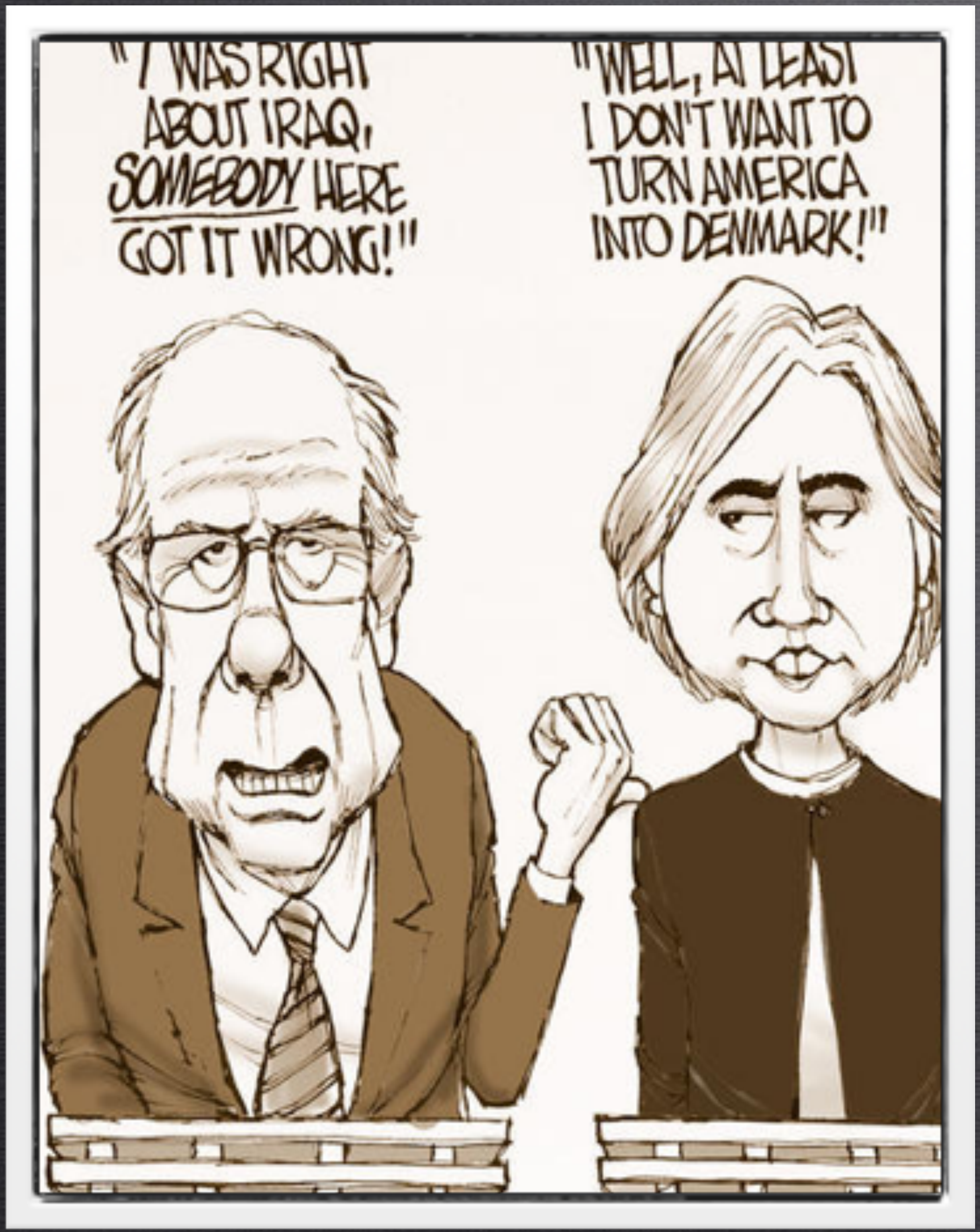


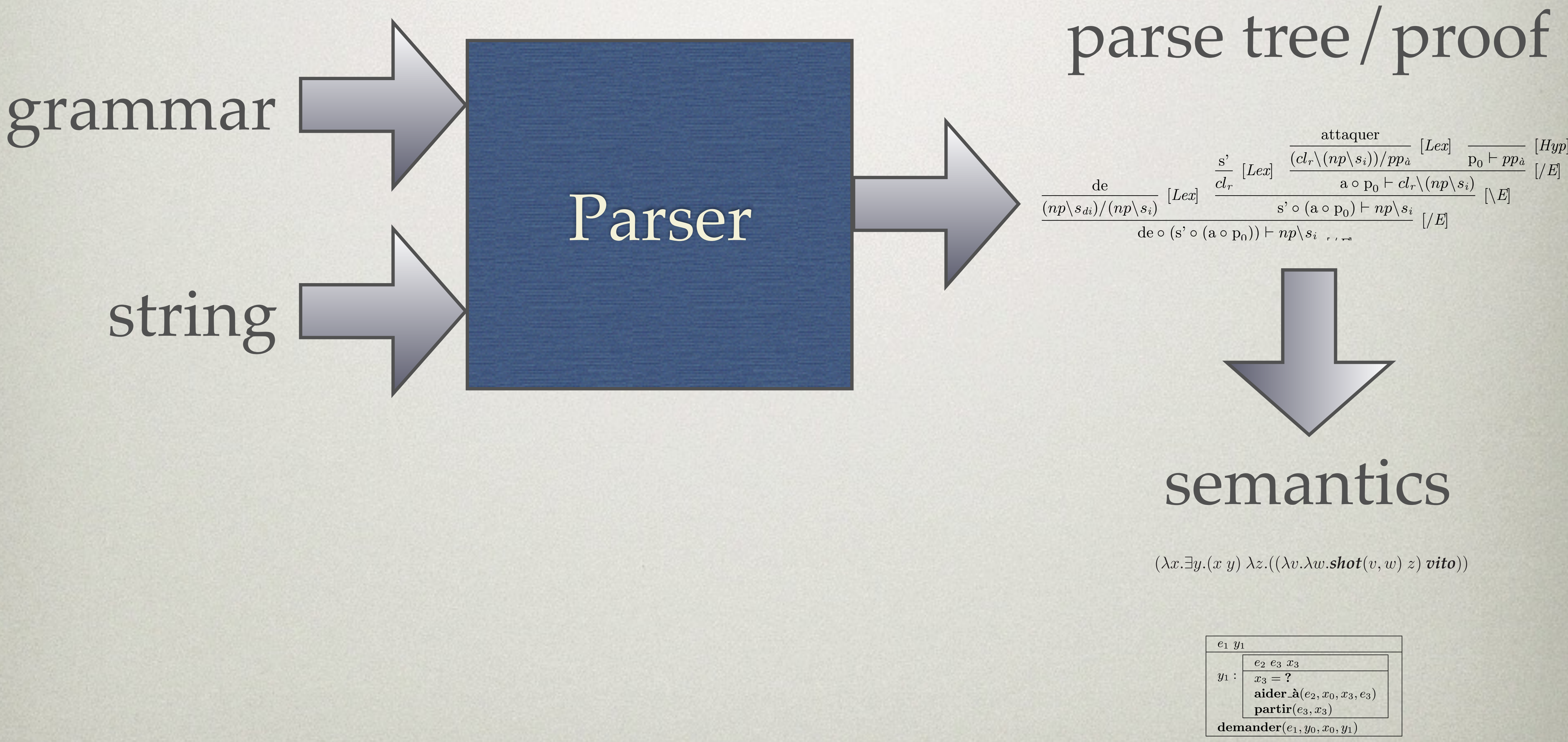
Computational SEmantics for Debates

Richard Moot
(CNRS, LaBRI)
NASSLLI 2016/SaLMoM

Introduction

- Can a computer understand natural language well enough for useful applications?
- Many applications require deep representation of the meaning of a sentence.
- Method: combination of symbolic and non-symbolic methods

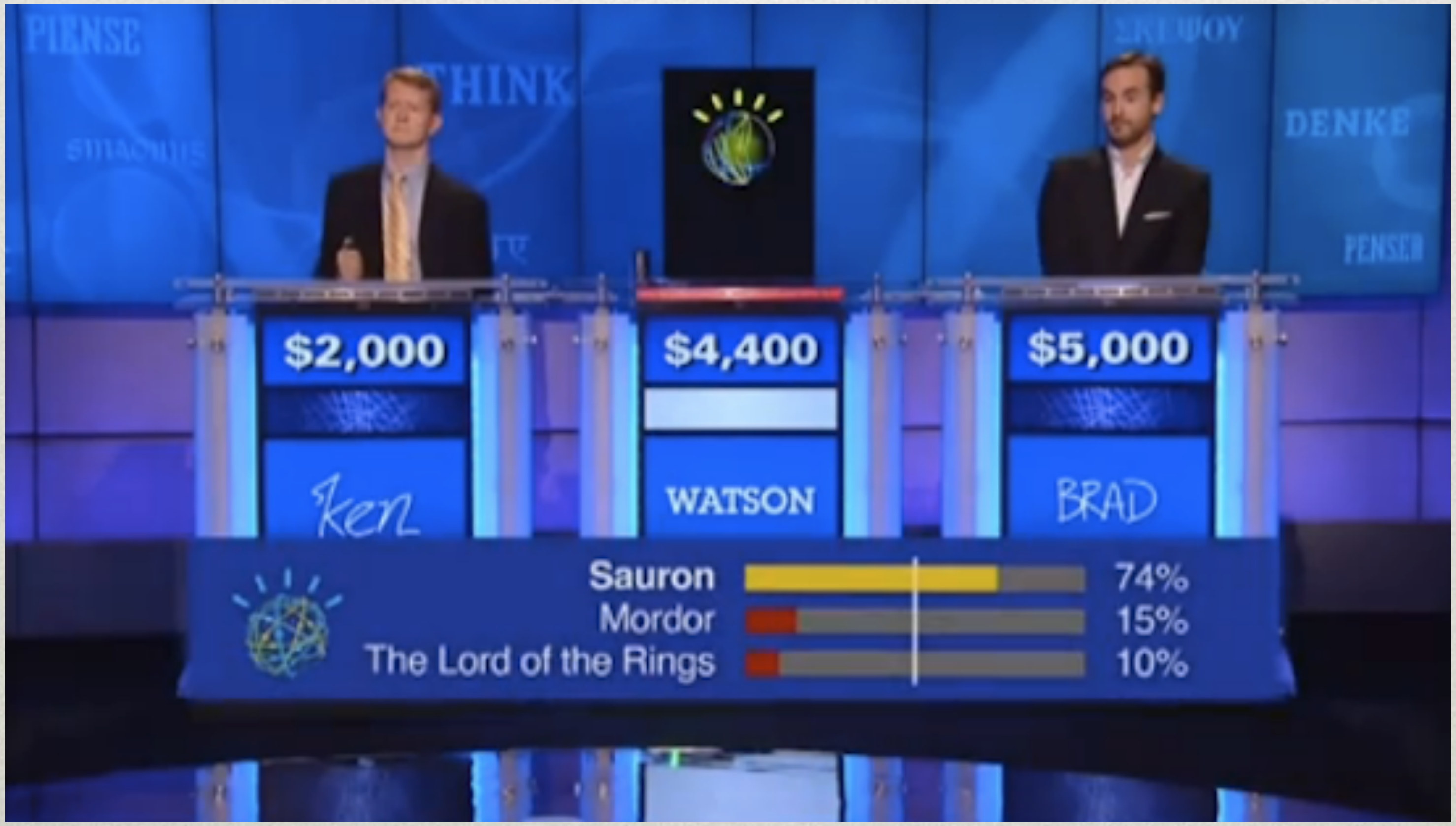
Architecture Overview



Question Answering
Natural Language Interface

Question Answering
Natural Language Interface

Wanted for general evilness; last seen at the tower of Barad-Dur; it's a giant eye, folks, kinda hard to miss



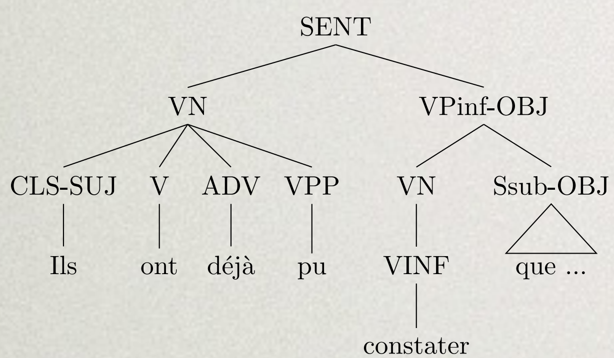
Question Answering
Natural Language Interface

U.S. Cities

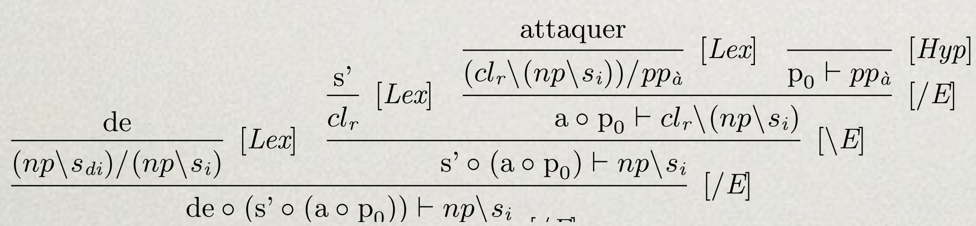
Its largest airport is named for a World War II hero; its second largest, for a World War II battle

Outline

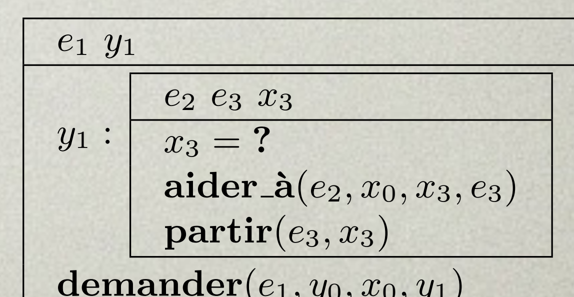
French Treebank



Grammar Extraction

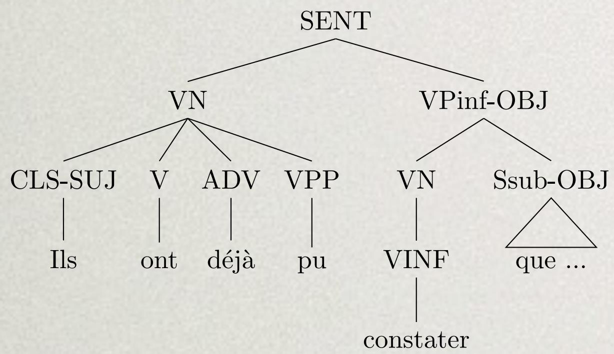


Applications



Outline

French Treebank



Grammar Extraction

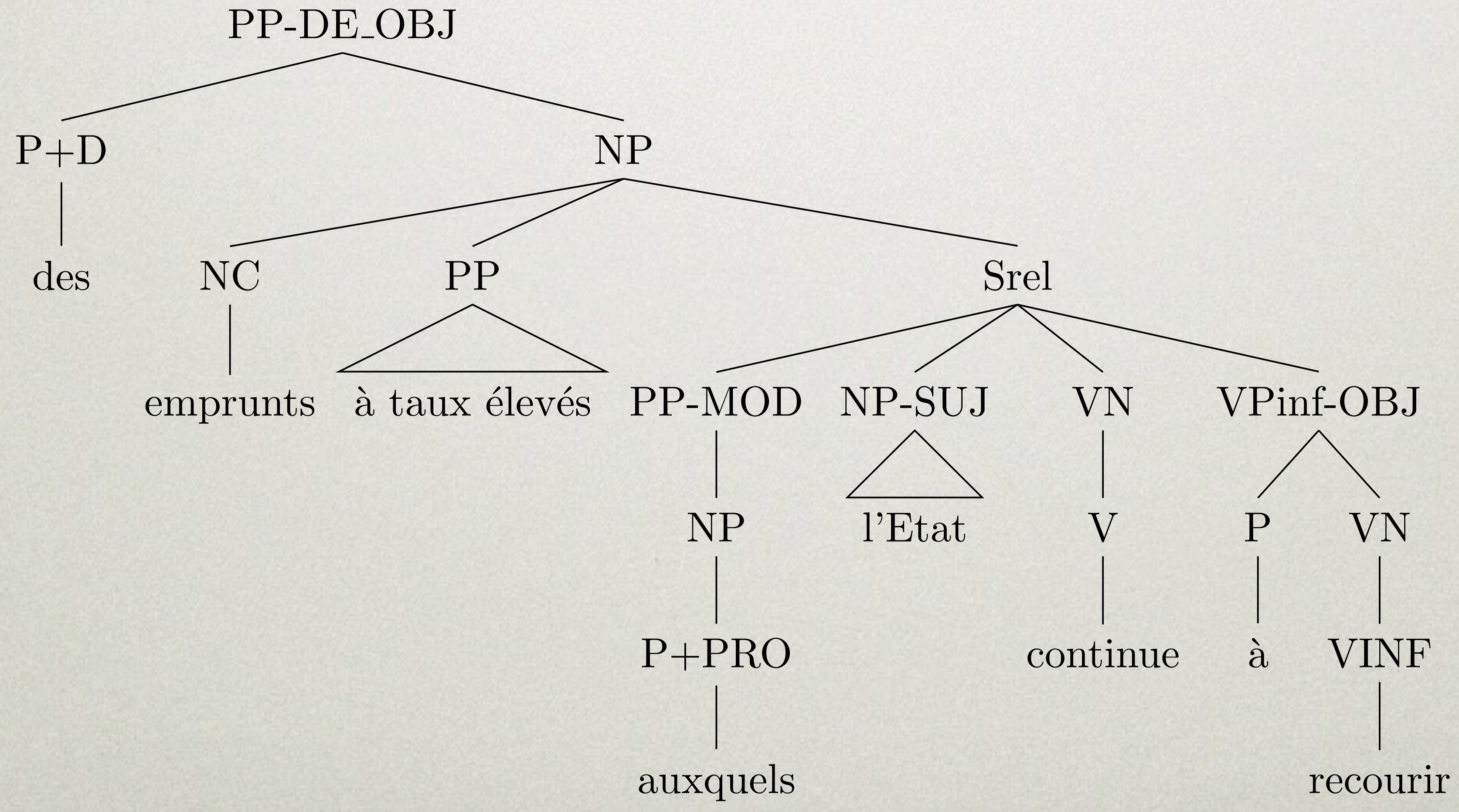
$$
\dfrac{\dfrac{\text{de}}{(np\backslash s_{di})/(np\backslash s_i)}\,[Lex] \quad \dfrac{\dfrac{\text{s'}}{cl_r}\,[Lex] \quad \dfrac{\dfrac{\text{attaquer}}{(cl_r\backslash(np\backslash s_i))/pp_{à}}\,[Lex] \quad \dfrac{}{\text{p}_0 \vdash pp_{à}}\,[Hyp]_1}{\text{a} \circ \text{p}_0 \vdash cl_r\backslash(np\backslash s_i)}\,[/E]}{\text{s'} \circ (\text{a} \circ \text{p}_0) \vdash np\backslash s_i}\,[\backslash E]}{\text{de} \circ (\text{s'} \circ (\text{a} \circ \text{p}_0)) \vdash np\backslash s_i}\,[/E]
$$

Applications

$$
\begin{array}{|l|}
\hline
e_1\ y_1 \\
\hline
y_1 : \begin{array}{|l|}
\hline
e_2\ e_3\ x_3 \\
\hline
x_3 = ? \\
\textbf{aider\_à}(e_2, x_0, x_3, e_3) \\
\textbf{partir}(e_3, x_3) \\
\hline
\end{array} \\
\textbf{demander}(e_1, y_0, x_0, y_1) \\
\hline
\end{array}
$$

The French Treebank

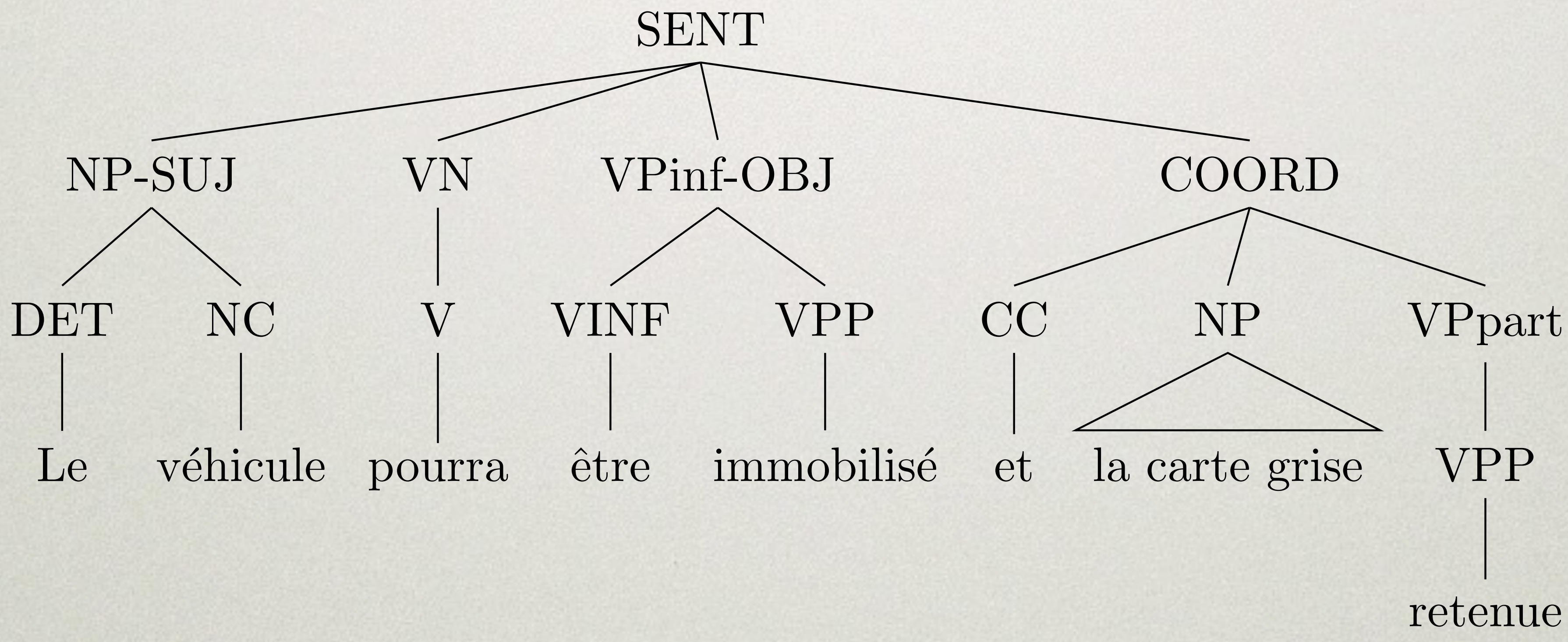
Example PP from the corpus (slightly simplified)



≈ continue_to_resort_to(state,loans)

The French Treebank

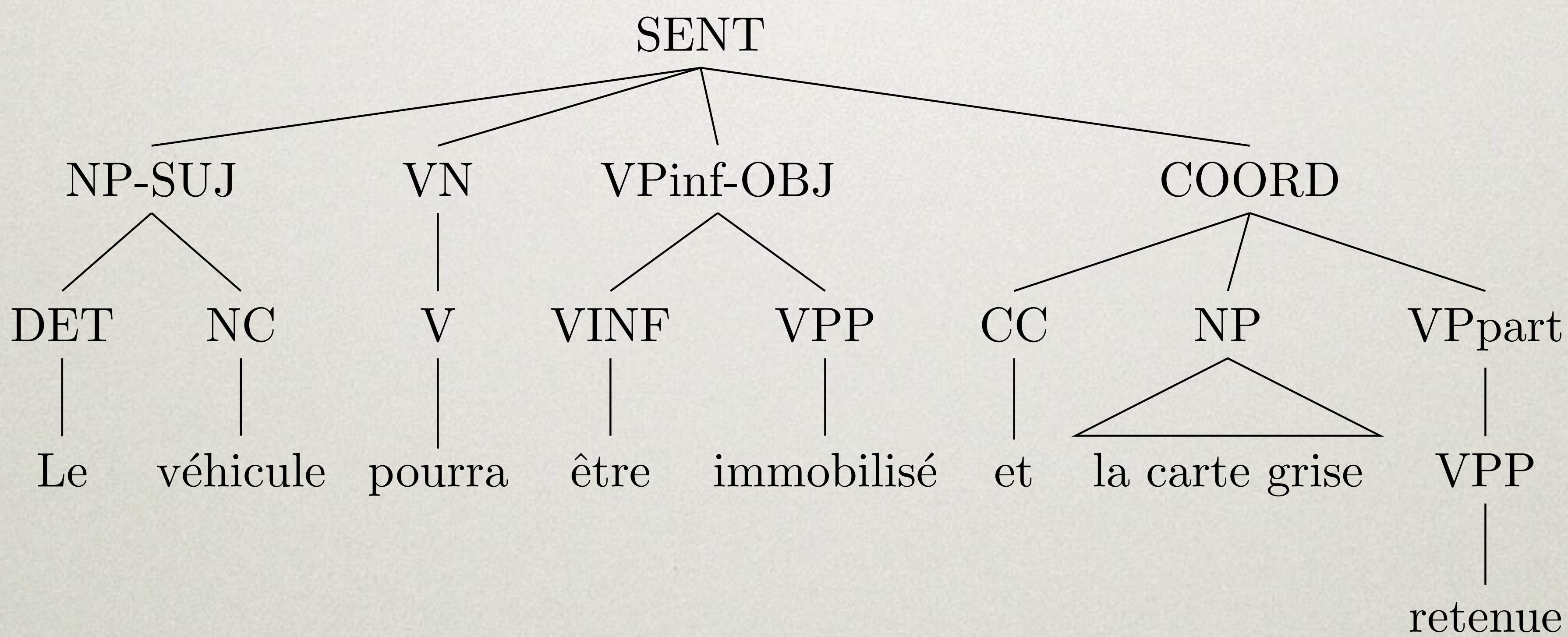
Example from the corpus (slightly simplified)



“The car could be immobilized and the car registration (could be) withheld”

The French Treebank

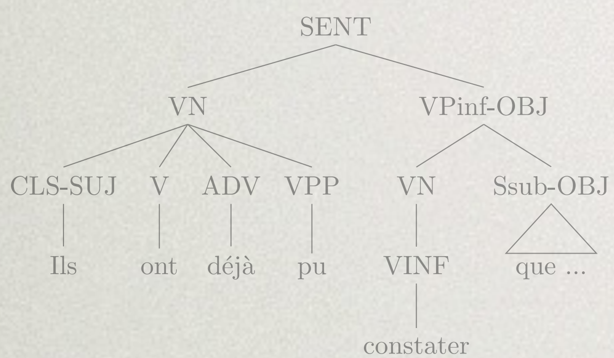
Example from the corpus (slightly simplified)



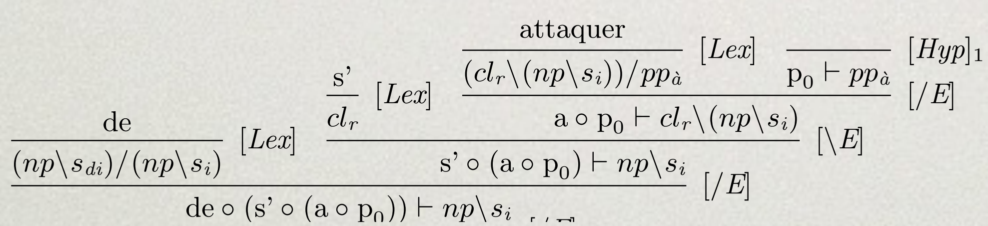
$\approx \exists x.\text{pourvoir}(\text{immobiliser}(x,\text{véhicule}))\wedge$
$\exists y.\text{pouvoir}(\text{retenir}(y,\text{cart\_grise}))$

Outline

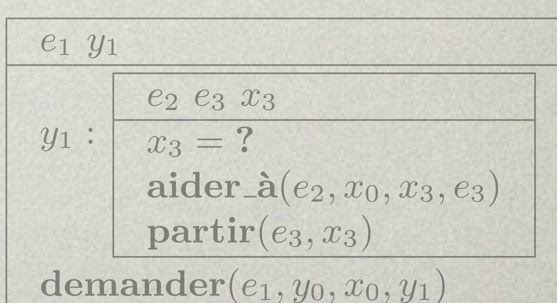
French Treebank



Grammar Extraction



Applications



AB Grammars and Elimination Rules

- atomic formulas np, n, pp, s, s_{inf}, s_{pass}, s_{ppart}, s_{whq}, ...
- if A and B are formulas, then A/B and B\A are formulas as well

$$\frac{A/B \quad B}{A}[/E] \qquad\qquad \frac{B \quad B\backslash A}{A}[\backslash E]$$

AB Grammars

np	Jean, l'étudiant, …
n	étudiant, économie, …
s	Jean dort, Jean aime Marie
np\s	dort, aime Marie
np/n	un, chaque, l'
(np\s)/np	aime, étudie

Example

un	étudiant	dort
np/n	n	np\s

$$\frac{A/B \quad B}{A}[/E] \qquad \frac{B \quad B\backslash A}{A}[\backslash E]$$

Example

un étudiant dort

$$\dfrac{np/n \qquad n}{np}[/E] \qquad np\backslash s$$

$$\dfrac{A/B \qquad B}{A}[/E] \qquad \dfrac{B \qquad B\backslash A}{A}[\backslash E]$$

Example

un étudiant

$$\dfrac{\dfrac{np/n \qquad n}{np}[/E] \qquad \overset{\text{dort}}{np\backslash s}}{s}[/E]$$

$$\dfrac{A/B \qquad B}{A}[/E] \qquad \dfrac{B \qquad B\backslash A}{A}[\backslash E]$$

From AB grammars to Type-Logical Grammars

- though AB grammars work well enough for simple cases, getting the semantics right requires a somewhat richer system
- introduction rules ("traces" and their semantics)
- structural rules ("movement", essentially restricted tree rewrite operations)

From AB grammars to Type-Logical Grammars

$$\frac{\begin{array}{c}\ldots.\ \ [B]^i \\ \\ A\end{array}}{A/B}[/I]^i \qquad \frac{\begin{array}{c}[B]^i\ \ldots \\ \\ A\end{array}}{B/A}[\backslash I]^i$$

[B] is the rightmost free hypothesis in the proof

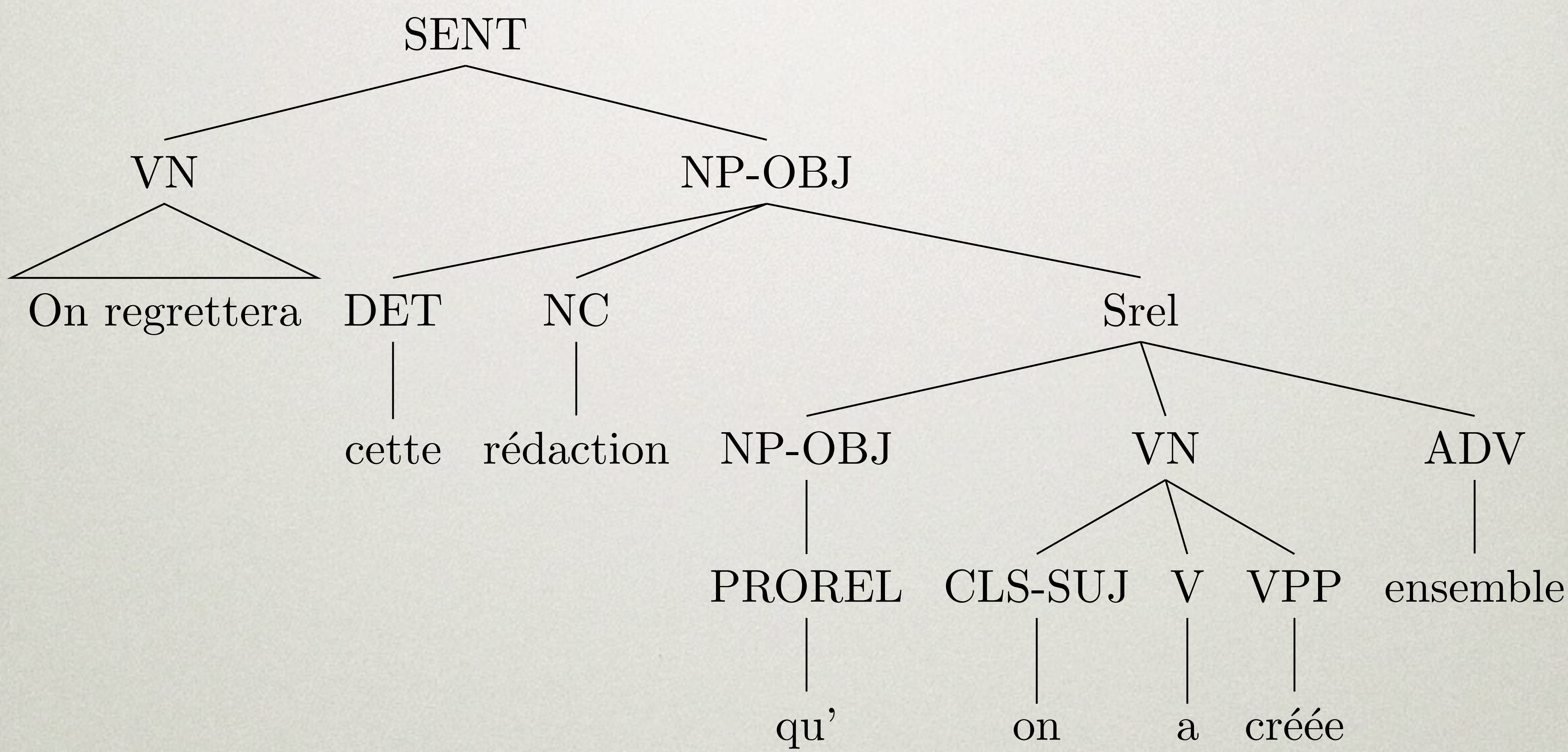
[B] is the leftmost free hypothesis in the proof

"..." contains at least one other non-discharged formula

Each introduction rule discharges exactly one formula.

A unique integer *i* links the rule application with this formula

Introduction Rules: Example



Introduction Rules: Example

redaction	qu'	on	a	créée
n	$(n\backslash n)/(s/np)$	np	$(np\backslash s)/(np\backslash s_{ppart})$	$(np\backslash s_{ppart})/np$

Introduction Rules: Example

redaction	qu'	on	a	créée	
n	$(n\backslash n)/(s/np)$	np	$(np\backslash s)/(np\backslash s_{ppart})$	$(np\backslash s_{ppart})/np$	np

Introduction Rules: Example

redaction qu' on a créée

$$n \quad (n\backslash n)/(s/np) \quad np \quad (np\backslash s)/(np\backslash s_{ppart}) \quad \dfrac{(np\backslash s_{ppart})/np \qquad np}{np\backslash s_{ppart}}\,[/E]$$

Introduction Rules: Example

redaction qu' on a créée

$$n \quad (n\backslash n)/(s/np) \quad np \quad \dfrac{(np\backslash s)/(np\backslash s_{ppart}) \qquad \dfrac{(np\backslash s_{ppart})/np \qquad np}{np\backslash s_{ppart}}[/E]}{np\backslash s}[/E]$$

Introduction Rules: Example

redaction qu' créée

$$\dfrac{}{}$$

n $(n\backslash n)/(s/np)$

a

$(np\backslash s_{ppart})/np$ np

$\dfrac{(np\backslash s_{ppart})/np \quad np}{np\backslash s_{ppart}}$ [/E]

on $(np\backslash s)/(np\backslash s_{ppart})$

$\dfrac{(np\backslash s)/(np\backslash s_{ppart}) \quad np\backslash s_{ppart}}{np\backslash s}$ [/E]

np

$\dfrac{np \quad np\backslash s}{s}$ [\E]

Introduction Rules: Example

redaction qu' créée

$n \quad (n\backslash n)/(s/np)$

$$\dfrac{\dfrac{np \qquad \dfrac{(np\backslash s)/(np\backslash s_{ppart}) \qquad \dfrac{(np\backslash s_{ppart})/np \qquad [np]^1}{np\backslash s_{ppart}}[/E]}{np\backslash s}[/E]}{s}[\backslash E]}{s/np}[/I]^1$$

on a

(Word labels: "on" above np; "a" above $(np\backslash s)/(np\backslash s_{ppart})$; "créée" above $(np\backslash s_{ppart})/np$.)

Introduction Rules: Example

redaction

n

créée

a

$(np\backslash s_{ppart})/np$ $[np]^1$

[/E]

on $(np\backslash s)/(np\backslash s_{ppart})$ $np\backslash s_{ppart}$

[/E]

np $np\backslash s$ [\E]

qu’

s

[/I]^1

$(n\backslash n)/(s/np)$ s/np

[/E]

$n\backslash n$

Introduction Rules: Example

$$\dfrac{\overset{\text{redaction}}{n} \qquad \dfrac{\overset{\text{qu'}}{(n\backslash n)/(s/np)} \qquad \dfrac{\dfrac{\overset{\text{on}}{np} \qquad \dfrac{\overset{\text{a}}{(np\backslash s)/(np\backslash s_{ppart})} \qquad \dfrac{\overset{\text{créée}}{(np\backslash s_{ppart})/np} \qquad [np]^1}{np\backslash s_{ppart}}\,[/E]}{np\backslash s}\,[/E]}{s}\,[\backslash E]}{s/np}\,[/I]^1}{n\backslash n}\,[/E]}{n}\,[\backslash E]$$

Beyond AB grammarS Introduction Rules

qu'	on	a	créée	ensemble
$(n\backslash n)/(s/np)$	np	$(np\backslash s)/(np\backslash s_{ppart})$	$(np\backslash s_{ppart})/np$	$(np\backslash s)\backslash(np\backslash s)$

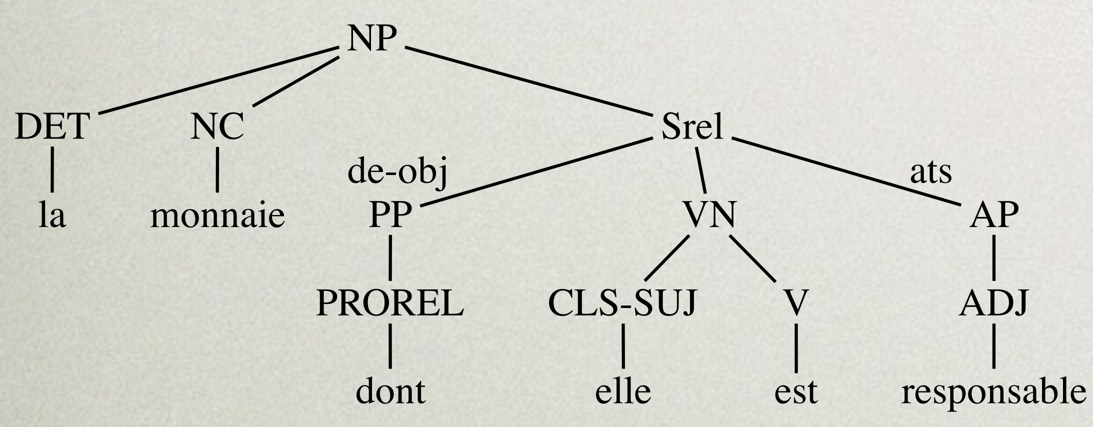
Beyond AB grammarS Introduction Rules

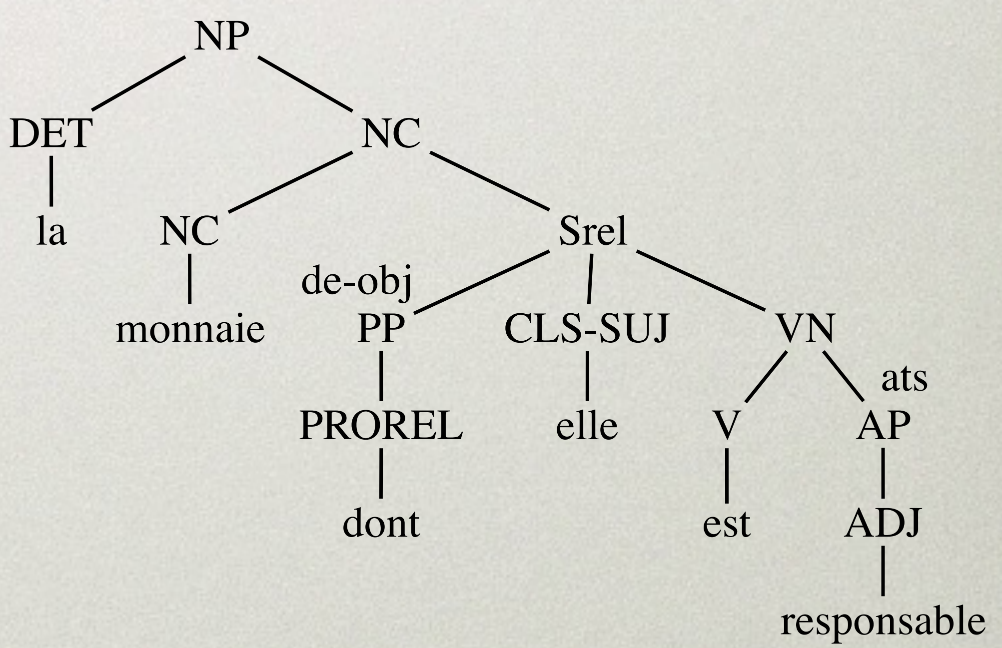
$$\ldots.\ [B]^i\ \ldots.$$

$$\frac{A}{A/\Diamond\Box B}\ [/I]^i$$

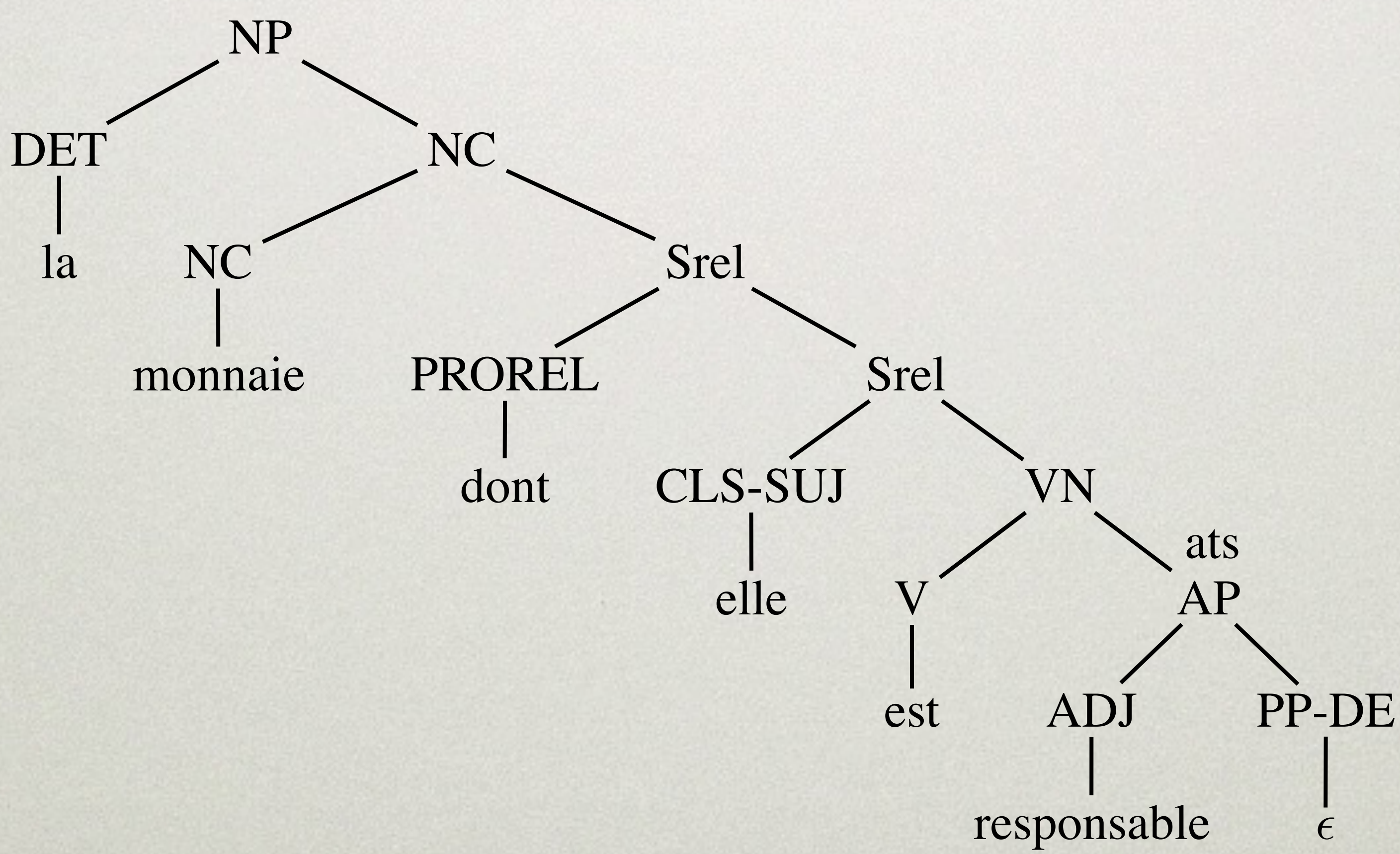
qu'	on	a	créée	ensemble
$(n\backslash n)/(s/\Diamond\Box np)$	np	$(np\backslash s)/(np\backslash s_{ppart})$	$(np\backslash s_{ppart})/np$	$(np\backslash s)\backslash(np\backslash s)$

Treebank Extraction

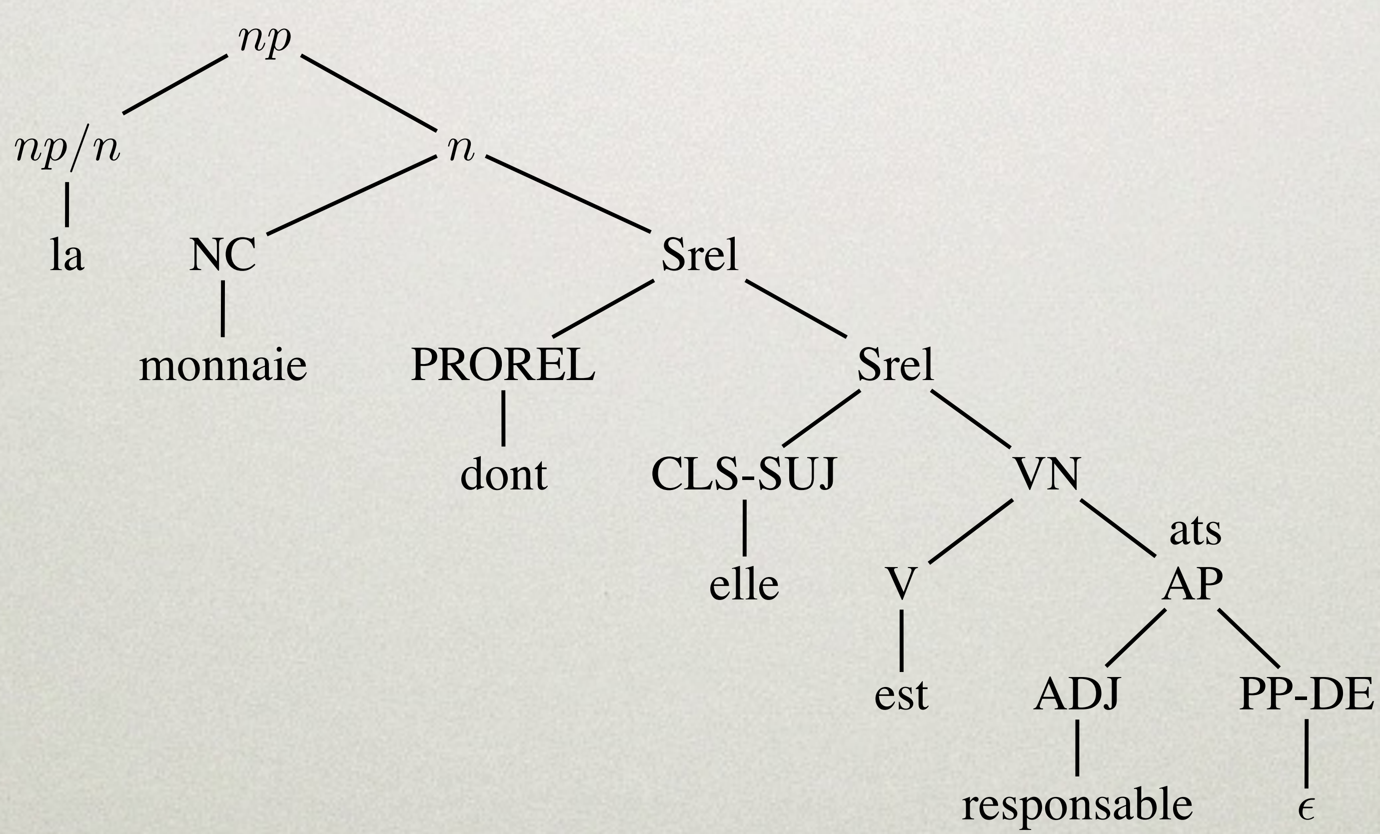




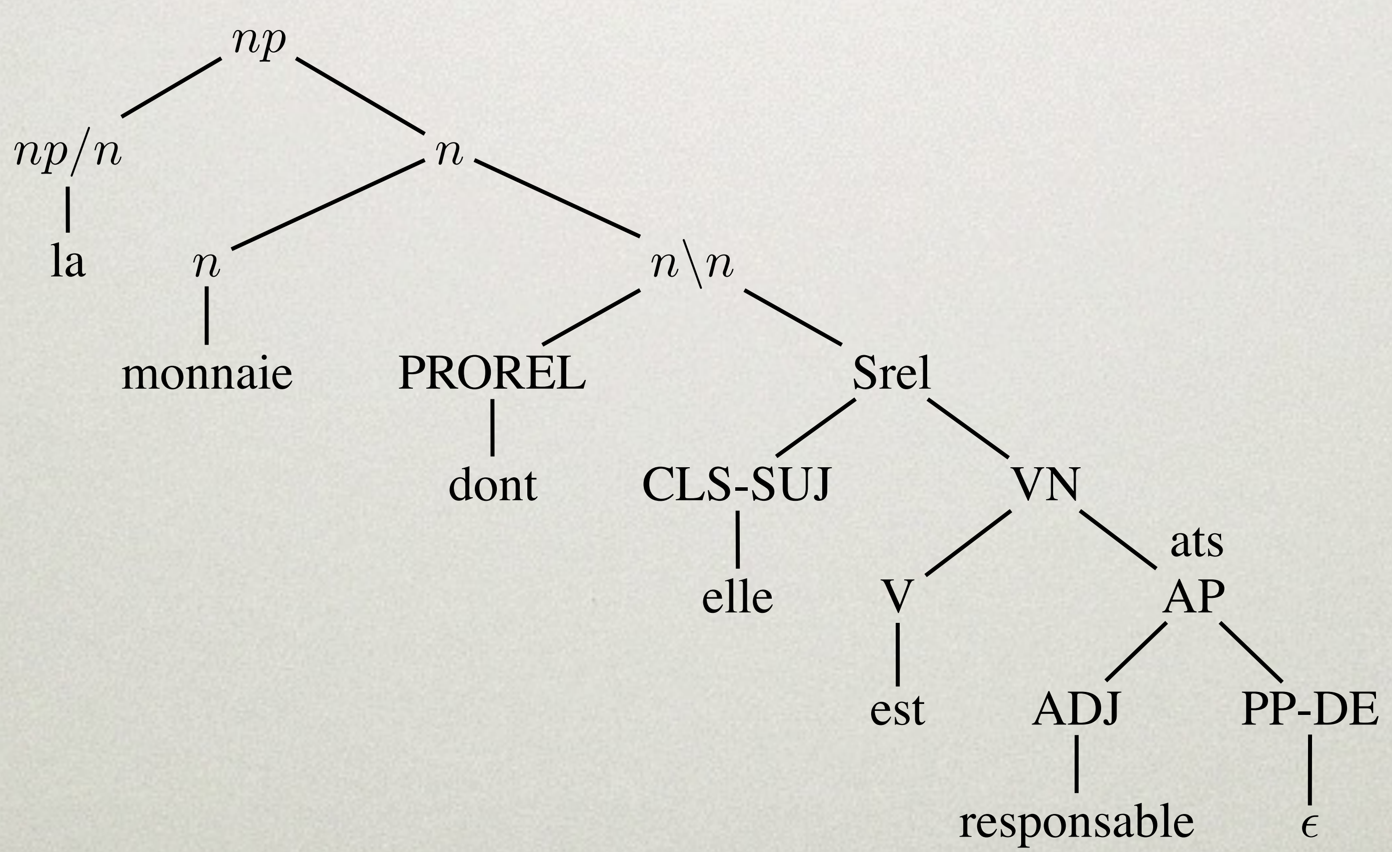
Treebank Extraction



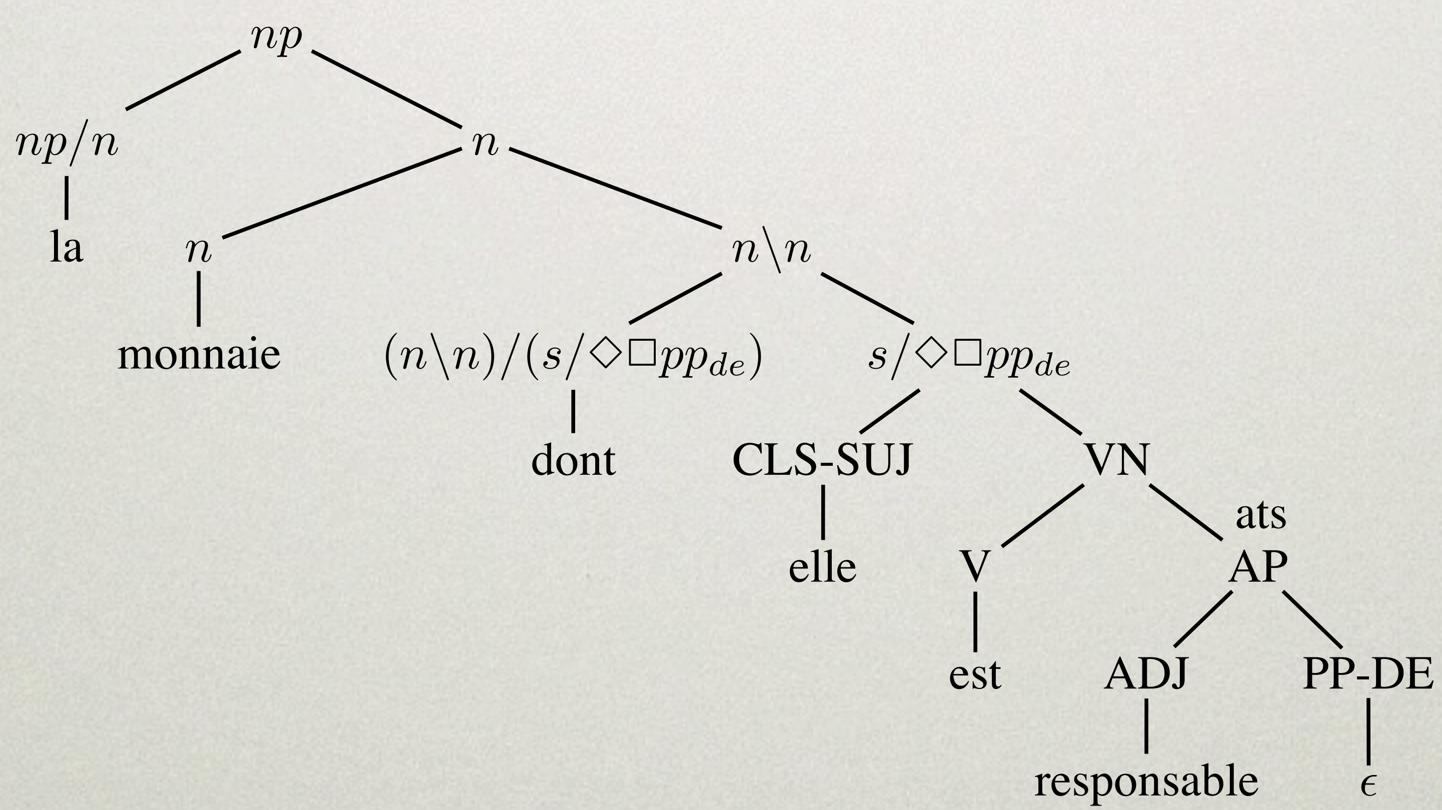
Treebank Extraction



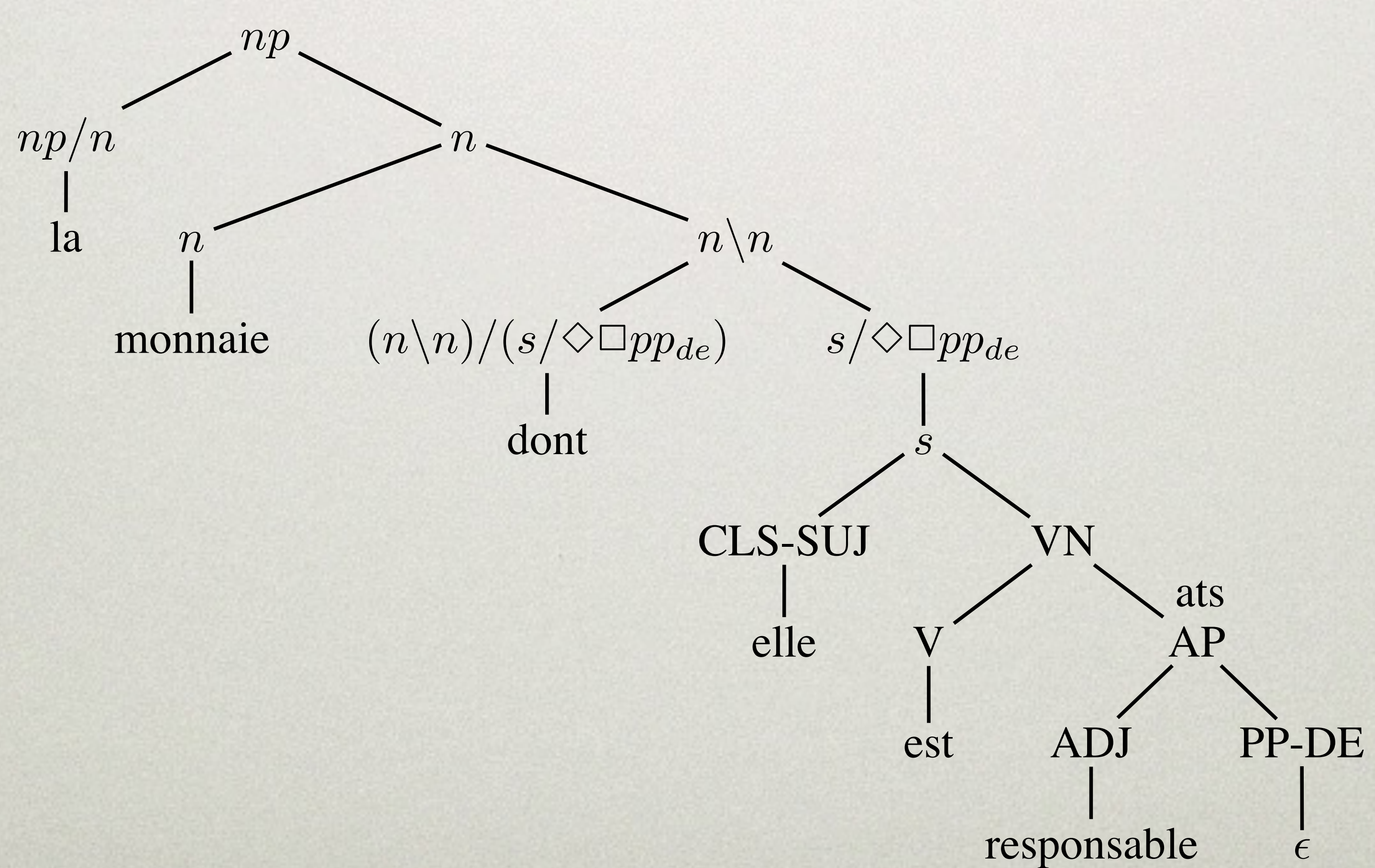
Treebank Extraction



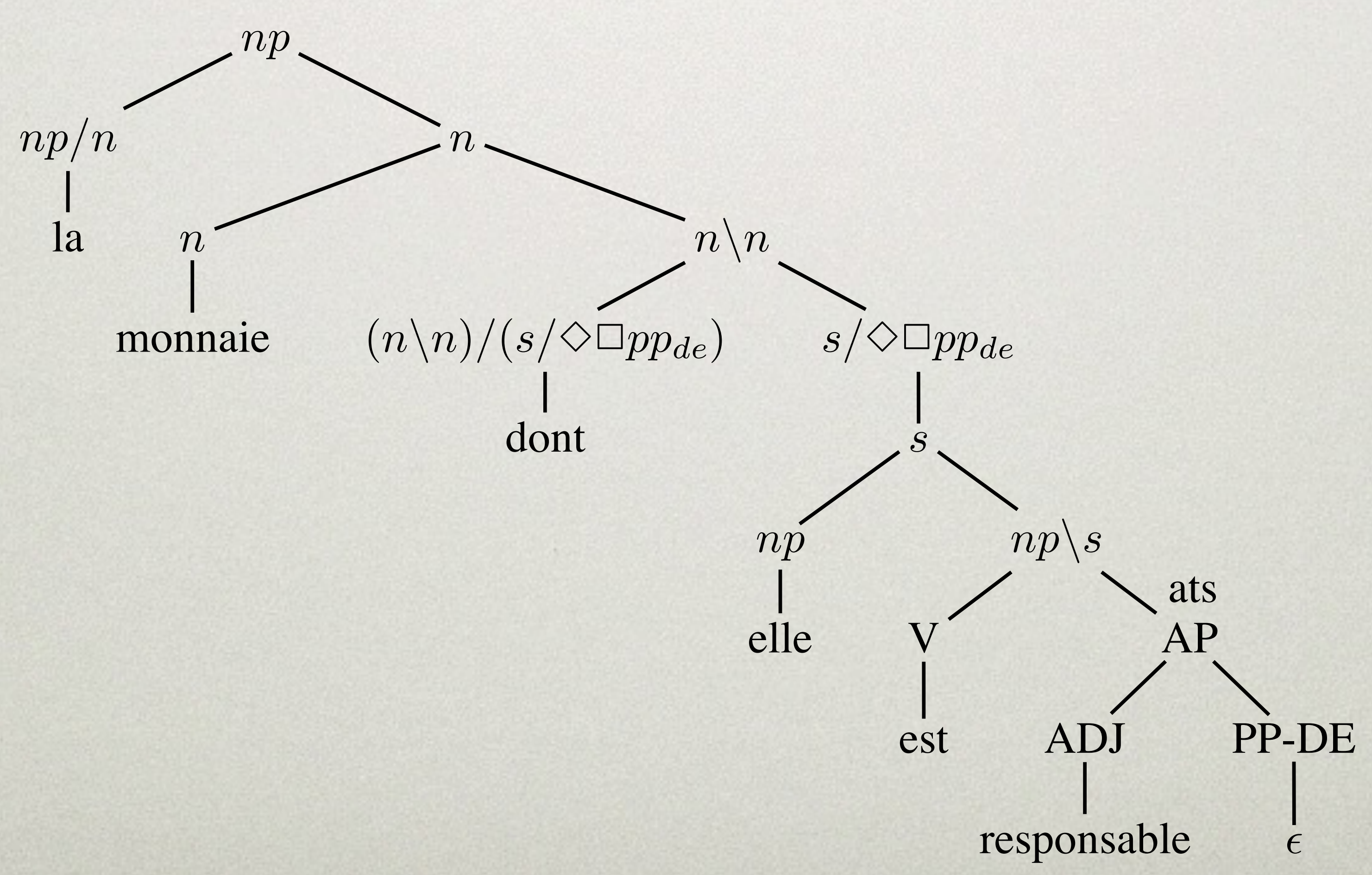
Treebank Extraction



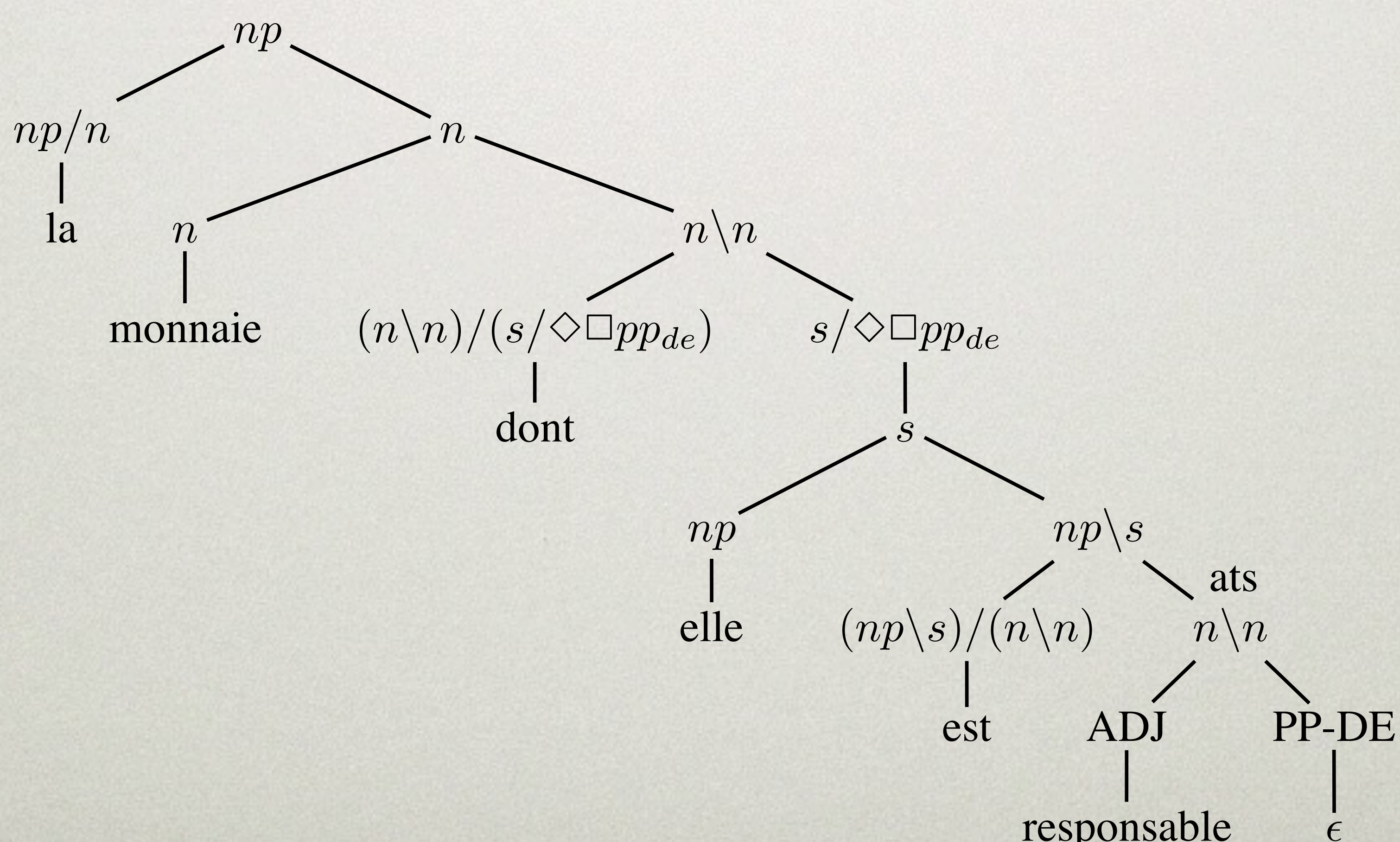
Treebank Extraction



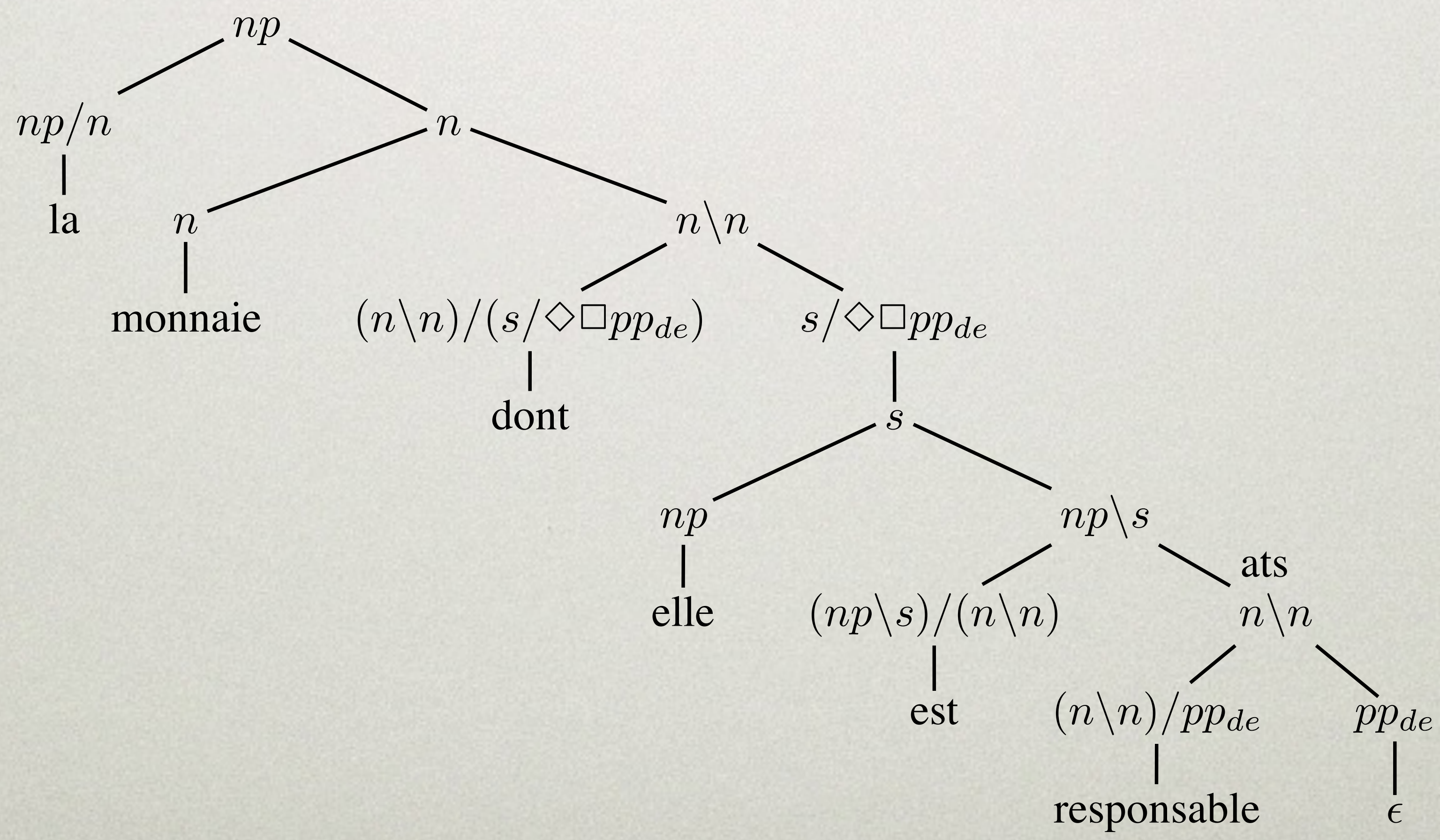
Treebank Extraction



Treebank Extraction

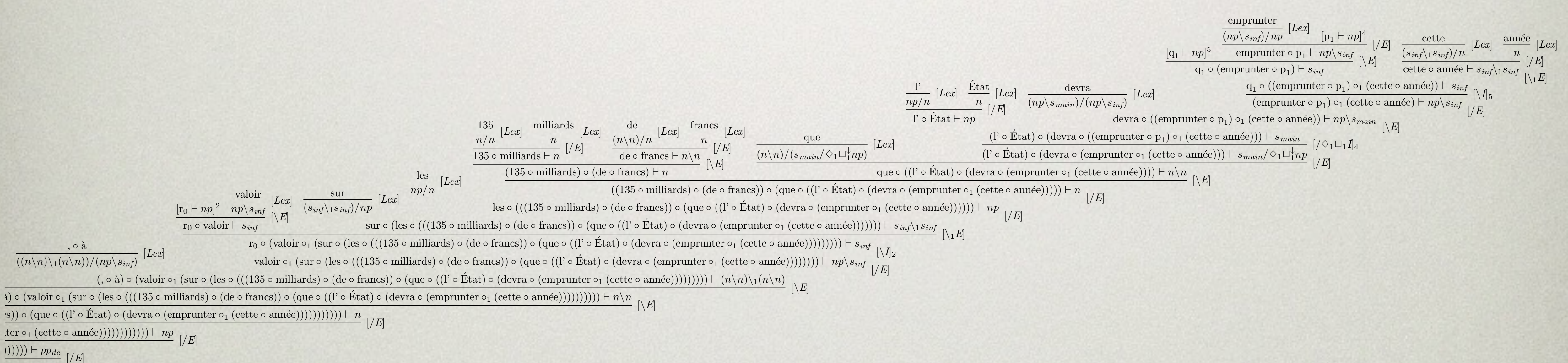


Treebank Extraction



Treebank Example

Treebank Example: Zoom 1



Treebank Example: Zoom2

$$\frac{\text{emprunter}}{(np\backslash s_{inf})/np}[Lex] \qquad [\text{p}_1 \vdash np]^4$$

$$\frac{(np\backslash s_{inf})/np \qquad [\text{p}_1 \vdash np]^4}{\text{emprunter} \circ \text{p}_1 \vdash np\backslash s_{inf}}[/E]$$

$$\frac{[\text{q}_1 \vdash np]^5 \qquad \text{emprunter} \circ \text{p}_1 \vdash np\backslash s_{inf}}{\text{q}_1 \circ (\text{emprunter} \circ \text{p}_1) \vdash s_{inf}}[\backslash E]$$

$$\frac{\frac{\text{cette}}{(s_{inf}\backslash_1 s_{inf})/n}[Lex] \qquad \frac{\text{année}}{n}[Lex]}{\text{cette} \circ \text{année} \vdash s_{inf}\backslash_1 s_{inf}}[/E]$$

$$\frac{\text{q}_1 \circ (\text{emprunter} \circ \text{p}_1) \vdash s_{inf} \qquad \text{cette} \circ \text{année} \vdash s_{inf}\backslash_1 s_{inf}}{\text{q}_1 \circ ((\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash s_{inf}}[\backslash_1 E]$$

$$\frac{\text{q}_1 \circ ((\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash s_{inf}}{(\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année}) \vdash np\backslash s_{inf}}[\backslash I]_5$$

$$\frac{\frac{\text{devra}}{(np\backslash s_{main})/(np\backslash s_{inf})}[Lex] \qquad (\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année}) \vdash np\backslash s_{inf}}{\text{devra} \circ ((\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash np\backslash s_{main}}[/E]$$

$$\frac{\frac{\text{l'}}{np/n}[Lex] \qquad \frac{\text{État}}{n}[Lex]}{\text{l'} \circ \text{État} \vdash np}[/E]$$

$$\frac{\text{l'} \circ \text{État} \vdash np \qquad \text{devra} \circ ((\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash np\backslash s_{main}}{(\text{l'} \circ \text{État}) \circ (\text{devra} \circ ((\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}}[\backslash E]$$

$$\frac{(\text{l'} \circ \text{État}) \circ (\text{devra} \circ ((\text{emprunter} \circ \text{p}_1) \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}}{(\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}/\Diamond_1\Box_1^{\downarrow}np}[/\Diamond_1\Box_1 I]_4$$

$$\frac{\frac{\text{que}}{(n\backslash n)/(s_{main}/\Diamond_1\Box_1^{\downarrow}np)}[Lex] \qquad (\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}/\Diamond_1\Box_1^{\downarrow}np}{\text{que} \circ ((\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année})))) \vdash n\backslash n}[/E]$$

$$\frac{\frac{135}{n/n}[Lex] \qquad \frac{\text{milliards}}{n}[Lex]}{135 \circ \text{milliards} \vdash n}[/E] \qquad \frac{\frac{\text{de}}{(n\backslash n)/n}[Lex] \qquad \frac{\text{francs}}{n}[Lex]}{\text{de} \circ \text{francs} \vdash n\backslash n}[/E]$$

$$\frac{135 \circ \text{milliards} \vdash n \qquad \text{de} \circ \text{francs} \vdash n\backslash n}{(135 \circ \text{milliards}) \circ (\text{de} \circ \text{francs}) \vdash n}[\backslash E]$$

$$\frac{(135 \circ \text{milliards}) \circ (\text{de} \circ \text{francs}) \vdash n \qquad \text{que} \circ ((\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année})))) \vdash n\backslash n}{((135 \circ \text{milliards}) \circ (\text{de} \circ \text{francs})) \circ (\text{que} \circ ((\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année}))))) \vdash n}[\backslash E]$$

$$\frac{\frac{\text{les}}{np/n}[Lex] \qquad ((135 \circ \text{milliards}) \circ (\text{de} \circ \text{francs})) \circ (\text{que} \circ ((\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année}))))) \vdash n}{\text{les} \circ (((135 \circ \text{milliards}) \circ (\text{de} \circ \text{francs})) \circ (\text{que} \circ ((\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année})))))) \vdash np}[/E]$$

Treebank Example: Zoom3

$$\dfrac{\text{emprunter}}{(np\backslash s_{inf})/np}\,[Lex]$$

$$\dfrac{(np\backslash s_{inf})/np \qquad [\mathrm{p}_1 \vdash np]^4}{\text{emprunter} \circ \mathrm{p}_1 \vdash np\backslash s_{inf}}\,[/E]$$

$$\dfrac{[\mathrm{q}_1 \vdash np]^5 \qquad \text{emprunter} \circ \mathrm{p}_1 \vdash np\backslash s_{inf}}{\mathrm{q}_1 \circ (\text{emprunter} \circ \mathrm{p}_1) \vdash s_{inf}}\,[\backslash E]$$

$$\dfrac{\text{cette}}{(s_{inf}\backslash_1 s_{inf})/n}\,[Lex] \qquad \dfrac{\text{année}}{n}\,[Lex]$$

$$\dfrac{(s_{inf}\backslash_1 s_{inf})/n \qquad n}{\text{cette} \circ \text{année} \vdash s_{inf}\backslash_1 s_{inf}}\,[/E]$$

$$\dfrac{\mathrm{q}_1 \circ (\text{emprunter} \circ \mathrm{p}_1) \vdash s_{inf} \qquad \text{cette} \circ \text{année} \vdash s_{inf}\backslash_1 s_{inf}}{\mathrm{q}_1 \circ ((\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash s_{inf}}\,[\backslash_1 E]$$

$$\dfrac{\mathrm{q}_1 \circ ((\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash s_{inf}}{(\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année}) \vdash np\backslash s_{inf}}\,[\backslash I]_5$$

$$\dfrac{\dfrac{\text{devra}}{(np\backslash s_{main})/(np\backslash s_{inf})}\,[Lex] \qquad (\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année}) \vdash np\backslash s_{inf}}{\text{devra} \circ ((\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash np\backslash s_{main}}\,[/E]$$

$$\dfrac{\dfrac{\text{l'}}{np/n}\,[Lex] \qquad \dfrac{\text{État}}{n}\,[Lex]}{\text{l'} \circ \text{État} \vdash np}\,[/E]$$

$$\dfrac{\text{l'} \circ \text{État} \vdash np \qquad \text{devra} \circ ((\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année})) \vdash np\backslash s_{main}}{(\text{l'} \circ \text{État}) \circ (\text{devra} \circ ((\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}}\,[\backslash E]$$

$$\dfrac{(\text{l'} \circ \text{État}) \circ (\text{devra} \circ ((\text{emprunter} \circ \mathrm{p}_1) \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}}{(\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}/\Diamond_1\Box_1^{\downarrow} np}\,[/\Diamond_1\Box_1 I]_4$$

$$\dfrac{\dfrac{\text{que}}{(n\backslash n)/(s_{main}/\Diamond_1\Box_1^{\downarrow} np)}\,[Lex] \qquad (\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année}))) \vdash s_{main}/\Diamond_1\Box_1^{\downarrow} np}{\text{que} \circ ((\text{l'} \circ \text{État}) \circ (\text{devra} \circ (\text{emprunter} \circ_1 (\text{cette} \circ \text{année})))) \vdash n\backslash n}\,[/E]$$

$$[\backslash E]$$

Some Statistics for the Extracted Treebank

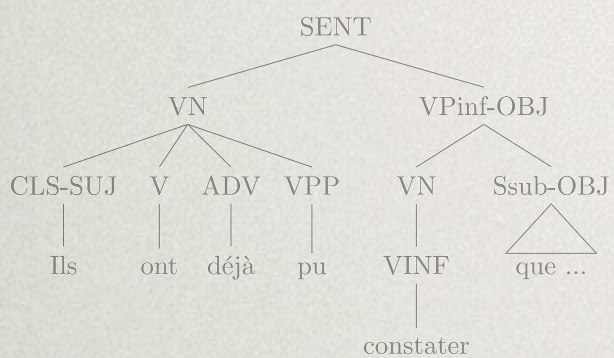
- 15,590 sentences, 425,918 words
- 45,270 distinct lexical entries
- 883 different formulas (with 659 occurring more than once)
- By comparison: 12,617 CFG rules

Differences With the CCGbank

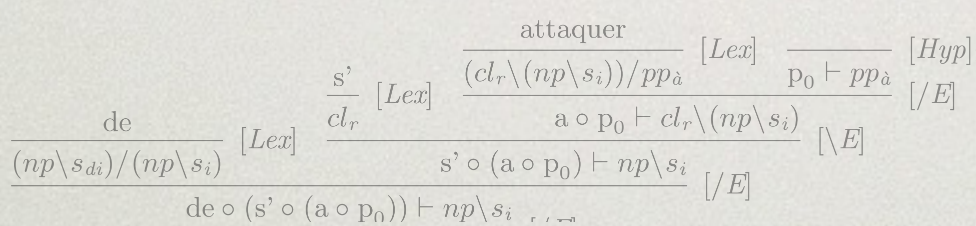
- Different treatment of extraction
- No lexical rules or non-logical axioms
- Coordination is handled in the lexicon
- Interpunction symbols can be assigned coordination-like formulas

Outline

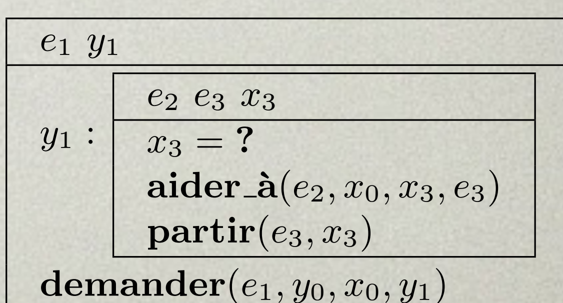
French Treebank



Grammar Extraction



Applications



Applications

- Wide-coverage parsing for French
- Wide-coverage semantics for French
- Toward analysis of debates

Wide-Coverage Parsing

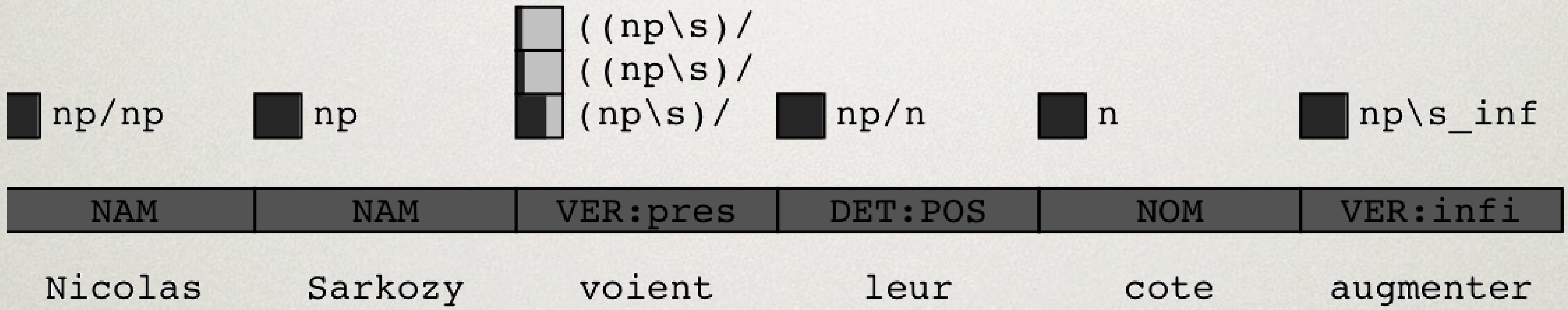
- How can we parse very big grammars efficiently?
- Bottlenecks: lexicon size, grammatical combinatorics

Lexicon Size

- Many frequent words occur with very many different formulas
- Classic solution: supertagging

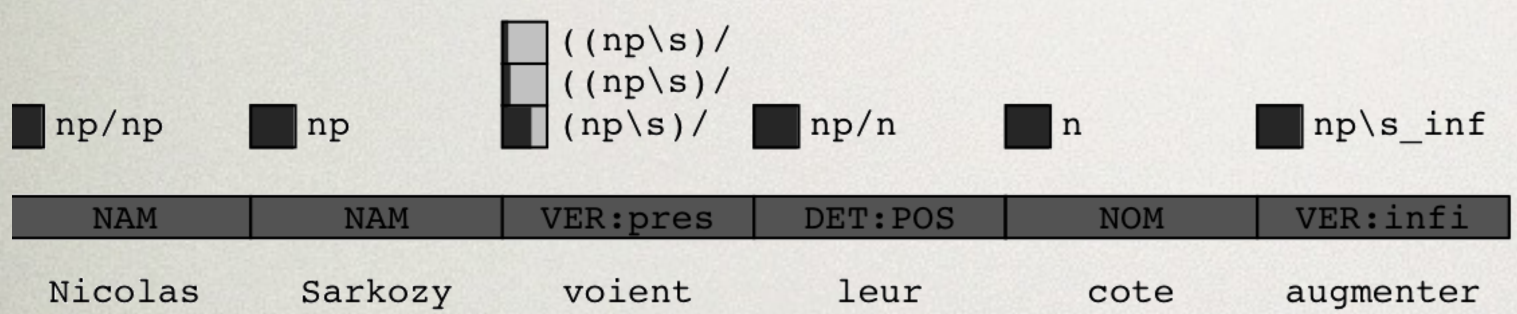
est - "is"	
$(np\backslash s)/np$	23,2 %
$(np\backslash s)/(n\backslash n)$	20,6 %
$(np\backslash s)/(np\backslash s_{pass})$	16,8 %
$(cl_r\backslash(np\backslash s))/(cl_r\backslash(np\backslash s_{ppart}))$	10,8 %
$(np\backslash s)/pp$	8,1 %
$(np\backslash s)/(np\backslash s_{ppart})$	6,3 %
$(np\backslash s)/(np\backslash s_{infX})$	2,8 %
$((np\backslash s)/s_q)/(n\backslash n)$	2,2 %

What Supertagging Does



- Supertagging = statistical finite state approximation of lexical lookup
- Assigns each word the contextually most likely (set of) formulas

What Supertagging Does



voient:

$(np\backslash s)/np$ (65.8%)

$((np\backslash s)/np)/pp$ (15.9%)

$((np\backslash s)/(np/s_{inf}))/np$ (10.7%)

- Supertagging = statistical finite state approximation of lexical lookup
- Assigns each word the contextually most likely (set of) formulas

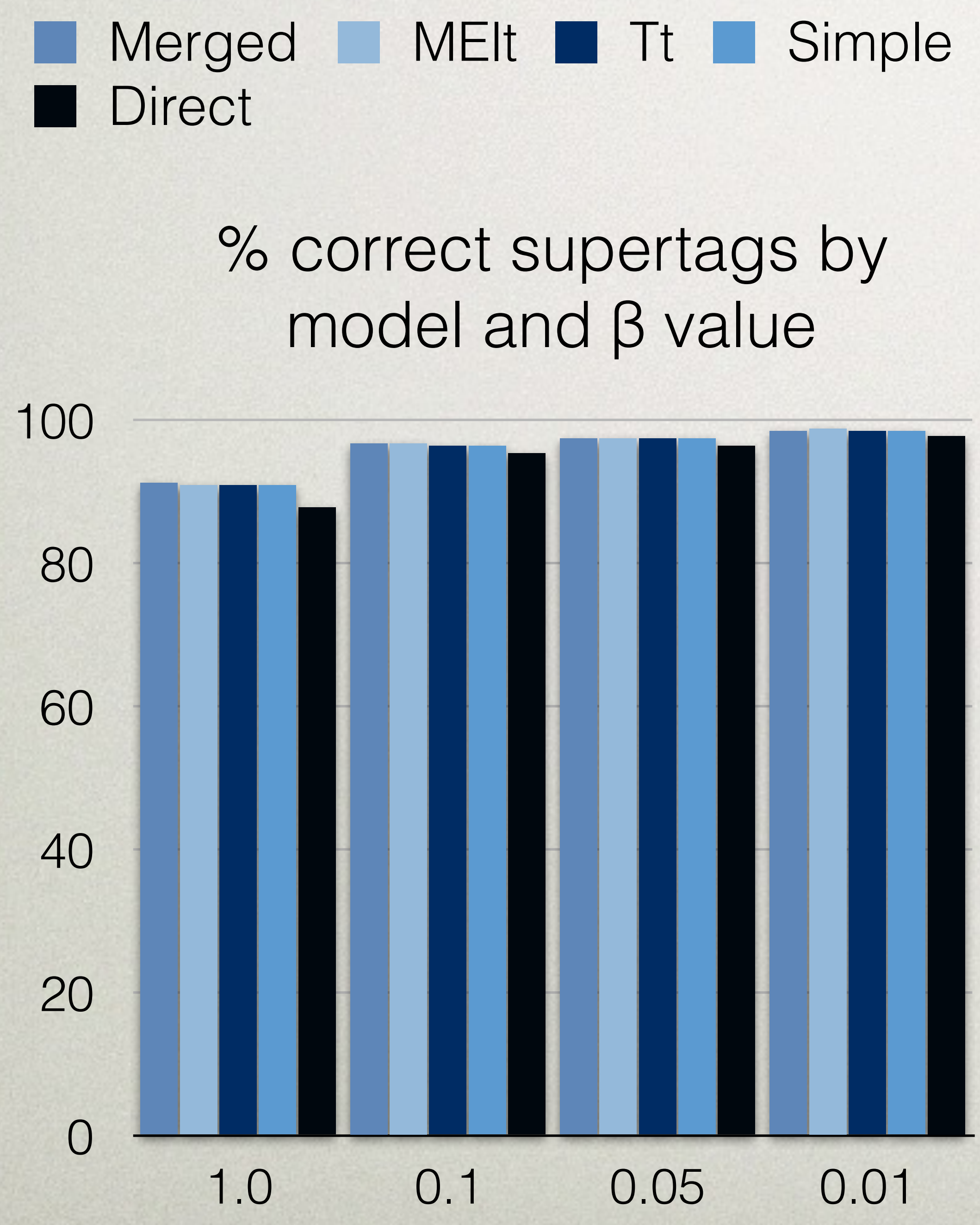
Supertagger Performance

Corpus	POS	Super	0,1	0,01	F/w
FTB	97,8 %	90,6 %	96,4 %	98,4 %	2,3

Supertagger Performance

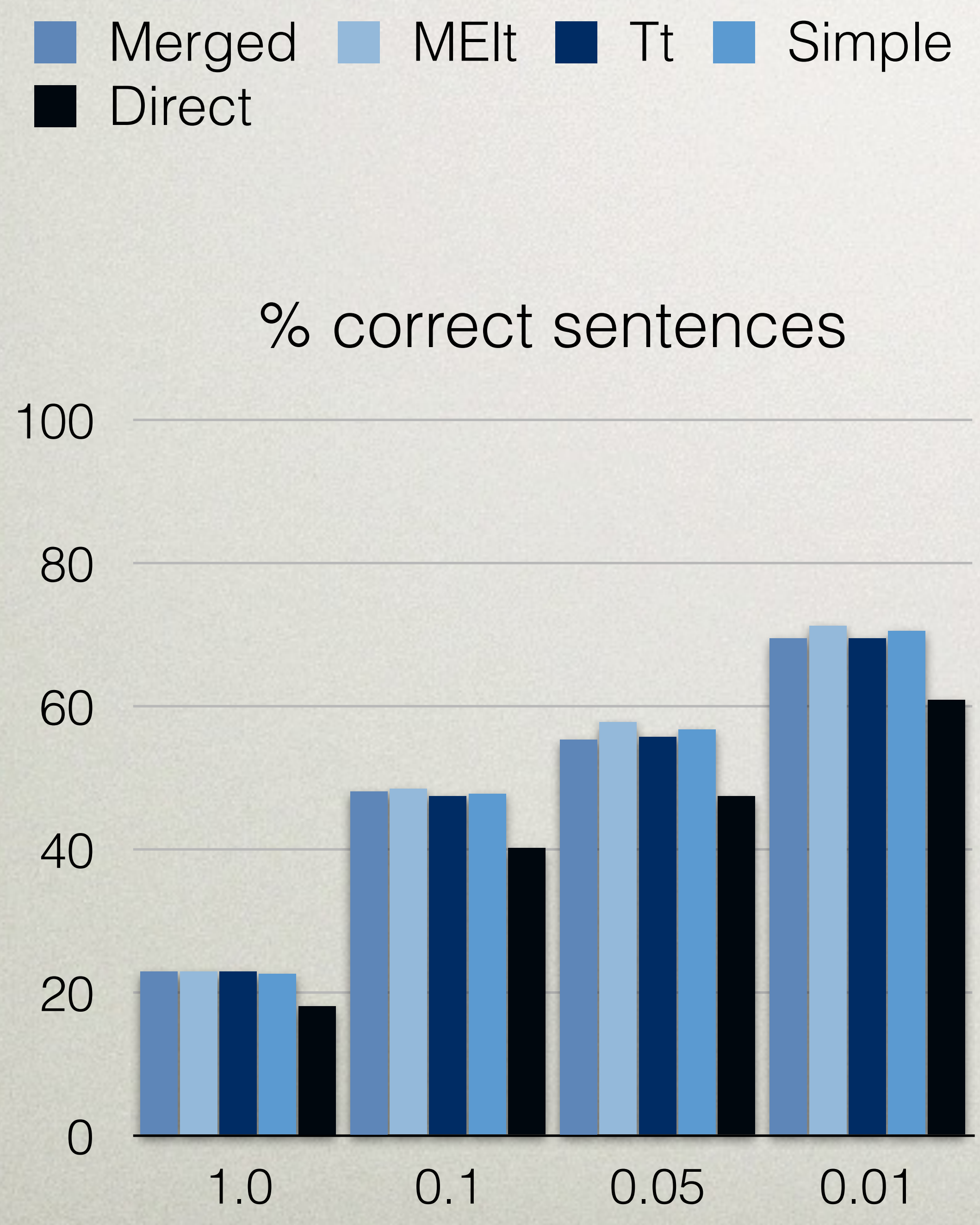
Corpus	POS	Super	0,1	0,01	F/w
FTB	97,8 %	90,6 %	96,4 %	98,4 %	2,3
Le Monde 2010	97,3 %	89,9 %	95,8 %	97,9 %	2,2
Sequoia	97,3 %	88,1 %	94,8 %	97,6 %	2,4
Itipy	95,7 %	86,7 %	93,8 %	97,1 %	2,6

Detailed Performance



- Results with the use of different values of β.
- In a sense, the β value allows us to trade coverage for efficiency: at higher values of β, we parse more sentences, but we do so more slowly.

Detailed Performance



- Finally, here is the percentage of sentences which are assigned the correct sequence of supertags for the different settings of β and the different POS models.
- Note that the number of sentences for which a parse is found is actually better (close to 90% at β=0.01)

Chart-Parsing Type-Logical Grammars

- Prototype exhaustive chart parser for Type-Logical Grammars, strongly inspired by Shieber e.a. (1995)
- Fine-tuned for grammars in the restricted form as produced by the extraction algorithm
- LaTeX output (natural deduction proofs and semantics)

Chart-Parsing Type-Logical Grammars

(1)

$$
\dfrac{
\dfrac{
\dfrac{\dfrac{\text{Alain}}{np/np}[Lex] \quad \dfrac{\text{Juppé}}{np}[Lex]}{\text{Alain} \circ \text{Juppé} \vdash np}[/E]
\quad
\dfrac{\dfrac{\text{et}}{(np\backslash np)/np}[Lex] \quad \dfrac{\dfrac{\text{Nicolas}}{np/np}[Lex] \quad \dfrac{\text{Sarkozy}}{np}[Lex]}{\text{Nicolas} \circ \text{Sarkozy} \vdash np}[/E]}{\text{et} \circ (\text{Nicolas} \circ \text{Sarkozy}) \vdash np\backslash np}[/E]
}{(\text{Alain} \circ \text{Juppé}) \circ (\text{et} \circ (\text{Nicolas} \circ \text{Sarkozy})) \vdash np}[\backslash E]
\quad
\dfrac{
\dfrac{\dfrac{\text{voient}}{((np\backslash s_{main})/(np\backslash s_{inf}))/np}[Lex] \quad \dfrac{\dfrac{\text{leur}}{np/n}[Lex] \quad \dfrac{\text{cote}}{n}[Lex]}{\text{leur} \circ \text{cote} \vdash np}[/E]}{\text{voient} \circ (\text{leur} \circ \text{cote}) \vdash (np\backslash s_{main})/(np\backslash s_{inf})}[/E]
\quad
\dfrac{\text{augmenter}}{np\backslash s_{inf}}[Lex]
}{(\text{voient} \circ (\text{leur} \circ \text{cote})) \circ \text{augmenter} \vdash np\backslash s_{main}}[/E]
}{((\text{Alain} \circ \text{Juppé}) \circ (\text{et} \circ (\text{Nicolas} \circ \text{Sarkozy}))) \circ ((\text{voient} \circ (\text{leur} \circ \text{cote})) \circ \text{augmenter}) \vdash s_{main}}[\backslash E]
$$

Semantics

- Type-logical grammar proofs are a subset of intuitionistic proofs, which correspond to lambda-terms.
- Lexical substitution followed by beta normalization gives us the full sentence meaning

Formulas as Types

- This is the standard way of assigning types to formulas in categorial grammar.
- Exception 1: np is lifted from e to $(e \rightarrow t) \rightarrow t$
- Exception 2: pp is assigned the same type as np

Formula	Type
type(np)	$(e \rightarrow t) \rightarrow t$
type(pp)	$(e \rightarrow t) \rightarrow t$
type(n)	$e \rightarrow t$
type(s)	t
type(B/A)	type(A) → type(B)
type(A\B)	type(A) → type(B)

Proofs as Typed Lambda Terms

- Proofs in categorial grammar correspond to lambda terms

$$\frac{t{:}A/B \qquad u{:}B}{(t\ u){:}A} \qquad \frac{u{:}B \qquad t{:}B\backslash A}{(t\ u){:}A}$$

- These lambda terms abstract away from the directions of the implications.

$$\frac{\begin{matrix}[x{:}B]\\ \vdots \\ t{:}A\end{matrix}}{A/B{:}\lambda x.t} \qquad \frac{\begin{matrix}[x{:}B]\\ \vdots \\ t{:}A\end{matrix}}{B\backslash A{:}\lambda x.t}$$

Lexical Semantics

- Montague-style semantics, where the meaning of *love* is **love′**
- Has the advantage of being scaleable, since many lexical entries follow a specific pattern
- Uses DRT

Semantics

Example entries (slightly simplified)

marché: λx. [| marché(x)]

Marie: λP. [y | nommé(y,Marie)] ⊕ (P y)

chaque: λP λQ.([z |] ⊕ (P z)) → (Q z)

Semantics

Example entries (slightly simplified)

aime:	$(np\backslash s)/np$	$\lambda y.\ \lambda x.$		aimer(x,y)
semble:	$(np\backslash s)/(np\backslash s_{inf})$	$\lambda P.\ \lambda x.$	s	sembler(s) s: (P x)
veut:	$(np\backslash s)/(np\backslash s_{inf})$	$\lambda P.\ \lambda x.$	s	vouloir(x,s) s: (P x)

Semantics

- 634 words in the lexicon, with idiosyncratic properties
- 346 lexical schemata, eg. (np\s)/np for word *w* means semantics *w′* (≡ $\lambda y.\lambda x.\ w'(x,y)$)

Going Further

- We have a system which, given a natural language sentence provides a representation of its meaning.
- For formal semanticists, the job is done (though more detailed analysis of many phenomena is surely necessary).
- How can we use these meanings?

Discourse and Debate

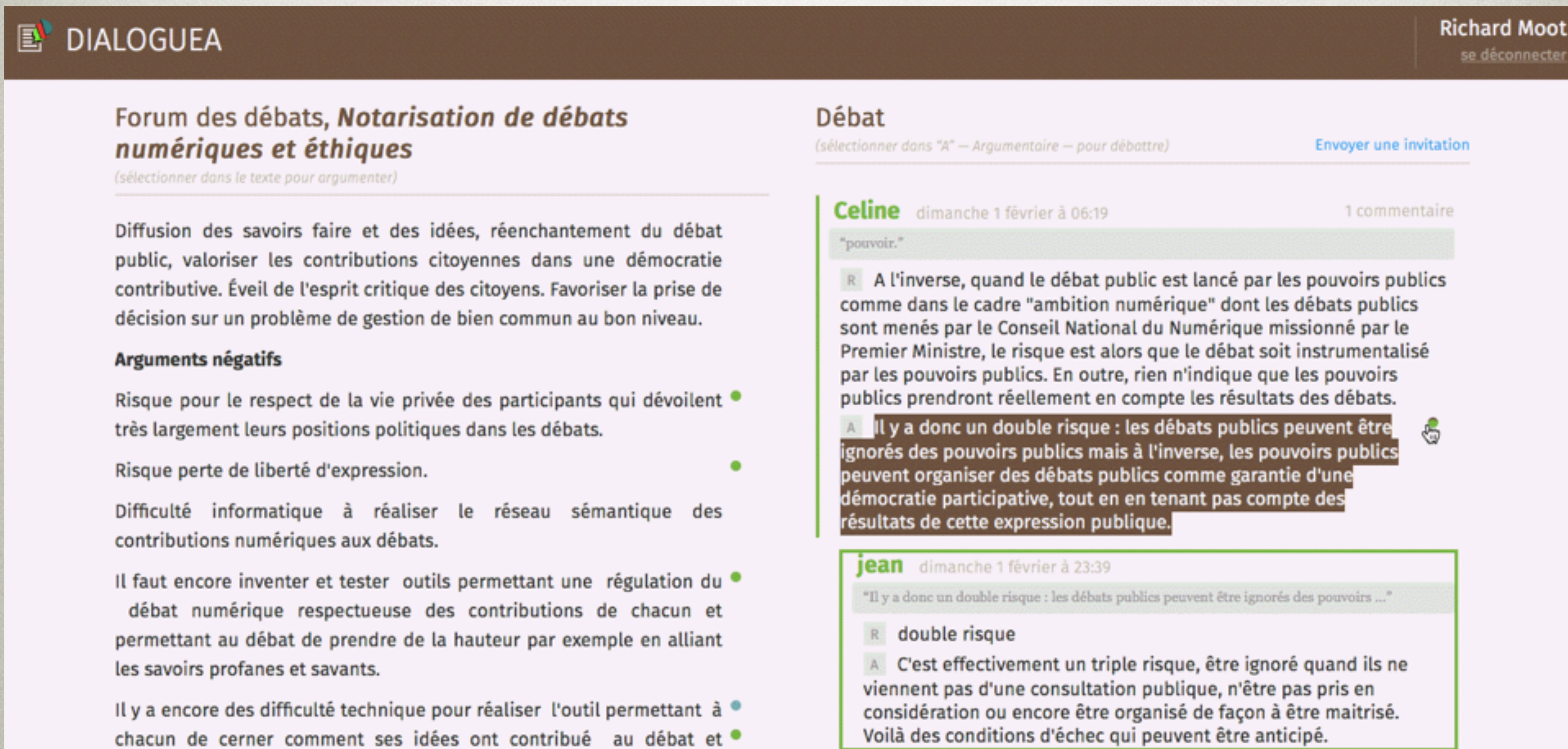
- Project AREN (started May 2016)
- How do we derive discourse relations, for example as used in SDRT?
- Can we automatically detect logical fallacies or turn an informal argument into a formal argument, for example by suggesting hidden premisses?

Dialoguea

- Platform for debates (dialoguea.fr)

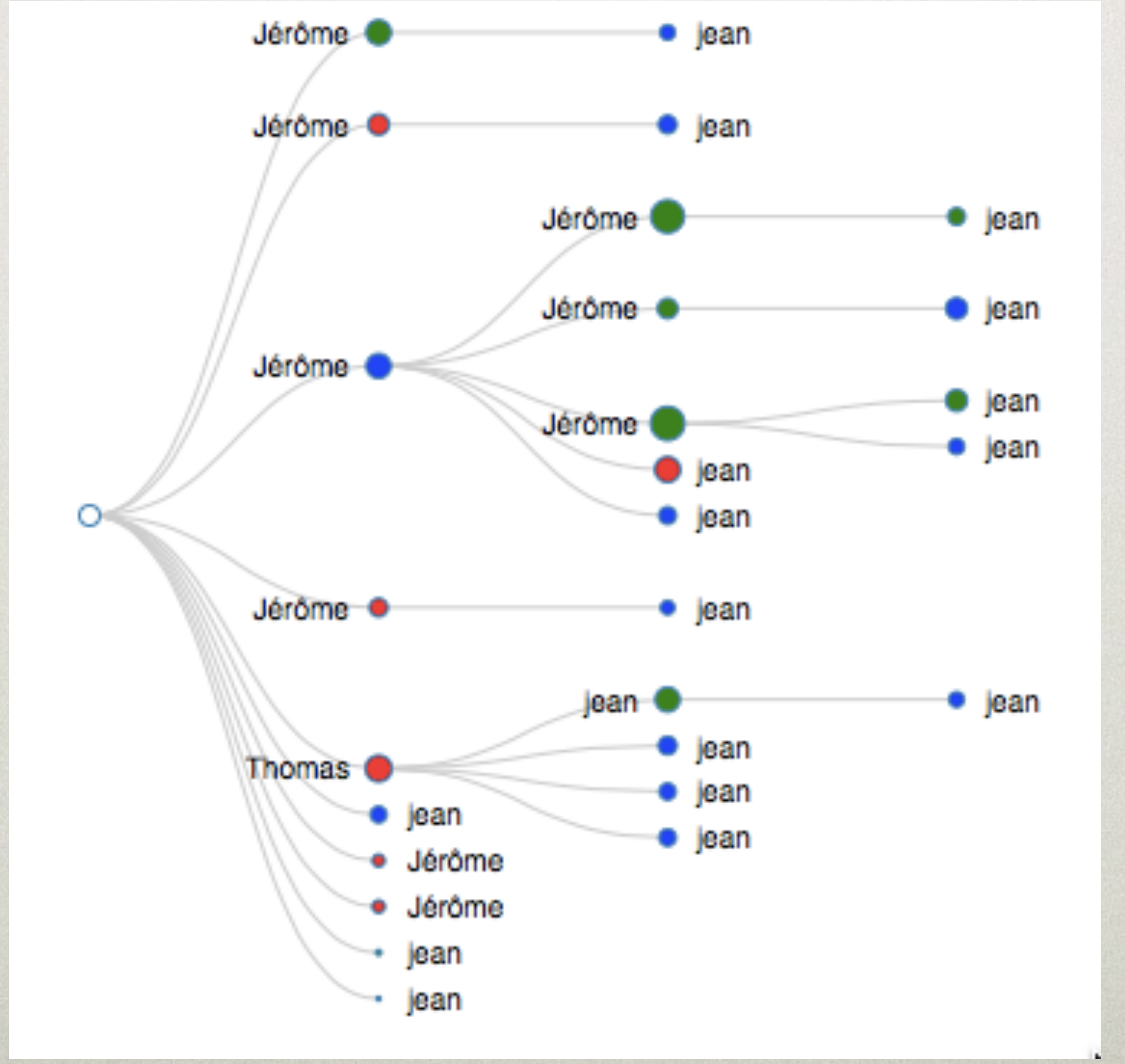
1. select sentence fragment
2. paraphrase it in your own words
3. note “agree”, “disagree” or “don’t understand”
4. argue your position

Dialoguea



https://dialoguea.fr

Dialoguea



Jérôme
jean
Jérôme
jean
Jérôme
jean
Jérôme
jean
Jérôme
Jérôme
jean
jean
jean
jean
Jérôme
jean
jean
jean
Thomas
jean
jean
jean
jean
Jérôme
Jérôme
jean
jean

Paraphrase

- The selected text should entail the paraphrase.
- If the paraphrase entails the text but not inversely then either:

1. the paraphrase clarifies/makes explicit intended meaning
2. it is not a paraphrase but a straw man

Agree, Disagree, Don't understand

- Not understanding is usually accompanied by a request for clarification (possibly giving question/answer pairs, but also possibly pointing to failed entailments)
- Can we determine agreement/disagreement from logical form only? Logical contradiction should imply disagreement

Logical Fallacies

- Failure of entailment can point to

1. missing axioms (eg. world knowledge),

2. error in computing the semantics,

or 3. fallacious reasoning

- Of course not all logical fallacies can be automatically detected

What is entailment?

A text T entails a hypothesis H if humans reading T would typically infer that H is most likely true

paraphrased from Dagan e.a. 2013

Entailment example

T

Google and NASA announced a working agreement, Wednesday, that could result in the Internet giant building a complex of up to 1 million square feet on NASA-owned property, adjacent to Moffett Field, near Mountain View.

H

Google may build a campus on NASA property

Entailment example

T Eating lots of foods that are a good source of fiber may keep your blood glucose from rising fast after you eat.

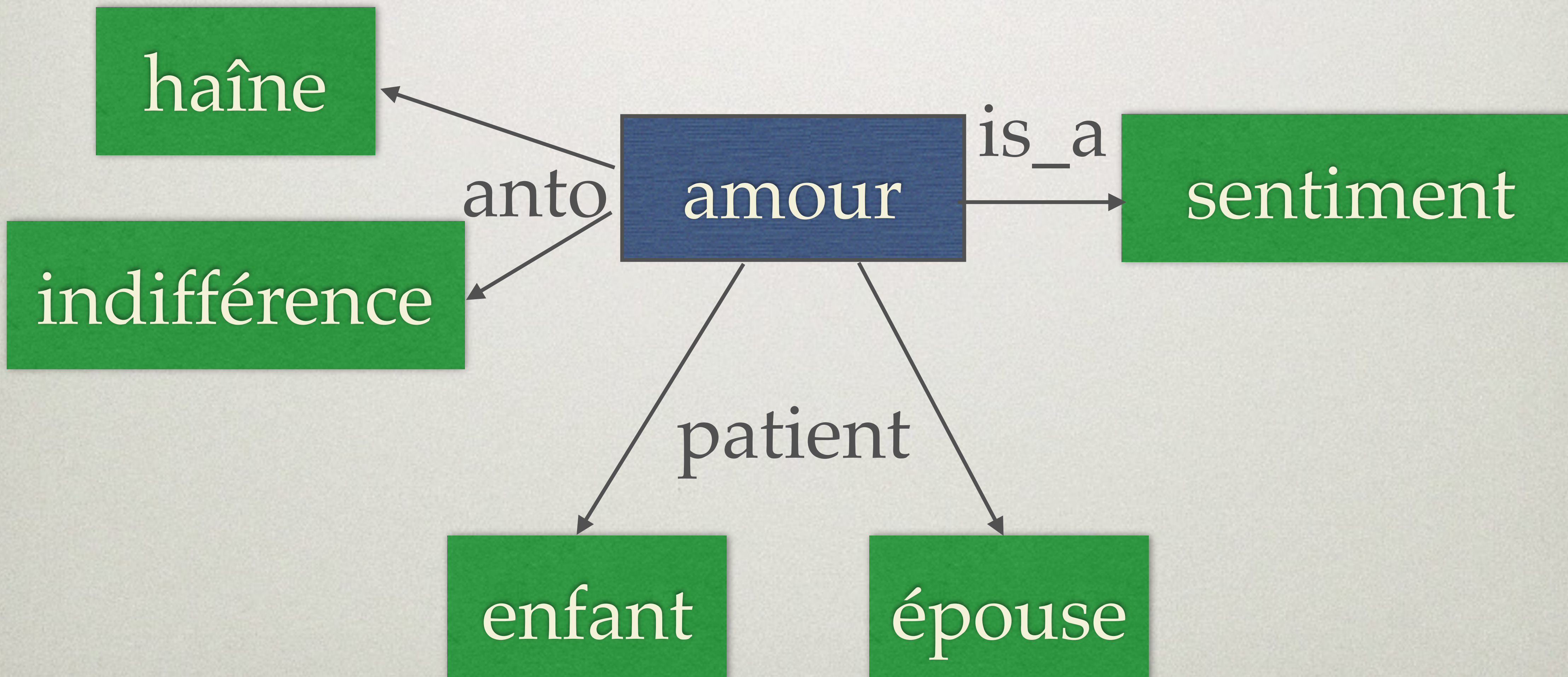
H Fiber improves blood sugar control.

What are atomic meanings?

- Montague's answer: atomic formulas (ie. the meaning of "*love*" is **love′**).
- Ontology answer: basic meanings are nodes in an ontology.
- Vector space answer: basic meanings are vectors in *n*-dimensional space.

What are atomic meanings?

- Ontology answer: basic meanings are nodes in an ontology.



http://www.jeuxdemots.org/jdm-accueil.php

What are atomic meanings?

- Vector space answer: basic meanings are vectors in *n*-dimensional space.

amour	1.0
tendresse	0.71
bonheur	0.67
amitié	0.59
platonique	0.57

http://fauconnier.github.io

Theorem Proving

- different choices of atomic meaning lead to different types reasoning/ theorem proving and, as a consequence, to different predictions with respect to entailment
- we explore each of the three options in turn

Vanilla Resolution

- Montague's solution corresponds to plain resolution theorem proving (or your preferred choice of theorem prover)
- We match *identical* positive and negative atomic formulas

Vanilla Resolution Theorem Proving

$$\frac{\begin{array}{l}\text{human}(X) \rightarrow \text{mortal}(X) \\ \text{human}(\text{socrates})\end{array}}{\text{mortal}(\text{socrates})}$$

1. translate to conjunctive normal form
2. add negation of the conclusion
3. derive a contradiction

Vanilla Resolution Theorem Proving

¬human(X) ∨ mortal(X)

human(socrates)

¬mortal(socrates)

Vanilla Resolution Theorem Proving

¬human(X) ∨ mortal(X)

human(socrates)

¬mortal(socrates)

mortal(socrates)

Drawbacks

- no world knowledge at all
- picky about exact predicates used
- not picky enough about the exact predicates used (eg. word sense disambiguation)

Resolution Theorem Proving with Ontology

treat ontology as set of logical claims:

- "p is_a q" corresponds to $p(X) \rightarrow q(X)$, or equivalently $\neg p(X) \vee q(X)$
- "p anto q" corresponds to $p(X) \rightarrow \neg q(X)$, or equivalently $\neg p(X) \vee \neg q(X)$

Resolution Theorem Proving with Ontology

$\neg$animal(X) $\vee$ mortel(X)

humain(socrates)

$\neg$mortel(socrates)

Resolution Theorem Proving with Ontology

$\neg$animal(X) ∨ mortel(X)

$\neg$mammifère(X) ∨ animal(X)

$\neg$humain(X) ∨ mammifère(X)

humain(socrates)

$\neg$mortel(socrates)

Resolution Theorem Proving with Ontology

$\neg$animal(X) ∨ mortel(X)

$\neg$mammifère(X) ∨ animal(X)

$\neg$humain(X) ∨ mammifère(X)

humain(socrates)

$\neg$mortel(socrates)

$\neg$mammifère(X) ∨ mortel(X)

Resolution Theorem Proving with Ontology

$\neg$animal(X) ∨ mortel(X)

$\neg$mammifère(X) ∨ animal(X)

$\neg$humain(X) ∨ mammifère(X)

humain(socrates)

$\neg$mortel(socrates)

$\neg$mammifère(X) ∨ mortel(X)

$\neg$humain(X) ∨ animal(X)

Advantages/Drawbacks

- some “light-weight” world knowledge
- different relations are useful in different situations “is_a”, “synonym”, “antonym”, “has_part” etc.
- can at least *identify* that some words have multiple senses

Resolution Theorem Proving with vectors

reasoning by similarity:

- find maximum-weight abduction for the statement
- standard resolution as a base case

Resolution Theorem Proving with vectors

$\neg$animal(X) $\vee$ mortel(X)

humain(socrates)

$\neg$mortel(socrates)

Resolution Theorem Proving with vectors

¬animal(X) ∨ mortel(X)

humain(socrates)

¬mortel(socrates)

¬vétérinaire(X) ∨ animal(X)

Advantages/Drawbacks

- weighted deduction/abduction
- computes "most similar" derivation, with standard resolution as a base case
- similarity has a *very* uneasy relation with derivability (though we may infer other weights à la Raina, Ng & Manning 2005)

Putting it all together

- Can we combine all these components?
- Use weights as preferences but use the ontology to reject combinations

AREN project

- The AREN project will provide us with numerous structured debates.
- These debates will contain potential entailment pairs and potential question/answer pairs.
- First experiments start soon

Conclusions

- Categorial grammars produce meanings in the tradition of Montague.
- I’ve presented some initial exploration of possible uses of wide-coverage semantics to entailment and to the analysis of structured debates.

Dependencies and Acknowledgements

- CNRS, ANR Polymnie and Région Aquitaine
- SWI Prolog (parser),
- TclTk (GUI),
- Clark & Curran (2004) tools (supertagger)
- LeFFF (lemmatizer)