of}(e_\beta, e_\alpha) \end{array}}$$

Summary

Linguistic Forms

are interpreted to

SDRSs — describe narrative structure

are converted to

DRSs — describe event structure

are evaluated in

Models

(1) π_1 : John had a great lunch .

π_2 : He had soup.

π_3 : Then he had pasta.

$$\Pi = \{\pi_0, \pi_1, \pi_2, \pi_3, \lambda\}, L = \pi_3.$$

$$\mathcal{F}(\pi_0) = \mathit{Elaboration}(\pi_1, \lambda) \quad \mathcal{F}(\lambda) = \mathit{Narration}(\pi_2, \pi_3)$$

$$\mathcal{F}(\pi_1) = K_1 = \begin{array}{|l|}\hline j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \\ \hline \end{array}, \mathcal{F}(\pi_2) = K_2 = \begin{array}{|l|}\hline s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \hline \end{array}, \mathcal{F}(\pi_3) = K_3 = \begin{array}{|l|}\hline p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \hline \end{array}$$

(1) π_1 : John had a great lunch .

π_2 : He had soup.

π_3 : Then he had pasta.

$$\Pi = \{\pi_0, \pi_1, \pi_2, \pi_3, \lambda\}, L = \pi_3.$$

$$\mathcal{F}(\pi_0) = \textit{Elaboration}(\pi_1, \lambda) \quad \mathcal{F}(\lambda) = \textit{Narration}(\pi_2, \pi_3)$$

$$\mathcal{F}(\pi_1){=}K_1{=}\begin{array}{|l|}\hline j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \\ \hline \end{array}, \mathcal{F}(\pi_2){=}K_2{=}\begin{array}{|l|}\hline s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \hline \end{array}, \mathcal{F}(\pi_3){=}K_3{=}\begin{array}{|l|}\hline p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \hline \end{array}$$

$$[\![\mathcal{F}(\pi_0)]\!]^S = [\![\textit{Elaboration}(\pi_1, \lambda)]\!]^S$$

(1) π_1 : John had a great lunch .

π_2 : He had soup.

π_3 : Then he had pasta.

$$\Pi = \{\pi_0, \pi_1, \pi_2, \pi_3, \lambda\}, L = \pi_3.$$

$$\mathcal{F}(\pi_0) = \mathit{Elaboration}(\pi_1, \lambda) \quad \mathcal{F}(\lambda) = \mathit{Narration}(\pi_2, \pi_3)$$

$$\mathcal{F}(\pi_1){=}K_1{=} \boxed{\begin{array}{l} j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \end{array}}, \mathcal{F}(\pi_2){=}K_2{=} \boxed{\begin{array}{l} s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \end{array}}, \mathcal{F}(\pi_3){=}K_3{=} \boxed{\begin{array}{l} p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \end{array}}$$

$$[\![\mathcal{F}(\pi_0)]\!]^S = [\![\mathit{Elaboration}(\pi_1, \lambda)]\!]^S$$
$$= [\![\mathcal{F}(\pi_1) \wedge \mathcal{F}(\lambda) \wedge \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S$$

(1) π_1 : John had a great lunch .

π_2 : He had soup.

π_3 : Then he had pasta.

$$\Pi = \{\pi_0, \pi_1, \pi_2, \pi_3, \lambda\}, L = \pi_3.$$

$$\mathcal{F}(\pi_0) = \mathit{Elaboration}(\pi_1, \lambda) \quad \mathcal{F}(\lambda) = \mathit{Narration}(\pi_2, \pi_3)$$

$$\mathcal{F}(\pi_1) = K_1 = \boxed{\begin{array}{l} j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \mathtt{lunch}(l) \\ \mathtt{great}(l) \\ \mathtt{had}(e_{\pi_1}, j, l) \end{array}}, \mathcal{F}(\pi_2) = K_2 = \boxed{\begin{array}{l} s, e_{\pi_2} \\ \hline \mathtt{soup}(s) \\ \mathtt{had}(e_{\pi_2}, j, s) \end{array}}, \mathcal{F}(\pi_3) = K_3 = \boxed{\begin{array}{l} p, e_{\pi_3} \\ \hline \mathtt{pasta}(p) \\ \mathtt{had}(e_{\pi_3}, j, p) \end{array}}$$

$$\begin{aligned} & [\![\mathcal{F}(\pi_0)]\!]^S = [\![\mathit{Elaboration}(\pi_1, \lambda)]\!]^S \\ & = [\![\mathcal{F}(\pi_1) \wedge \mathcal{F}(\lambda) \wedge \mathit{Info}_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\ & = [\![\mathcal{F}(\pi_1)]\!]^S + [\![\mathcal{F}(\lambda)]\!]^S + [\![\mathit{Info}_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \end{aligned}$$

(1) π_1 : John had a great lunch .
π_2 : He had soup.
π_3 : Then he had pasta.

$$\Pi = \{\pi_0, \pi_1, \pi_2, \pi_3, \lambda\}, L = \pi_3.$$
$$\mathcal{F}(\pi_0) = \mathit{Elaboration}(\pi_1, \lambda) \quad \mathcal{F}(\lambda) = \mathit{Narration}(\pi_2, \pi_3)$$

$$\mathcal{F}(\pi_1) = K_1 = \begin{array}{|l|}\hline j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \\ \hline \end{array}, \mathcal{F}(\pi_2) = K_2 = \begin{array}{|l|}\hline s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \hline \end{array}, \mathcal{F}(\pi_3) = K_3 = \begin{array}{|l|}\hline p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \hline \end{array}$$

$$\begin{aligned}
&[\![\mathcal{F}(\pi_0)]\!]^S = [\![\mathit{Elaboration}(\pi_1, \lambda)]\!]^S \\
&= [\![\mathcal{F}(\pi_1) \wedge \mathcal{F}(\lambda) \wedge \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![\mathcal{F}(\pi_1)]\!]^S + [\![\mathcal{F}(\lambda)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\lambda)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S
\end{aligned}$$

(1) π_1 : John had a great lunch .

π_2 : He had soup.

π_3 : Then he had pasta.

$$\Pi = \{\pi_0, \pi_1, \pi_2, \pi_3, \lambda\},\ L = \pi_3.$$
$$\mathcal{F}(\pi_0) = \mathit{Elaboration}(\pi_1, \lambda) \quad \mathcal{F}(\lambda) = \mathit{Narration}(\pi_2, \pi_3)$$

$$\mathcal{F}(\pi_1) = K_1 = \begin{array}{|l|} \hline j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \\ \hline \end{array},\ \mathcal{F}(\pi_2) = K_2 = \begin{array}{|l|} \hline s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \hline \end{array},\ \mathcal{F}(\pi_3) = K_3 = \begin{array}{|l|} \hline p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \hline \end{array}$$

$$\begin{aligned} & [\![\mathcal{F}(\pi_0)]\!]^S = [\![\mathit{Elaboration}(\pi_1, \lambda)]\!]^S \\ & = [\![\mathcal{F}(\pi_1) \wedge \mathcal{F}(\lambda) \wedge \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\ & = [\![\mathcal{F}(\pi_1)]\!]^S + [\![\mathcal{F}(\lambda)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\ & = [\![K_1]\!]^S + [\![\mathcal{F}(\lambda)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\ & = [\![K_1]\!]^S + [\![\mathit{Narration}(\pi_2, \pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \end{aligned}$$

(1) π_1 : John had a great lunch .
π_2 : He had soup.
π_3 : Then he had pasta.

$$\Pi = \{\pi_0, \pi_1, \pi_2, \pi_3, \lambda\}, L = \pi_3.$$
$$\mathcal{F}(\pi_0) = \mathit{Elaboration}(\pi_1, \lambda) \quad \mathcal{F}(\lambda) = \mathit{Narration}(\pi_2, \pi_3)$$

$$\mathcal{F}(\pi_1){=}K_1{=} \boxed{\begin{array}{l} j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \end{array}}, \mathcal{F}(\pi_2){=}K_2{=} \boxed{\begin{array}{l} s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \end{array}}, \mathcal{F}(\pi_3){=}K_3{=} \boxed{\begin{array}{l} p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \end{array}}$$

$$\begin{aligned} &[\![\mathcal{F}(\pi_0)]\!]^S = [\![\mathit{Elaboration}(\pi_1, \lambda)]\!]^S \\ &= [\![\mathcal{F}(\pi_1) \wedge \mathcal{F}(\lambda) \wedge \mathit{Info}_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\ &= [\![\mathcal{F}(\pi_1)]\!]^S + [\![\mathcal{F}(\lambda)]\!]^S + [\![\mathit{Info}_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\ &= [\![K_1]\!]^S + [\![\mathcal{F}(\lambda)]\!]^S + [\![\mathit{Info}_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\ &= [\![K_1]\!]^S + [\![\mathit{Narration}(\pi_2, \pi_3)]\!]^S + [\![\mathit{Info}_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \end{aligned}$$

need new slide

$$[\![\mathcal{F}(\pi_0)]\!]^S = [\![K_1]\!]^S + [\![\mathit{Narration}(\pi_2, \pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S$$

$$
\llbracket \mathcal{F}(\pi_0) \rrbracket^S = \llbracket K_1 \rrbracket^S + \llbracket \mathit{Narration}(\pi_2, \pi_3) \rrbracket^S + \llbracket \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda)) \rrbracket^S
$$
$$
= \llbracket K_1 \rrbracket^S + \llbracket \mathcal{F}(\pi_2) \wedge \mathcal{F}(\pi_3) \wedge \mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2)) \rrbracket^S + \llbracket \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda)) \rrbracket^S
$$

$$\llbracket \mathcal{F}(\pi_0) \rrbracket^S = \llbracket K_1 \rrbracket^S + \llbracket \mathit{Narration}(\pi_2, \pi_3) \rrbracket^S + \llbracket \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda)) \rrbracket^S$$
$$= \llbracket K_1 \rrbracket^S + \llbracket \mathcal{F}(\pi_2) \wedge \mathcal{F}(\pi_3) \wedge \mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2)) \rrbracket^S + \llbracket \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda)) \rrbracket^S$$
$$= \llbracket K_1 \rrbracket^S + \llbracket \mathcal{F}(\pi_2) \rrbracket^S + \llbracket \mathcal{F}(\pi_3) \rrbracket^S + \llbracket \mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2)) \rrbracket^S + \llbracket \mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda)) \rrbracket^S$$

$$
\begin{aligned}
&[\![\mathcal{F}(\pi_0)]\!]^S = [\![K_1]\!]^S + [\![\mathit{Narration}(\pi_2, \pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2) \wedge \mathcal{F}(\pi_3) \wedge \mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2)]\!]^S + [\![\mathcal{F}(\pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![K_2]\!]^S + [\![K_3]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S
\end{aligned}
$$

$$
\begin{aligned}
&[\![\mathcal{F}(\pi_0)]\!]^S = [\![K_1]\!]^S + [\![\mathit{Narration}(\pi_2, \pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2) \wedge \mathcal{F}(\pi_3) \wedge \mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2)]\!]^S + [\![\mathcal{F}(\pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![K_2]\!]^S + [\![K_3]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S
\end{aligned}
$$

$$
= [\![K_1]\!]^S + [\![K_2]\!]^S + [\![K_3]\!]^S + \boxed{\begin{array}{l} e_\lambda \\ \hline \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S
$$

$$
\begin{aligned}
[\![\mathcal{F}(\pi_0)]\!]^S &= [\![K_1]\!]^S + [\![\mathit{Narration}(\pi_2, \pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2) \wedge \mathcal{F}(\pi_3) \wedge \mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2)]\!]^S + [\![\mathcal{F}(\pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S \\
&= [\![K_1]\!]^S + [\![K_2]\!]^S + [\![K_3]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1), \mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S
\end{aligned}
$$

$$
= [\![K_1]\!]^S + [\![K_2]\!]^S + [\![K_3]\!]^S + \boxed{\begin{array}{l} e_\lambda \\ \hline \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S
$$

$$
= [\![K_1]\!]^S + \boxed{\begin{array}{l} s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \end{array}} + \boxed{\begin{array}{l} p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \end{array}} + \boxed{\begin{array}{l} e_\lambda \\ \hline \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S
$$

$$
\begin{aligned}
&[\![\mathcal{F}(\pi_0)]\!]^S = [\![K_1]\!]^S + [\![\mathit{Narration}(\pi_2,\pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1),\mathcal{F}(\lambda))]\!]^S\\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2) \wedge \mathcal{F}(\pi_3) \wedge \mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1),\mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1),\mathcal{F}(\lambda))]\!]^S\\
&= [\![K_1]\!]^S + [\![\mathcal{F}(\pi_2)]\!]^S + [\![\mathcal{F}(\pi_3)]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1),\mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1),\mathcal{F}(\lambda))]\!]^S\\
&= [\![K_1]\!]^S + [\![K_2]\!]^S + [\![K_3]\!]^S + [\![\mathit{Info}_{\mathit{Narr}}(\mathcal{F}(\pi_1),\mathcal{F}(\pi_2))]\!]^S + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1),\mathcal{F}(\lambda))]\!]^S
\end{aligned}
$$

$$
= [\![K_1]\!]^S + [\![K_2]\!]^S + [\![K_3]\!]^S + \boxed{\begin{array}{l} e_\lambda \\ \hline \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1),\mathcal{F}(\lambda))]\!]^S
$$

$$
= [\![K_1]\!]^S + \boxed{\begin{array}{l} s, e_{\pi_2} \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \end{array}} + \boxed{\begin{array}{l} p, e_{\pi_3} \\ \hline \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \end{array}} + \boxed{\begin{array}{l} e_\lambda \\ \hline \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1),\mathcal{F}(\lambda))]\!]^S
$$

continued next slide

$$= [\![K_1]\!]^S + \boxed{\begin{array}{l} s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![Info_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S$$

$$= [\![K_1]\!]^S + \boxed{\begin{array}{l} s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![Info_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S$$

$$= [\![\mathcal{F}(\lambda) > K_1 \wedge \neg(K_1 > \mathcal{F}(\lambda)) \wedge \boxed{\begin{array}{l} \\ \hline \texttt{part-of}(e_\lambda, e_{\pi_1} \end{array}}]\!]^S$$

$$= [\![K_1]\!]^S + \boxed{\begin{array}{l} s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \end{array}} + [\![\mathit{Info}_{\mathit{Elab}}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda))]\!]^S$$

$$= [\![\mathcal{F}(\lambda) > K_1 \wedge \neg(K_1 > \mathcal{F}(\lambda)) \wedge \boxed{\begin{array}{l} \\ \hline \texttt{part-of}(e_\lambda, e_{\pi_1} \end{array}}]\!]^S$$

$$= \boxed{\begin{array}{l} \\ \hline [\![\mathcal{F}(\lambda)]\!]^S > K_1 \end{array}} + \boxed{\begin{array}{l} \\ \hline \neg \boxed{\begin{array}{l} \\ \hline K_1 > [\![\mathcal{F}(\lambda)]\!]^S \end{array}} \end{array}} + \boxed{\begin{array}{l} \\ \hline \texttt{part-of}(e_\lambda, e_{\pi_1}) \end{array}}$$

$$= \llbracket K_1 \rrbracket^S + \boxed{\begin{array}{l} s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \mathtt{soup}(s) \\ \mathtt{had}(e_{\pi_2}, j, s) \\ \mathtt{pasta}(p) \\ \mathtt{had}(e_{\pi_3}, j, p) \\ \mathtt{part\text{-}of}(e_{\pi_2}, e_\lambda) \\ \mathtt{part\text{-}of}(e_{\pi_3}, e_\lambda) \\ \mathtt{end}(e_{\pi_2}) \approx \mathtt{start}(e_{\pi_3}) \end{array}} + \llbracket Info_{Elab}(\mathcal{F}(\pi_1), \mathcal{F}(\lambda)) \rrbracket^S$$

$$= \llbracket \mathcal{F}(\lambda) > K_1 \wedge \neg(K_1 > \mathcal{F}(\lambda)) \wedge \boxed{\begin{array}{l} \\ \hline \mathtt{part\text{-}of}(e_\lambda, e_{\pi_1} \end{array}} \rrbracket^S$$

$$= \boxed{\begin{array}{l} \\ \hline \llbracket \mathcal{F}(\lambda) \rrbracket^S > K_1 \end{array}} + \boxed{\begin{array}{l} \\ \hline \neg \boxed{\begin{array}{l} \\ \hline K_1 > \llbracket \mathcal{F}(\lambda) \rrbracket^S \end{array}} \end{array}} + \boxed{\begin{array}{l} \\ \hline \mathtt{part\text{-}of}(e_\lambda, e_{\pi_1}) \end{array}}$$

$$= \boxed{\begin{array}{l} j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \mathtt{lunch}(l) \\ \mathtt{great}(l) \\ \mathtt{had}(e_{\pi_1}, j, l) \end{array}} + \underbrace{\boxed{\begin{array}{l} s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \mathtt{soup}(s) \\ \mathtt{had}(e_{\pi_2}, j, s) \\ \mathtt{pasta}(p) \\ \mathtt{had}(e_{\pi_3}, j, p) \\ \mathtt{part\text{-}of}(e_{\pi_2}, e_\lambda) \\ \mathtt{part\text{-}of}(e_{\pi_3}, e_\lambda) \\ \mathtt{end}(e_{\pi_2}) \approx \mathtt{start}(e_{\pi_3}) \end{array}}}_{= \llbracket \mathcal{F}(\lambda) \rrbracket^S} + \boxed{\begin{array}{l} \\ \hline \llbracket \mathcal{F}(\lambda) \rrbracket^S > K_1 \\ \neg \boxed{\begin{array}{l} \\ \hline K_1 > \llbracket \mathcal{F}(\lambda) \rrbracket^S \end{array}} \\ \mathtt{part\text{-}of}(e_\lambda, e_{\pi_1}) \end{array}}$$

$$
\begin{array}{|l|}\hline j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \\ \hline \end{array}
+
\begin{array}{|l|}\hline s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \\ \hline \end{array}
+
\begin{array}{|l|}\hline \\ \hline
\begin{array}{|l|}\hline s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \\ \hline \end{array}
>
\begin{array}{|l|}\hline j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \\ \hline \end{array}
\\
\neg
\begin{array}{|l|}\hline \\ \hline
\begin{array}{|l|}\hline j, l, e_{\pi_1} \\ \hline \text{John}(j) \\ \texttt{lunch}(l) \\ \texttt{great}(l) \\ \texttt{had}(e_{\pi_1}, j, l) \\ \hline \end{array}
>
\begin{array}{|l|}\hline s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\ \hline \texttt{soup}(s) \\ \texttt{had}(e_{\pi_2}, j, s) \\ \texttt{pasta}(p) \\ \texttt{had}(e_{\pi_3}, j, p) \\ \texttt{part-of}(e_{\pi_2}, e_\lambda) \\ \texttt{part-of}(e_{\pi_3}, e_\lambda) \\ \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \\ \hline \end{array}
\\ \hline \end{array}
\\
\texttt{part-of}(e_\lambda, e_{\pi_1})
\\ \hline \end{array}
$$

$$
\begin{array}{|l|}
\hline
j, l, e_{\pi_1}, s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\
\hline
\begin{array}{ll}
\text{John}(j) & \texttt{pasta}(p) \\
\texttt{lunch}(l) & \texttt{had}(e_{\pi_3}, j, p) \\
\texttt{great}(l)\ \texttt{had}(e_{\pi_1}, j, l) & \texttt{part-of}(e_{\pi_2}, e_\lambda) \\
\texttt{soup}(s) & \texttt{part-of}(e_{\pi_3}, e_\lambda) \\
\texttt{had}(e_{\pi_2}, j, s) & \texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3})
\end{array} \\
\\
\begin{array}{|c|}
\hline
\\
\hline
\begin{array}{|l|}
\hline
s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\
\hline
\texttt{soup}(s) \\
\texttt{had}(e_{\pi_2}, j, s) \\
\texttt{pasta}(p) \\
\texttt{had}(e_{\pi_3}, j, p) \\
\texttt{part-of}(e_{\pi_2}, e_\lambda) \\
\texttt{part-of}(e_{\pi_3}, e_\lambda) \\
\texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \\
\hline
\end{array}
>
\begin{array}{|l|}
\hline
j, l, e_{\pi_1} \\
\hline
\text{John}(j) \\
\texttt{lunch}(l) \\
\texttt{great}(l) \\
\texttt{had}(e_{\pi_1}, j, l) \\
\hline
\end{array} \\
\hline
\end{array} \\
\\
\neg
\begin{array}{|c|}
\hline
\\
\hline
\begin{array}{|l|}
\hline
j, l, e_{\pi_1} \\
\hline
\text{John}(j) \\
\texttt{lunch}(l) \\
\texttt{great}(l) \\
\texttt{had}(e_{\pi_1}, j, l) \\
\hline
\end{array}
>
\begin{array}{|l|}
\hline
s, e_{\pi_2}, p, e_{\pi_3}, e_\lambda \\
\hline
\texttt{soup}(s) \\
\texttt{had}(e_{\pi_2}, j, s) \\
\texttt{pasta}(p) \\
\texttt{had}(e_{\pi_3}, j, p) \\
\texttt{part-of}(e_{\pi_2}, e_\lambda) \\
\texttt{part-of}(e_{\pi_3}, e_\lambda) \\
\texttt{end}(e_{\pi_2}) \approx \texttt{start}(e_{\pi_3}) \\
\hline
\end{array} \\
\hline
\end{array} \\
\\
\texttt{part-of}(e_\lambda, e_{\pi_1}) \\
\hline
\end{array}
$$

Summary

Linguistic Forms

are interpreted to

SDRSs — describe narrative structure

are converted to

DRSs — describe event structure

are evaluated in

Models

Summary

Linguistic Forms

are interpreted to

SDRSs — describe narrative structure

are converted to

DRSs — describe event structure

are evaluated in

Models

Summary

Linguistic Forms

are interpreted to

Underspecified Logical Forms — partially describe content

are specified to

SDRSs — describe narrative structure

are converted to

DRSs — describe event structure

are evaluated in

Models

Review of the Semantics

Constructing Logical Form

Review of the Semantics

Constructing Logical Form

Underspecified Logical Form

- The idea is this: we construct a language for incomplete descriptions of SDRSs.
- So we need a language for “underspecified logical form” (ULF).
- We need a relation that says “this SDRS is described by this ULF”.

ULF Language: atoms and variables

- So what are the bits and pieces of an SDRS?

- DRSs

  $\rightarrow$ Any DRS K is an "atom" (or, constant symbol).
  (you can underspecify these too, but I won't)

- Labels

  $\rightarrow$ Take variable symbols for labels $l_1, l_2, \ldots$

- Discourse relations

  $\rightarrow$ Take a constant symbol D_R for each discourse relation R
  $\rightarrow$ Plus corresponding variable symbols $D_1, D_2, \ldots$

ULF Language: Structure

- We underspecify:
- What the contents are.
- Which contents are connected.
- How they are connected.

- Take two predicate symbols to describe assignment:

  $\rightarrow$ $labels(l, K)$
  $\rightarrow$ $relates(l_1, l_2, l_3, D)$

- And three to describe structure:

  $\rightarrow$ $outscopes(l_1, l_2)$
  $\rightarrow$ $accessible(l_1, l_2)$
  $\rightarrow$ $last(l_1)$

ULF Language: Anaphor

- Anaphora are a type of underspecification.
- So take a constant symbol v_x for each DRT-variable x (do this for every type of variable).
- And add a predicate symbol:

  $\rightarrow$ $\mathit{anaphor}(l, v)$

ULF Language: Anaphor

- Anaphora are a type of underspecification.
- So take a constant symbol v_x for each DRT-variable x (do this for every type of variable).
- And add a predicate symbol:

  $\rightarrow$ $anaphor(l, v)$

- (If you extend the language to partially describe microstructure, you can write anaphora as $x =?$ to indicate something like "x is not in the universe of K".)

Examples

– ULFs are constructed from surface form.

(2) There is a woman.

$$labels(l_1, \boxed{\begin{array}{l} \text{x} \\ \hline \texttt{woman}(\text{x}) \end{array}})$$

(3) She runs.

$$labels(l_2, \boxed{\begin{array}{l} \text{e,y} \\ \hline \texttt{run}(\text{e,y}) \end{array}}) \wedge anaphor(l_2, v_y)$$

Two Sentence Example

(4) There is a woman. She runs.

$$
\begin{aligned}
& \textit{labels}(l_1, \boxed{\begin{array}{l} \text{x} \\ \hline \texttt{woman}(\text{x}) \end{array}}) \\
\wedge\ & \textit{labels}(l_2, \boxed{\begin{array}{l} \text{e,y} \\ \hline \texttt{run}(\text{e,y}) \end{array}}) \wedge \textit{anaphor}(l_2, v_y) \\
\wedge\ & \textit{relates}(l_0, l_1, l_2, D) \\
\wedge\ & \textit{last}(l_2)
\end{aligned}
$$

ULF Language: Cue Phrases

- Add an (empirically sourced) vocabulary of linguistic cues to this language.

- therefore $\rightsquigarrow$ *therefore*(I)
- and then $\rightsquigarrow$ *and-then*(I)
- I hereby command $\rightsquigarrow$ *command*(I)
- I hereby assert $\rightsquigarrow$ *inform*(I)

- Including grammatical features:

- *declarative*(I)
- *interrogative*(I)
- *imperative*(I)

- Plus tense, aspect... — **anything useful from the grammar!**

From ULF to SDRS

- The underspecified language has the formulas we seen so far, closed under the logical constants $=$, $\neg$, $\vee$ and $\wedge$.
- Call a formulae in this language an ULF (underspecified logical form).

From ULF to SDRS

- The underspecified language has the formulas we seen so far, closed under the logical constants $=$, $\neg$, $\vee$ and $\wedge$.
- Call a formulae in this language an ULF (underspecified logical form).
- Now, this is conceptually a bit weird, but not hard:
- We want to define a turnstile $\models$ such that for an SDRS *S* and an ULF $\mathcal{K}$, *S* $\models \mathcal{K}$ iff all descriptions from $\mathcal{K}$ are realised in *S*.

From ULF to SDRS

- The underspecified language has the formulas we seen so far, closed under the logical constants $=$, $\neg$, $\vee$ and $\wedge$.
- Call a formulae in this language an ULF (underspecified logical form).
- Now, this is conceptually a bit weird, but not hard:
- We want to define a turnstile $\models$ such that for an SDRS S and an ULF $\mathcal{K}$, $S \models \mathcal{K}$ iff all descriptions from $\mathcal{K}$ are realised in S.
- A bit of notation: for DRSs $K_1 = \langle U_1, C_1 \rangle$, $K_2 = \langle U_2, C_2 \rangle$ say $K_1 \subseteq K_2$ iff $U_1 \subseteq U_2$ and $C_1 \subseteq C_2$.

Assignment Function

- Let $S = (\Pi, \mathcal{F}, L)$ be an SDRS and A be a function s.t.:
  - $\rightarrow$ for each variable l_i, $A(l_i) \in \Pi$
  - $\rightarrow$ for each variable D_i, $A(D_i)$ is some discourse relation.
  - $\rightarrow$ $A(D_R) = R$ for all discourse relations R
  - $\rightarrow$ $A(v_x) = x$ for all and DRT-variables x.
- (i.e. the variables are implicitly existentially quantified)

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models \mathit{last}(l_1)$ iff $A(l_1) = L$.

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models \mathit{last}(l_1)$ iff $A(l_1) = L$.
- $S, A \models \mathit{labels}(l, K)$ iff $K \subseteq \mathcal{F}(A(l))$ and $\mathcal{F}(A(l))$ does not use relation symbols not in K.

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models \mathit{last}(l_1)$ iff $A(l_1) = L$.
- $S, A \models \mathit{labels}(l, K)$ iff $K \subseteq \mathcal{F}(A(l))$ and $\mathcal{F}(A(l))$ does not use relation symbols not in K.
- $S, A \models \mathit{relates}(l_1, l_2, l_3, D)$ iff $A(D)(A(l_2), A(l_3))$ is a conjunct of $\mathcal{F}(A(l_1))$.

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models \textit{last}(l_1)$ iff $A(l_1) = L$.
- $S, A \models \textit{labels}(l, K)$ iff $K \subseteq \mathcal{F}(A(l))$ and $\mathcal{F}(A(l))$ does not use relation symbols not in K.
- $S, A \models \textit{relates}(l_1, l_2, l_3, D)$ iff $A(D)(A(l_2), A(l_3))$ is a conjunct of $\mathcal{F}(A(l_1))$.
- $S, A \models \textit{outscopes}(l_1, l_2)$ iff $A(l_2)$ outscopes (in S) $A(l_1)$.

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models \mathit{last}(l_1)$ iff $A(l_1) = L$.
- $S, A \models \mathit{labels}(l, K)$ iff $K \subseteq \mathcal{F}(A(l))$ and $\mathcal{F}(A(l))$ does not use relation symbols not in K.
- $S, A \models \mathit{relates}(l_1, l_2, l_3, D)$ iff $A(D)(A(l_2), A(l_3))$ is a conjunct of $\mathcal{F}(A(l_1))$.
- $S, A \models \mathit{outscopes}(l_1, l_2)$ iff $A(l_2)$ outscopes (in S) $A(l_1)$.
- $S, A \models \mathit{accessible}(l_1, l_2)$ iff $A(l_1)$ is accessible (in S) from $A(l_2)$.

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models \mathit{last}(l_1)$ iff $A(l_1) = L$.
- $S, A \models \mathit{labels}(l, K)$ iff $K \subseteq \mathcal{F}(A(l))$ and $\mathcal{F}(A(l))$ does not use relation symbols not in K.
- $S, A \models \mathit{relates}(l_1, l_2, l_3, D)$ iff $A(D)(A(l_2), A(l_3))$ is a conjunct of $\mathcal{F}(A(l_1))$.
- $S, A \models \mathit{outscopes}(l_1, l_2)$ iff $A(l_2)$ outscopes (in S) $A(l_1)$.
- $S, A \models \mathit{accessible}(l_1, l_2)$ iff $A(l_1)$ is accessible (in S) from $A(l_2)$.
- $S, A \models \mathit{anaphor}(l, v)$ iff there is a DRT variable z introduced in some segment $\lambda \in \Pi$ (of S) such that

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models last(l_1)$ iff $A(l_1) = L$.
- $S, A \models labels(l, K)$ iff $K \subseteq \mathcal{F}(A(l))$ and $\mathcal{F}(A(l))$ does not use relation symbols not in K.
- $S, A \models relates(l_1, l_2, l_3, D)$ iff $A(D)(A(l_2), A(l_3))$ is a conjunct of $\mathcal{F}(A(l_1))$.
- $S, A \models outscopes(l_1, l_2)$ iff $A(l_2)$ outscopes (in S) $A(l_1)$.
- $S, A \models accessible(l_1, l_2)$ iff $A(l_1)$ is accessible (in S) from $A(l_2)$.
- $S, A \models anaphor(l, v)$ iff there is a DRT variable z introduced in some segment $\lambda \in \Pi$ (of S) such that
  i. there is a relation R and labels α and β with $\mathcal{F}(\alpha) = R(\beta, A(l))$;
  ii. λ is accessible to β; and
  iii. $\mathcal{F}(A(l))$ has a conjunct $A(v) = z$.

Satisfaction

- $S, A \models x = y$ iff $A(x) = A(y)$ (for any variables or constants x, y)
- $S, A \models last(l_1)$ iff $A(l_1) = L$.
- $S, A \models labels(l, K)$ iff $K \subseteq \mathcal{F}(A(l))$ and $\mathcal{F}(A(l))$ does not use relation symbols not in K.
- $S, A \models relates(l_1, l_2, l_3, D)$ iff $A(D)(A(l_2), A(l_3))$ is a conjunct of $\mathcal{F}(A(l_1))$.
- $S, A \models outscopes(l_1, l_2)$ iff $A(l_2)$ outscopes (in S) $A(l_1)$.
- $S, A \models accessible(l_1, l_2)$ iff $A(l_1)$ is accessible (in S) from $A(l_2)$.
- $S, A \models anaphor(l, v)$ iff there is a DRT variable z introduced in some segment $\lambda \in \Pi$ (of S) such that
  i. there is a relation R and labels α and β with $\mathcal{F}(\alpha) = R(\beta, A(l))$;
  ii. λ is accessible to β; and
  iii. $\mathcal{F}(A(l))$ has a conjunct $A(v) = z$.
- If $cue(l)$ is a linguistic cue predicate, $S, A \models cue(l)$ always.

Two Sentence Example

(5) There is a woman. She runs.

$$A(l_0) = \pi_0, A(l_1) = \pi_1, A(l_2) = \pi_2,$$
$$A(D) = \mathit{Elaboration}$$

$$\Pi = \{\pi_0, \pi_1, \pi_2\}, L = \pi_2$$

$$\mathcal{F}(\pi_1) = \begin{array}{|l|}\hline \text{x} \\ \hline \texttt{woman}(\text{x}) \\ \hline \end{array}$$

$$\mathcal{F}(\pi_2) = \begin{array}{|l|}\hline \text{e,y} \\ \hline \texttt{run}(\text{e,y}) \\ y = x \\ \hline \end{array}$$

$$\mathcal{F}(\pi_0) = \mathit{Elaboration}(\pi_1, \pi_2)$$

$$\models$$

$$\mathit{labels}(l_1, \begin{array}{|l|}\hline \text{x} \\ \hline \texttt{woman}(\text{x}) \\ \hline \end{array})$$
$$\wedge\ \mathit{labels}(l_2, \begin{array}{|l|}\hline \text{e,y} \\ \hline \texttt{run}(\text{e,y}) \\ \hline \end{array})$$
$$\wedge\ \mathit{anaphor}(l_2, v_y)$$
$$\wedge\ \mathit{relates}(l_0, l_1, l_2, D)$$
$$\wedge\ \mathit{last}(l_2)$$

Two Sentence Example

(5) There is a woman. She runs.

$$A(l_0) = \pi_0, A(l_1) = \pi_1, A(l_2) = \pi_2,$$
$$A(D) = QAP$$
$$\Pi = \{\pi_0, \pi_1, \pi_2\}, L = \pi_2$$
$$\mathcal{F}(\pi_1) = \begin{array}{|l|} \hline x \\ \hline \texttt{woman}(x) \\ \hline \end{array}$$
$$\mathcal{F}(\pi_2) = \begin{array}{|l|} \hline e,y \\ \hline \texttt{run}(e,y) \\ y = x \\ \hline \end{array}$$
$$\mathcal{F}(\pi_0) = QAP(\pi_1, \pi_2)$$

$$\models \quad labels(l_1, \begin{array}{|l|} \hline x \\ \hline \texttt{woman}(x) \\ \hline \end{array})$$
$$\wedge\ labels(l_2, \begin{array}{|l|} \hline e,y \\ \hline \texttt{run}(e,y) \\ \hline \end{array})$$
$$\wedge\ anaphor(l_2, v_y)$$
$$\wedge\ relates(l_0, l_1, l_2, D)$$
$$\wedge\ last(l_2)$$

Linguistic Form to Narrative Structure

- So, given the linguistic form of a discourse, we:
  - → Compute for every *clause* the corresponding DRS K (by the DRT construction algo), except that we don't resolve anaphora here.
  - → Pick an unused label variable l_1 and add $\mathit{labels}(l_1, K)$.
  - → (If there is an ambiguity, you can also add $\mathit{labels}(l_1, K) \vee \mathit{labels}(l_1, K')$).
  - → For every anaphor x in K add $\mathit{anaphor}(l_1, v_x)$.
  - → Add appropriate predicates on l for cue phrases and linguistic features (aspect etc.).
  - → For every clause except the very first one, pick another two unused label variables l_0, l_2 and add $\mathit{relates}(l_0, l_2, l_1, D)$ (i.e. l_1 attaches somewhere)
- Call the conjunction of all these $\mathcal{K}$.

Linguistic Form to Narrative Structure

- So, given the linguistic form of a discourse, we:
  - → Compute for every *clause* the corresponding DRS K (by the DRT construction algo), except that we don't resolve anaphora here.
  - → Pick an unused label variable l_1 and add $\mathit{labels}(l_1, K)$.
  - → (If there is an ambiguity, you can also add $\mathit{labels}(l_1, K) \vee \mathit{labels}(l_1, K')$).
  - → For every anaphor x in K add $\mathit{anaphor}(l_1, v_x)$.
  - → Add appropriate predicates on l for cue phrases and linguistic features (aspect etc.).
  - → For every clause except the very first one, pick another two unused label variables l_0, l_2 and add $\mathit{relates}(l_0, l_2, l_1, D)$ (i.e. l_1 attaches somewhere)
- Call the conjunction of all these $\mathcal{K}$.

Not good enough!