Compréhension automatique du langage naturel

quelques applications récentes
des liens entre logique et langage naturel

Christian Retoré

Équipe Texte, LIRMM, Université de Montpellier

ÉNS Cachan

7 septembre 2016

I.1. Avant-propos

Merci à Jean Goubault de m'avoir invité.

Présentation partielle et partiale d'un domaine :

la linguistique informatique ou traitement automatique des langues

avec un point de vue **logique** (théorie des types, lambda calcul, système déductifs — j'aurais pu inclure la théorie des langages) qui ne sont pas les seules méthodes (approches quantitatives, statistiques)

Comment en suis-je arrivé là ? maîtrise de maths pures à Paris 6, DEA et thèse de maths à Paris 7 en logique, avec intérêt initial pour la littérature, le langage, la linguistique. Où ai-je développé cela ? Chercheur INRIA (Nancy, Rennes) puis prof à l'université de Bordeaux, puis à celle de Montpellier.

I.2. Plan

1. Présentation du traitement automatique des langues de la linguistique et des liens avec la logique.
2. De la phrase à des formules logiques représentant son ou ses sens. (Grammaires catégorielles, sémantique de Montague)
3. Extension :
   — sens des mots (sémantique lexicale, lexique génératif montagovien)
   — quantification en langage naturel et opérateurs de Hibert ε et τ.

II Traitement automatique des langues

II.1. Famille de problèmes à traiter

Deux tâches classiques (traduction= A puis B)

A analyse automatique (de texte, de parole, annoté ou non)
résultat ?

B génération automatique (de texte, de parole)
à partir de quoi ?

Domaine propre (recherche d'information) mais aussi utile pour A et B ci-dessus

— fouille, acquisition (des données pour les autre tâches ou RI)

II.2. Applications

Traduction automatique, aide à la traduction

> The flesh is weak but the spirit is willing.
> (russian)
> The meat is rotten but the vodka is strong.

Aide à la traduction : domaine spécifique, bonne représentation des connaissances, reformation, interaction avec l'utilisateur.

Correcteurs orthographiques (Word(Synapse) : vert : grammaires rouge : lexique)

(1) QuelS livreS crois-tu qu'il sait que je pense que tu as luS ?

Dialogue homme machine

(2) Quels sont les films des années cinquante qui passent actuellement à Bordeaux ?

(3) Les enfants prendront une pizza.

Résumé automatique

Réponse à des questions (question answering cf. text entailment)

Geach était il l'élève de Wittgenstein ?

Domaine relié : **recherche d'information** différence : Big Data, beaucoup de données, analyse superficielle, méthodes statistiques, machine learning

"ici, on ne vend pas de CD seulement des vinyles"

"production laitière / production de lait"
mais "production minière / production de mine(s)"

III Les niveaux d’analyse de la langue & leurs méthodes

III.1. Les sons : phonétique/phonologie

III.1.a Phonétique

Acoustique / système phonatoire/auditif

Traitement du signal / médecine

III.1.b Phonologie

Les sons abstraits : systéme discret (dans un continu)

(4) Bali / Paris indistincts pour un japonais

Automates et transducteurs

III.2. Les mots : morphologie

III.2.a Morphologie dérivationnelle

changement de catégorie possible

—

allumer/allumage

témoigner/témoignage

maquiller/maquillage

garer/garage

égarer/égarement

aménager/aménagement

—

maison /maisonnette

camion /camionnette

carpe/carpette

III.2.b Morphologie flexionnelle

pas de changement de catégorie (automates et transducteurs)

cheval → chevaux

aller → allons

aller → va

aller → irons

III.2.c Étiquetage grammatical

nom, verbe, déterminant etc.

Automates ou probas (modèles de Markov cachés)

III.3. Analyse de la phrase (arbre) : syntaxe

(5) Réparer fais les la le.

(6) *Je fais la réparer

(7) Je la fais réparer

(8) * [[Pierre [mange une]] pomme]

(9) Pierre [mange [une pomme]]

Grammaires formelles (CFG, TAG etc)

Déduction formelle (cf. ci-après)

Arbres satisfaisant des contraintes
(par ex. "le sujet précède le verbe")

III.4. Le sens : sémantique

III.4.a Le sens d’un mot : sémantique lexicale

Sens, restrictions de sélection, mots associés (logique, probabilités, graphes (petit monde), jeux)

> Rôle télique : être lu, informer, cultiver,
>
> Rôle constitutif : pages, couvertures
>
> Rôle agentif : imprimeur, auteur,

III.4.b Le sens d’une phrase : semantique compositionnelle, formelle

Phrase -> formule logique

Sens : ensemble des mondes possibles dans lequel la phrase est vraie

(ces 2 approches sont compatibles mais ne coïncident pas forcément)

III.5. L’interprétation en en contexte (discours, dialogue) : pragmatique

Prise en compte du contexte linguistique et extra linguistique

(10) a. Il est tombé. Quelqu’un l’a poussé.
b. "l"="il" + causalité

(11) a. Allons plutôt dans ce restaurant.
b. "nous" ? "ce" ?

III.6. Deux notions de sémantique

De quoi ça parle (analyse de texte, par des méthodes statistiques).

Qui fait quoi, ce qui est affirmé et réfuté (analyse de quelques phrases par des méthodes logiques).

> Geach était il l'élève de Wittgenstein ?
>
> Wikipédia : En 1941, IL épousa la philosophe Elizabeth Anscombe, grâce à LAQUELLE IL entra en contact avec Ludwig Wittgenstein. Bien qu'IL N'ait JAMAIS suivi l'enseignement académique de CE DERNIER, cependant IL EN éprouva fortement l'influence.

Dans cet exposé : syntaxe (déductive) et sémantique (logique) de la phrase et des mots.

IV Logique et langage : historique

IV.1. Syntaxe/grammaire et sémantique/logique 1

phrase = proposition qui peut être vrai ou fausse

lees syllogismes sont formulées en langage naturel.

Baroco

tout conducteur montpelliérain est pressé A
certains conducteurs ne sont pas pressés O
— donc —
certains conducteurs ne sont pas des conducteurs montpelliérains O

IV.2. Syntaxe/grammaire et sémantique/logique 2

verbe intransitif = prédicat à un argument (=sujet)

verbe transitif = prédicat à deux arguments

nom commun = prédicat

adjectif = propriété ou propriété de propriété ? ?

Les étudiants sont des musiciens.
$\rightarrow$ Les étudiants parisiens sont de(s) musiciens parisiens
$\not\rightarrow$ Les bons étudiants sont des bons musiciens.

IV.3. De re / de dicto (Thomas d'Aquin, Ockham)

(12) a. James Bond croit qu'un prof de l'ENS est un espion.

b. Il a vu Jean refermer son tiroir rapidement. (de re)

c. Il a trouvé un code sur un papier qui traînait. (de dicto)

IV.4. Sens/dénotation (Abélard,Frege)

(13) a. L'étoile du matin (Vénus).

b. L'étoile du berger. (Vénus)

IV.5. Compositionnalité du sens

Le sens du tout est fonction du sens des partie (Frege) et de la structure syntaxique (Montague) :

(14) a. Marie aime Pierre.
b. Pierre aime Marie.

(15) a. Mon voisin que je n'apprécie guère n'aime pas la musique.
b. Je n'apprécie guère la musique que mon voisin n'aime pas.

(16) Si [un paysan$_i$ possède un âne$_k$] alors [il$_i$ le$_k$ bat].

IV.6. Différences

La langue diffère de l'interprétation logique :

(17) a. J'avais trois clés dans ma poche, je les ai toutes perdues sauf une.
b. Je la range dans un tiroir.

(18) a. J'avais trois clés dans ma poche, j'en ai perdu deux.
b. * Je la range dans un tiroir.

V Quelques applications spécifiques

V.1. Exemple d’utilisation des méthodes formelles, logiques

Inférence textuelle (text entailment) :
une phrase est elle conséquence d’un (petit) texte ?

extraire un système de de préférence (pour l’aide à la décision).

Re-construire a structure logique d’un dialogue, d’un discours,..

Analyse = satisfaction de contraintes
(permet de traités d’énoncés dégradés)

V.2. Reconstruction d’itinéraires

Projet Itipy (INRIA et ex. régions Aquitaine et Midi Pyrénées)

Corpus XVII^{e} XX^{e} 576 334 mots (surtout XIX^{e})

Récits de voyages dans les Pyrénées.

Peut-on reconstruire les itinéraires ?

V.3. Examples ITIPY

Nous coupons ici un sentier qui vient du port de Barroude (...)

Plus loin, de nobles hêtres montent sur le versant (...) cette route qui monte sans cesse pendant deux lieues

Le chemin pavé de calcaire et de pierres luisantes (...) serpente à travers fourrés de buis et de noisetiers

Puis, cinq minutes nous conduisent à un petit pont (...) qui nous porte sur la rive droite.

Problème : parenthèses ouvertes (comparaison avec d'autres voyages) mais dont la fermeture est peu claire (même pour un être humain).

V.4. Projet AREN Argumentation et numérique 1

Université (info, pédagogie) + Académie de Montpellier + 3 start up + 2 associations

Plateforme de débats en ligne écrit par des classes de lycées sur des sujets interdisciplinaires à partir de synthèses scienfique (ControverSciences asso ENS Lyon).

ex : neuroéthique (SVT / philo)

1. surlignage d'un extrait du texte ou d'une intervention précédente
2. re formulation
3. choix (d'accord — pas d'accord — je ne comprends pas)
4. justification argumentée de quelque lignes

V.5. Projet AREN Argumentation et numérique 2

Analyse automatique :

correction de la reformulation,

trouver les arguments (par opposition à des émotions),

nature des arguments,

comment se répondent ils ?

Structure du débat (S DRT graphe étiqueté avec une structure récursive) –> Outil de navigation dans le débat ;

Restitution en classe, explications etc.

V.6. Spéculatif : analyse de la créativité poétique

Pas la génération de poèmes (Oulipo)

Pas de statistiques sur les mots, les phonèmes etc mais plutôt une analyse logique standard qui va bloquer souvent en précisant la nature du blocage.

Poésie : presque pas de contexte extra linguistique

Image poétique réside dans ce qui résiste à l’analyse (par ex. métaphore : composition inattendue)

Par exemple :
"La secrète noirceur du lait..."
Jacques Audiberti cité par J.-Y. Girard dans "linear logic"

V.7. Ensuite (= après la pause)

Phrase –> formule logique

1. Phrase
2. Analyse syntaxique dans une grammaire catégorielle (preuve dans une logique particulière)
3. Conversion en lambda terme linéaire de type proposition
4. Insertion du sens des mots
5. Beta réduction
6. Formule logique = sens de la phrase

(19) a. Les enfants prendront une pizza.

b. $\exists x.\ \mathit{pizza}(x) \wedge \forall w.\ (\mathit{enfant}(w) \Rightarrow \mathit{prendront}(w, x))$

c. $\forall u.\ \mathit{enfants}(u) \Rightarrow \exists .x\ \mathit{pizza}(x) \wedge \mathit{prendront}(u, x)$

Puis : comment prendre en compte le sens lexical.
Et finalement : conclusion.

A Categorial Grammars

A.1. What are categorial grammars?

- A *lexicon* mapping words to (small) sets of formulas
- A *logic* specifying the meaning and the behaviour of the logical connectives

Universal grammar is a logic. Language variation is restricted to the lexicon.

A.2. AB grammars

Not a logic (yet!) but the foundation of categorial grammars.

A.3. Atomic formulas

s (sentence),

np (noun phrase), for example: John, the tall student

n (noun), for example: student, book, ...

Maybe some others: *pp* (for prepositional phrases), *inf* (for infinitival phrases), ...

Goal: all grammatical sentence should be derivable as being of category *s* (in a sense we will make precise).

A.4. Formulas

Formulas are inductively defined as follows.

- Atomic formulas are formulas.
- If A and B are formulas, then (A/B) (we say A *over* B) and $(B\backslash A)$ (we say B *under* A) are formulas.

Intuition: a formula of the form A/B combines with a B to its *right* to form an A, a formula $B\backslash A$ combines with a B to its *left* to form an A.

A.5. Example formulas, example lexicon (strict)

The following are formulas: (np/n), $(np\backslash s)$, $((np\backslash s)/np)$, $((n\backslash n)/(np\backslash s))$

$$
\begin{aligned}
Lex(the) &= \{\,(np/n)\,\}\\
Lex(an) &= \{\,(np/n)\,\}\\
Lex(president) &= \{\,n\,\}\\
Lex(actress) &= \{\,n\,\}\\
Lex(likes) &= \{\,((np\backslash s)/np)\,\}
\end{aligned}
$$

A.6. Example formulas, example lexicon (sloppy)

The following are formulas: np/n, $np\backslash s$, $(np\backslash s)/np$, $(n\backslash n)/(np\backslash s)$

$$\begin{aligned} Lex(the) &= np/n \\ Lex(an) &= np/n \\ Lex(president) &= n \\ Lex(actress) &= n \\ Lex(likes) &= (np\backslash s)/np \end{aligned}$$

A.7. AB grammars: rules

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

A.8. AB grammars: rules

$$\frac{A/B \quad B}{A}\,[/E] \qquad\qquad \frac{\dfrac{\textit{the}}{np/n} \quad \dfrac{\textit{president}}{n}}{np}\,[/E]$$

$A = np$, $B = n$

A.9. AB grammars: rules

$$\frac{A/B \quad B}{A}\,[/E] \qquad\qquad \frac{\dfrac{an}{np/n} \quad \dfrac{actress}{n}}{np}\,[/E]$$

$A = np$, $B = n$

A.10. AB grammars: rules

$$\frac{A/B \quad B}{A}\,[/E] \qquad\qquad \frac{\dfrac{likes}{(np\backslash s)/np} \quad \dfrac{\dfrac{an}{np/n} \quad \dfrac{actress}{n}}{np}\,[/E]}{np\backslash s}\,[/E]$$

$A = np\backslash s$, $B = np$

A.11. AB grammars: rules

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

$$\begin{array}{c} \textit{likes an actress} \\ \vdots \\ np\backslash s \end{array}$$

$B = np$, $A = s$

A.12. AB grammars: rules

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

the president likes an actress

$$\frac{np \qquad\qquad np\backslash s}{s}\,[\backslash E]$$

$B = np$, $A = s$

A.13. Modifiers

1. A student slept.
2. A student slept in class.
3. A student slept in class during the exam.
4. A student slept in class during the exam yesterday at 15h while snoring.

"*in class*" modifies a sentence s and is therefore assigned the formula $s\backslash s$ (or if you prefer, the vp modifier $(np\backslash s)\backslash(np\backslash s)$).

"*class*" is a noun n, therefore a lexical possibility for "*in*" should be $(s\backslash s)/n$ or $((np\backslash s)\backslash(np\backslash s))/n$.

A.14. Relative phrases

The	*student*	*who*	*slept*
np/n	n	$(n\backslash n)/(np\backslash s)$	$np\backslash s$

The	*student*	*whom*	*the professor*	*woke*
np/n	n	$(n\backslash n)/(s/np)$	np	$(np\backslash s)/np$

A.15. Context free grammars

A context-free grammar is defined by:

[Non Terminals] a set NT of symbols called non terminals, one of them being the start symbol S.

[Terminals] set T of symbols, disjoint from NT, called terminals (or words according to the linguistic viewpoint)

[Production rules] a finite set of production rules of the form $X \rightarrow W$ with $X \in NT$ and $W \in (T \cup NT)^*$

A.16. Context free grammars

A context free grammars is said to be:

- in strong Greibach normal form when all rules are $X \to a$ or $X \to aY$ or $X \to aYZ$, with $a \in T$ and $X, Y, Z \in NT$
- in Chomsky normal form when all rules are $X \to a$ or $X \to YZ$ with $a \in T$ and $X, Y, Z \in NT$

Any context free grammar can be turned into an a grammar of both normal forms, both generating the same language.

A.17. From AB grammars to context free grammars

Given an AB grammar, there exists a context free grammar (in Chomsky normal form) that generates the same language.

Take all categories and subcategories from the lexicon as non terminals, add rules:

$Y \to X\ (X \backslash Y)$,

$Y \to (Y/X)\ X$ and

$X \to a$ whenever $a : X$

A.18. From context free grammars to AB grammars

Given a context free grammar, which can be assumed to be in Greibach normal form, there exists an AB grammar that generates the same language.

$X \to a$ becomes $a : X$

$X \to aY$ becomes $a : X/Y$

$X \to aYZ$ becomes $a : (X/Z)/Y$

A.19. Lambek grammars — Natural deduction

$$\frac{A/B \quad B}{A}\,[/E] \qquad\qquad \frac{\begin{array}{c}\ldots [B]^n \\ \vdots \\ A\end{array}}{A/B}\,[/I]^n$$

$$\frac{B \quad B\backslash A}{A}\,[\backslash E] \qquad\qquad \frac{\begin{array}{c}[B]^n \ldots \\ \vdots \\ A\end{array}}{B\backslash A}\,[\backslash I]^n$$

Conditions: $[B]$ is the rightmost (for $/I$) resp. leftmost (for $\backslash I$) undischarged hypothesis *and* the proof has another undischarged hypothesis. $[B]$ is discharged after application of the rule.

A.20. A very interesting book

a	*very*	*interesting*	*book*
np/n	$(n/n)/(n/n)$	n/n	n

A.21. *A very book

$$
\begin{array}{ccc}
\textit{a} & \textit{very} & \textit{book} \\
np/n & (n/n)/(n/n) & n
\end{array}
$$

A.22. Relations between formulas

What is the relation between a sentence modifier $s\backslash s$ and a vp modifier $(np\backslash s)\backslash(np\backslash s)$?

Let's try to prove $s\backslash s$ from $(np\backslash s)\backslash(np\backslash s)$ and vice versa.

A.23. Relatives

The	*student*	*whom*	*the professor*	*woke*
np/n	n	$(n\backslash n)/(s/np)$	np	$(np\backslash s)/np$

np trace $\approx$ hypothetical np

movement $\approx$ introduction rule

A.24. Coordination

- John and Mary left.
  ($\mathit{and} = (np\backslash np)/np$)
- John left and slammed the door.
  ($\mathit{and} = ((np\backslash s)\backslash(np\backslash s))/(np\backslash s)$)
- John understood and implemented Dijkstra's algorithm.
  ($\mathit{and} = (((np\backslash s)/np)\backslash((np\backslash s)/np))/((np\backslash s)/np)$)

A.25. Coordination — Non-constituent coordination

- John loves but Mary hates Noam.
  ($\mathit{and} = ((s/np)\backslash(s/np))/(s/np)$)
- John has understood and will implement Dijkstra's algorithm.
  ($\mathit{and} = (((np\backslash s)/np)\backslash((np\backslash s)/np))/((np\backslash s)/np)$)

A.26. Quantifiers

Everyone	*wants*	*a*	*GoT-wedding*
$s/(np\backslash s)$	$(np\backslash s)/np$	$((s/np)\backslash s)/n$	n

Exercise: try with the more interesting (and better!) formula $(np\backslash s)/((s/np)\backslash s)$ for *wants*

A.27. Quantifiers and Coordination

Harry	*and*	*all students*	*complained*
np	$(X\backslash X)/X$	$s/(np\backslash s)$	$np\backslash s$

A barber	*shaves*	*everyone*	*but*	*himself*
$s/(np\backslash s)$	$(np\backslash s)/np$	$(s/np)\backslash s$	$(Y\backslash Y)/Y$	$((np\backslash s)/np)\backslash(np\backslash s$

A.28. Coordination: Problems

Though the standard theory of coordination in the Lambek calculus works well, even for many cases which have puzzled researchers in other formalisms, there are some well-known problem cases.

1. book which John read Great Expectations and Mary loves
2. The mother of and John thinks that Mary left.

A.29. Models for the Lambek calculus

Particular models:

data: a free monoid M

base category $[p]$ arbitrary part of M

$$[F\backslash G] = \{u | \forall x \in [F] xu \in [G]\}$$
$$[G/F] = \{u | \forall x \in [F] ux \in [G]\}$$

$A_1, \ldots, A_n \vdash X$ is said to be valid in an interpretation if $[A_1] \cdots [A_n] \subset [X]$ where $A \cdot B = \{xy | x \in A, y \in B\}$.

Soundness: Every derivable sequent is valid in any interpretation.

Method: by induction on the derivation.

Completeness: A sequent that is valid in any interpretation is derivable.

Method: consider the monoid of sequences of categories a.k.a contexts.

Given a formula F let us call $Ctx(F) = \{\Gamma | \Gamma \vdash F \text{is provable}\}$

Observe that $G \in Ctx(G)$ for any G.

Interpret any atom p by $[p] = Ctx(p)$.

Lemma: for any formulae $Ctx(F) = [F]$ as defined above.

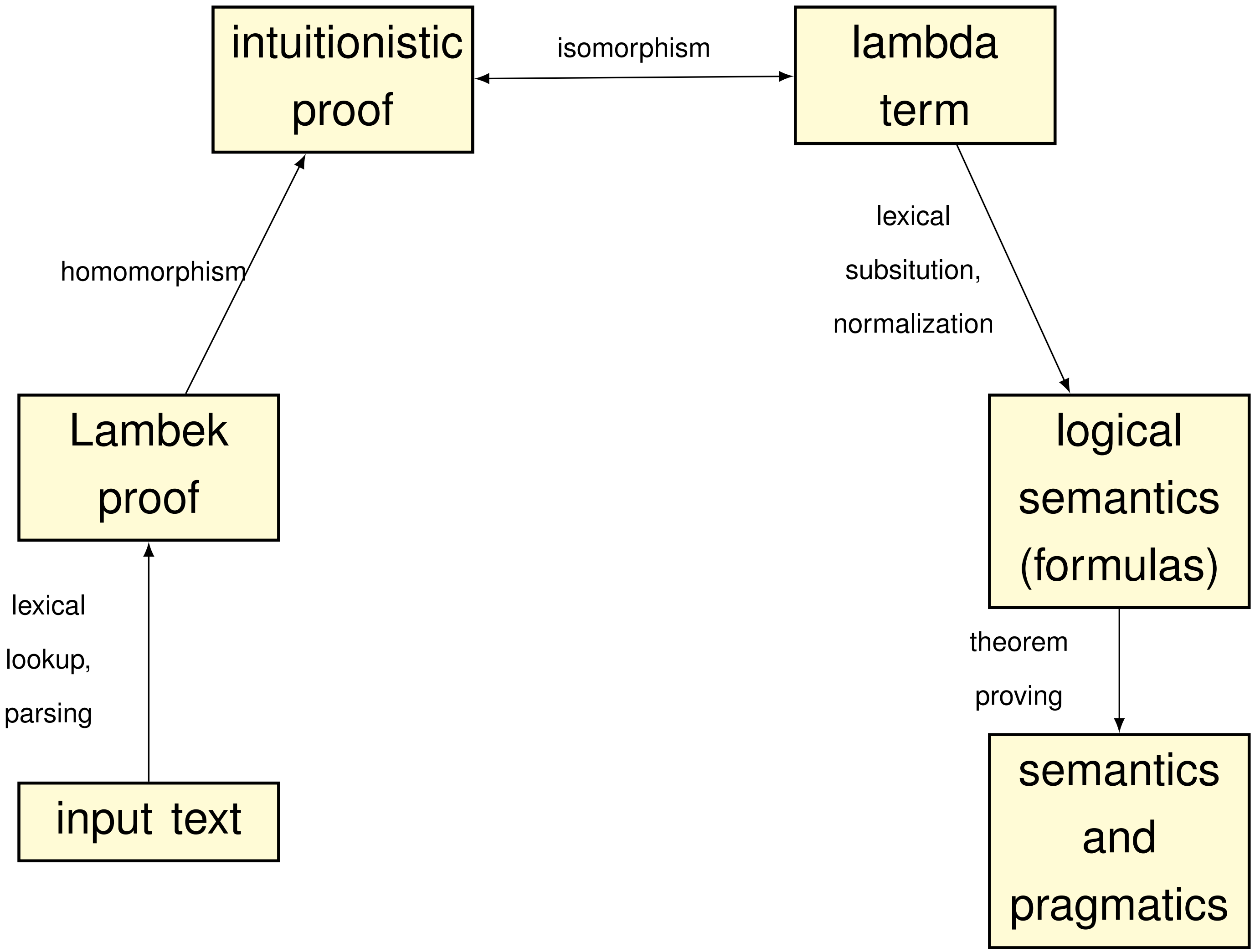
Completeness follows, since $X \in Ctx(X) = [X]$ for all X

C Montague Semantics

C.1. Overview

- Montague Grammar and the simply typed lambda calculus (reminder)
- Curry-Howard formulas-as-types interpretation
- Montague semantics for the Lambek calculus

C.2. Architecture



	Introduction rules	Elimination rules
Intuitionistic	$\dfrac{\begin{array}{c}[A]^n\\ \vdots\\ B\end{array}}{A \to B} \to I_n$	$\dfrac{A \quad A \to B}{B} \to E$
Lambek	$\dfrac{\begin{array}{c}[A]^n \ldots\\ \vdots\\ B\end{array}}{A \backslash B} \backslash I_n$ $\dfrac{\begin{array}{c}\ldots [A]^n\\ \vdots\\ B\end{array}}{B / A} / I_n$	$\dfrac{A \quad A \backslash B}{B} \backslash E$ $\dfrac{B / A \quad A}{B} / E$

C.3. Types and terms: Curry-Howard

A proof of $A \to B$ is a function that maps proofs of A to proofs of B.

Think of a formula/type as the set of its proofs.

Types are.... formulae.

λ-terms encode proofs $u : U$ means u is a term of type U.

We will also write $u : U$ as u^U.

C.4. Terms: Curry-Howard

1. *hypotheses* variables of each type which are terms of this type
2. *constants* there can be constants of each type
3. *abstraction* if $x : U$ is a **variable** and $t : T$ then $(\lambda x^U.\, t) : U \to V$.
4. *application* if $f : U \to V$ and $t : U$ then $(f\, t) : V$

With such typed terms we can faithfully encode proofs.

Variables are hypotheses (that are simultaneously cancelled).

C.5. Reduction and Normalisation

Reduction: $(\lambda x : U.\ t)^{U \to V} u^{U}$ reduces to $t[x := u] : V$.

Every simply typed lambda term reduces to a unique normal form, regardless the reduction strategy used.

C.6. Representing formulae within lambda calculus — connectives

Assume that the base types are $\mathbf{e}$ and $\mathbf{t}$ and that the only constants are

We need the following logical constants:

Constant	Type
$\exists$	$(e \to t) \to t$
$\forall$	$(e \to t) \to t$
$\wedge$	$t \to (t \to t)$
$\vee$	$t \to (t \to t)$
$\supset$	$t \to (t \to t)$

C.7. Representing formulae within lambda calculus — language constants

The language constants for First Order Logic (for a start):

- R_q of type $\mathbf{e} \to (\mathbf{e} \to (\ldots . \to \mathbf{e} \to \mathbf{t}))$
  e.g. likes: $e \to e \to t$, sleeps $e \to t$
- f_q of type $\mathbf{e} \to (\mathbf{e} \to (\ldots . \to \mathbf{e} \to \mathbf{e}))$

C.8. Formulae and normal lambda terms

Proposition 4 *A normal lambda-term of type* $\mathbf{t}$ *using only the constants given above corresponds to a formula of first-order logic.*

C.9. Example: From formulae to normal lambda terms

$$\forall x.\, barber(x) \supset shaves(x, x)$$

$$\forall(\lambda x^{\mathbf{e}}.\ (\supset barber(x))((shaves(x))(x)))$$

Another one?

Detailed examples: a FOL formula as a term and as a natural deduction proof.

C.10. For Montague semantics

Non normal lambda terms of type $\mathbf{t}$ coming from syntax do not really correspond to formulae.

Hence we need:

- normalisation
- a proof that the normal terms do correspond to formulae, as we just shown.

C.11. Montague semantics. Types.

Simply typed lambda terms

$$types ::= e \mid t \mid types \rightarrow types$$

chair , *sleep* $e \rightarrow t$

likes transitive verb $e \rightarrow (e \rightarrow t)$

C.12. Montague semantics: Syntax/semantics.

(Syntactic type)*	=	Semantic type	
s^*	=	t	a sentence is a proposition
np^*	=	e	a noun phrase is an entity
n^*	=	$e \rightarrow t$	a noun is a subset of the set of entities
$(A\backslash B)^* = (B/A)^*$	=	$A \rightarrow B$	extends easily to all syntactic categories of a Categorial Grammar e.g. a Lambek CG

Logical operations (and, or, some, all the,.....) are the lambda-term constants defined above.

C.13. Montague semantics
Logic within lambda-calculus

Words in the lexicon need constants for their denotation:

likes	$\lambda x \lambda y\ (\mathrm{likes}\ y)\ x$	$x : e,\ y : e, \mathrm{likes} : e \rightarrow (e \rightarrow t)$
<< likes >> is a two-place predicate		
Garance	$\lambda P\ (P\ \mathrm{Garance})$	$P : e \rightarrow t,\ \mathrm{Garance} : e$
<< Garance >> is viewed as the properties that << Garance >> holds		

C.14. Montague semantics. Computing the semantics 1/5

1. Replace in the lambda-term issued from the syntax the words by the corresponding term of the lexicon.

2. Reduce the resulting λ-term of type t to obtain its normal form, which corresponds to a logical formula, the "meaning".

word	***syntactic type*** u
	semantic type u^*
	semantics: λ***-term of type*** u^*
	x^v ***means that the variable or constant*** x ***is o***
some	$(s/(np\backslash s))/n$
	$(e \to t) \to ((e \to t) \to t)$
	$\lambda P^{e\to t}\ \lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}(P\ x)(Q$
statements	n
	$e \to t$
	$\lambda x^e(\mathtt{statement}^{e\to t}\ x)$
speak_about	$(np\backslash s)/np$
	$e \to (e \to t)$
	$\lambda y^e\ \lambda x^e\ ((\mathtt{speak\_about}^{e\to(e\to t)}\ x)\ y)$
themselves	$((np\backslash s)/np)\backslash(np\backslash s)$
	$(e \to (e \to t)) \to (e \to t)$
	$\lambda P^{e\to(e\to t)}\ \lambda x^e\ ((P\ x)\ x)$

C.15. Syntactic proof

Let us first show that "*Some statements speak about themselves*" belongs to the language generated by this lexicon. So let us prove (in natural deduction) the following:

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

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

C.16. Syntactic Proof to Semantic proof

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

Using the homomorphism from syntactic types to semantic types we obtain the following intuitionistic deduction.

$$\dfrac{\dfrac{(e\to t)\to(e\to t)\to t \quad e\to t}{(e\to t)\to t}\,{\to}E \qquad \dfrac{e\to e\to t \quad (e\to e\to t)\to e\to t}{e\to t}\,{\to}E}{t}\,{\to}E$$

C.17. Semantic Proof to Lambda Term

$$\cfrac{\cfrac{(e \to t) \to (e \to t) \to t \qquad e \to t}{(e \to t) \to t}\,{\to}E \qquad \cfrac{e \to e \to t \qquad (e \to e \to t) \to e \to t}{e \to t}\,{\to}E}{t}\,{\to}E$$

$$\cfrac{\cfrac{So^{(e\to t)\to(e\to t)\to t} \qquad Sta^{e\to t}}{\left(So\ \ Sta\right)^{(e\to t)\to t}}\,{\to}E \qquad \cfrac{SpA^{e\to e\to t} \qquad Refl^{(e\to e\to t)\to e\to t}}{\left(Refl\ \ SpA\right)^{e\to t}}\,{\to}E}{\left(\left(So\ \ Sta\right)\ \left(Refl\ \ SpA\right)\right)^{t}}\,{\to}E$$

C.18. Montague semantics. Computing the semantics. 3/5

The syntax (e.g. a Lambek categorial grammar) yields a λ-term representing this deduction simply is

((some statements) (themselves speak_about)) of type t

C.19. Montague semantics. Computing the semantics. 4/5

$$\Bigg(\Big(\lambda P^{e\to t}\,\lambda Q^{e\to t}\,(\exists^{(e\to t)\to t}\,(\lambda x^{e}(\wedge(P\;x)(Q\;x)))))\Big)$$

$$\Big(\lambda x^{e}(\mathtt{statement}^{e\to t}\;x)\Big)\Bigg)$$

$$\Bigg(\Big(\lambda P^{e\to(e\to t)}\;\lambda x^{e}\;((P\;x)x)\Big)$$

$$\Big(\lambda y^{e}\;\lambda x^{e}\;((\mathtt{speak\_about}^{e\to(e\to t)}\;x)y)\Big)\Bigg)$$

$$\downarrow\beta$$

$$\Big(\lambda Q^{e\to t}\;(\exists^{(e\to t)\to t}\;(\lambda x^{e}(\wedge^{t\to(t\to t)}(\mathtt{statement}^{e\to t}\;x)(Q\;x))))\Big)$$

$$\Big(\lambda x^{e}\;((\mathtt{speak\_about}^{e\to(e\to t)}\;x)x)\Big)$$

$$\downarrow\beta$$

$$\Big(\exists^{(e\to t)\to t}\;(\lambda x^{e}(\wedge(\mathtt{statement}^{e\to t}\;x)((\mathtt{speak\_about}^{e\to(e\to t)}\;x)x)))\Big)$$

C.20. Montague semantics. Computing the semantics. 5/5

This term represent the following formula of predicate calculus (in a more pleasant format):

$$\exists x : e\ (\texttt{statement}(x) \wedge \texttt{speak\_about}(x, x))$$

This is a (simplistic) semantic representation of the analysed sentence.

C.21. Autre exemple, en français cette fois

C.22. Sémantique compositionnelle : principes

Le sens d'une expression composée est fonction du sens de ses parties (Frege) et de leur assemblage syntaxique (Montague)

(Catégorie syntaxique)*	$=$	Type sémantique	
S^*	$=$	t	une phrase est une proposition
np^*	$=$	e	un groupe nominal est une entité/individu
n^*	$=$	$e \rightarrow t$	un nom commun est une propriété des entités
$(A\backslash B)^* = (B/A)^*$	$=$	$A \rightarrow B$	propage la traduction

C.23. Des constantes pour les opérations logiques

Constant	Type
$\exists$	$(e \rightarrow t) \rightarrow t$
$\forall$	$(e \rightarrow t) \rightarrow t$
$\wedge$	$t \rightarrow (t \rightarrow t)$
$\vee$	$t \rightarrow (t \rightarrow t)$
$\supset$	$t \rightarrow (t \rightarrow t)$

C.24. Des constantes pour les prédicats du langage

La dénotation des mots requiert des prédicats :

aime	$\lambda x \lambda y\ (\mathrm{aime}\ y)\ x$	$x : e,\ y : e, \mathrm{aime} : e \rightarrow (e \rightarrow t)$
« aime » est un prédicat binaire		
Garance	$\lambda P\ (P\ \mathrm{Garance})$	$P : e \rightarrow t,\ \mathrm{Garance} : e$
« Garance » est décrite comme les propriétés de « Garance »		

C.25. Sémantique à la Montague : algorithme

1. analyse syntaxique preuve de S
2. conversion en lambda terme de type t sur e et t
3. insertion des lambda terme lexicaux (même type)
4. réduction

   $\rightarrow$ terme de type t

   = formule logique

   = sens de la phrase analysée

C.26. Exemple de calcul sémantique : les enfants prendront une pizza

mot	***catégorie syntaxique*** u ***type sémantique*** u^* ***sémantique :*** λ***-term of type*** u^* x^v ***signifie*** x ***(variable, constante) de type*** v
les	$(S/(np\backslash S))/n$ (subject) $((S/np)\backslash S)/n$ (object) $(e \to t) \to ((e \to t) \to t)$ $\lambda P^{e\to t}\ \lambda Q^{e\to t}\ (\forall^{(e\to t)\to t}\ (\lambda x^e(\Rightarrow^{t\to(t\to t)}\ (P\ x)(Q\ x))))$
une	$((S/np)\backslash S)/n$ (object) $(S/(np\backslash S))/n$ (subject) $(e \to t) \to ((e \to t) \to t)$ $\lambda P^{e\to t}\ \lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}(P\ x)(Q\ x))))$
enfant(s)	n $e \to t$ $\lambda x^e(\mathtt{enfant}^{e\to t}\ x)$
pizza	n $e \to t$ $\lambda x^e(\mathtt{pizza}^{e\to t}\ x)$
prendront	$(np\backslash S)/np$ $e \to (e \to t)$ $\lambda y^e\ \lambda x^e\ ((\mathtt{prendront}^{e\to(e\to t)}\ x)y)$

C.27. Analyse syntaxique ∃∀

Il y a deux analyse syntaxique possibles. Une :

∃∀

$$
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{(S/(np\backslash S))/n \quad n}{(S/(np\backslash S))}\,/_e
      \qquad
      \dfrac{(np\backslash S)/np \quad [np]^1}{(np\backslash S)}\,/_e
    }{S}\,/_e
  }{S/np}\,/_i(1)
  \qquad
  \dfrac{((S/np)\backslash S)/n \quad n}{(S/np)\backslash S}\,\backslash_e
}{S}\,\backslash_e
$$

C.28. Syntaxe $\rightarrow$ λ-terme sémantique de la phrase

$\exists\forall$

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\dfrac{\overset{\textit{les}}{(\mathbf{e}\to\mathbf{t})\to(\mathbf{e}\to\mathbf{t})\to\mathbf{t}} \qquad \overset{\textit{enfants}}{(\mathbf{e}\to\mathbf{t})}}{(\mathbf{e}\to\mathbf{t})\to\mathbf{t}}\;\to_e
\qquad
\dfrac{\overset{\textit{prendront}}{\mathbf{e}\to\mathbf{e}\to\mathbf{t}} \qquad \overset{o}{[\mathbf{e}]^1}}{\mathbf{e}\to\mathbf{t}}\;\to_e
}{\mathbf{t}}\;\to_e
}{\mathbf{e}\to\mathbf{t}}\;\to_i(1)
\qquad
\dfrac{\overset{\textit{une}}{(\mathbf{e}\to\mathbf{t})\to(\mathbf{e}\to\mathbf{t})\to\mathbf{t}} \qquad \overset{\textit{pizza}}{(\mathbf{e}\to\mathbf{t})}}{(\mathbf{e}\to\mathbf{t})\to\mathbf{t}}\;\to_e
}{\mathbf{t}}\;\to_e
}{}
$$

Le λ-terme correspondant est :

$\exists\forall = (\textit{une pizza})(\lambda o^{\mathbf{e}}(\textit{les enfants})(\textit{prendront } o))$

Il faut encore :

1. insérer les lambda terme lexicaux et
2. réduire/calculer

C.29. Calculs, par étapes 1/2

$(\textit{une pizza})$
$= (\lambda P^{e\to t}\ \lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}(P\ x)(Q\ x))))) (\lambda z^e(\texttt{pizza}^{e\to t}\ z))$
$= (\lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\lambda z^e(\texttt{pizza}^{e\to t}\ z))\ x)(Q\ x)))))$
$= (\lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\texttt{pizza}^{e\to t}\ x)))(Q\ x))))$

$(\textit{les enfants})$
$= (\lambda P^{e\to t}\ \lambda Q^{e\to t}\ (\forall^{(e\to t)\to t}\ (\lambda x^e(\Rightarrow^{t\to(t\to t)}\ (P\ x)(Q\ x))))) (\lambda u^e(\texttt{enfant}^{e\to t}\ u))$
$= (\lambda Q^{e\to t}\ (\forall^{(e\to t)\to t}\ (\lambda x^e(\Rightarrow^{t\to(t\to t)}\ ((\lambda u^e(\texttt{enfant}^{e\to t}\ u))\ x)(Q\ x)))))$
$= (\lambda Q^{e\to t}\ (\forall^{(e\to t)\to t}\ (\lambda x^e(\Rightarrow^{t\to(t\to t)}\ (\texttt{enfant}^{e\to t}\ x)(Q\ x)))))$

$(\textit{les enfants})(\textit{prendront } o) =$
$(\lambda Q^{e\to t}\ (\forall^{(e\to t)\to t}\ (\lambda w^e(\Rightarrow^{t\to(t\to t)}\ (\texttt{enfant}^{e\to t}\ w)(Q\ w))))) ((\lambda y^e\ \lambda x^e\ ((\texttt{prendront}^{e\to(e\to t)}\ x)$
$= (\lambda Q^{e\to t}\ (\forall^{(e\to t)\to t}\ (\lambda w^e(\Rightarrow^{t\to(t\to t)}\ (\texttt{enfant}^{e\to t}\ w)(Q\ w))))) (\lambda x^e\ ((\texttt{prendront}^{e\to(e\to t)}\ x)\ o))$
$= \forall^{(e\to t)\to t}\ (\lambda w^e(\Rightarrow^{t\to(t\to t)}\ (\texttt{enfant}^{e\to t}\ w)((\lambda x^e\ ((\texttt{prendront}^{e\to(e\to t)}\ x)\ o))\ w)))$
$= \forall^{(e\to t)\to t}\ (\lambda w^e(\Rightarrow^{t\to(t\to t)}\ (\texttt{enfant}^{e\to t}\ w)(((\texttt{prendront}^{e\to(e\to t)}\ w)\ o))))$

C.30. Calculs, par étapes 2/2

$$
\begin{aligned}
& (\mathit{une\ pizza})(\lambda o\ (\mathit{les\ enfants})(\mathit{prendront}\ o)) \\
&= (\lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^{e}(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))(Q\ x)))) \\
&\qquad (\lambda o \forall^{(e\to t)\to t}\ (\lambda w^{e}(\Rightarrow^{t\to(t\to t)}\ (\mathtt{enfant}^{e\to t}\ w)(((\mathtt{prendront}^{e\to(e\to t)}\ w)\ o))))) \\
&= (\exists^{(e\to t)\to t}\ (\lambda x^{e}(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x))) \\
&\qquad ((\lambda o \forall^{(e\to t)\to t}\ (\lambda w^{e}(\Rightarrow^{t\to(t\to t)}\ (\mathtt{enfant}^{e\to t}\ w)(((\mathtt{prendront}^{e\to(e\to t)}\ w)\ o)))))\ x))) \\
&= (\exists^{(e\to t)\to t}\ (\lambda x^{e}(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x))) \\
&\qquad (\forall^{(e\to t)\to t}\ (\lambda w^{e}(\Rightarrow^{t\to(t\to t)}\ (\mathtt{enfant}^{e\to t}\ w)((\mathtt{prendront}^{e\to(e\to t)}\ w)\ x))))))
\end{aligned}
$$

ce qui s'écrit communément :

$$\exists x.\ \mathit{pizza}(x) \wedge \forall w.\ (\mathit{enfant}(w) \Rightarrow \mathit{prendront}(w,x))$$

C.31. Avec l'autre analyse syntaxique...

∀∃

$$
\dfrac{\dfrac{(S/(np\backslash S))/n \quad n}{(S/(np\backslash S))}\,/_e \qquad \dfrac{\dfrac{\dfrac{\dfrac{[np]^1 \qquad \dfrac{(np\backslash S)/np \quad [np]^2}{(np\backslash S)}\,/_e}{S}\,\backslash_e}{S/np}\,/_i(2) \qquad \dfrac{((S/np)\backslash S)/n \quad n}{(S/np)\backslash S}\,\backslash_e}{S}\,\backslash_e}{np\backslash S}\,\backslash_i(1)}{S}\,/_e
$$

Qui correspond à l'analyse :

∀∃

$$\dfrac{\dfrac{\overset{\textit{les}}{(\mathbf{e}\to\mathbf{t})\to(\mathbf{e}\to\mathbf{t})\to\mathbf{t}} \qquad \overset{\textit{enfants}}{(\mathbf{e}\to\mathbf{t})}}{(\mathbf{e}\to\mathbf{t})\to\mathbf{t}}\to_e \qquad \dfrac{\dfrac{\dfrac{\dfrac{\overset{s}{[\ \mathbf{e}]^1}\qquad \dfrac{\overset{\textit{prendront}}{\mathbf{e}\to\mathbf{e}\to\mathbf{t}}\qquad \overset{o}{[\mathbf{e}]^2}}{\mathbf{e}\to\mathbf{t}}\to_e}{\mathbf{t}}\to_e}{\mathbf{e}\to\mathbf{t}}\to_i(2)\qquad \dfrac{\overset{\textit{une}}{(\mathbf{e}\to\mathbf{t})\to(\mathbf{e}\to\mathbf{t})\to\mathbf{t}}\qquad \overset{\textit{pizza}}{(\mathbf{e}\to\mathbf{t})}}{(\mathbf{e}\to\mathbf{t})\to\mathbf{t}}\to_e}{\mathbf{t}}\to_e}{\mathbf{e}\to\mathbf{t}}\to_i(1)}{\mathbf{t}}\to_e$$

λ-terme de la phrase :

$$\forall\exists = (\textit{les enfants})(\lambda s.\ (\textit{une pizza})(\lambda o\ ((\textit{prendront}\ o)\ s)))$$

on insère les λ-termes lexicaux et on calcule

(($\textit{une pizza}$) et ($\textit{les enfants}$) déjà faits)

C.32. Calculs (bis repetita placent)

$(\mathit{une\ pizza})(\lambda o\ ((\mathit{prendront}\ o)\ s))$
$= (\lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))(Q\ x))))$
$(\lambda o\ (((\lambda y^e\ \lambda x^e\ ((\mathtt{prendront}^{e\to(e\to t)}\ x)\ y))\ o)\ s)))$
$= (\lambda Q^{e\to t}\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))(Q\ x))))$
$(\lambda o\ ((\mathtt{prendront}^{e\to(e\to t)}\ s)\ o))$
$= (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))((\lambda o\ ((\mathtt{prendront}^{e\to(e\to t)}\ s)\ o))\ x)))$
$= (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))((\mathtt{prendront}^{e\to(e\to t)}\ s)\ x)))$

$\forall\exists = (\mathit{les\ enfants})(\lambda s.\ (\mathit{une\ pizza})(\lambda o\ ((\mathit{prendront}\ o)\ s)))$
$= (\lambda Q^{e\to t}\ (\forall^{(e\to t)\to t}\ (\lambda u^e(\Rightarrow^{t\to(t\to t)}\ (\mathtt{enfants}^{e\to t}\ u)(Q\ u)))))$
$(\lambda s.\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))((\mathtt{prendront}^{e\to(e\to t)}\ s)\ x))))$
$= (\forall^{(e\to t)\to t}\ (\lambda u^e(\Rightarrow^{t\to(t\to t)}\ (\mathtt{enfants}^{e\to t}\ u)$
$((\lambda s.\ (\exists^{(e\to t)\to t}\ (\lambda x^e(\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))((\mathtt{prendront}^{e\to(e\to t)}\ s)\ x))))\ u)))))$

$= (\forall^{(e\to t)\to t}\ (\lambda u^e(\Rightarrow^{t\to(t\to t)}\ (\mathtt{enfants}^{e\to t}\ u)$
$(\exists^{(e\to t)\to t}\ (\lambda x^e.\ (\wedge^{t\to(t\to t)}((\mathtt{pizza}^{e\to t}\ x)))((\mathtt{prendront}^{e\to(e\to t)}\ u)\ x))))))$

ce qui s'écrit communément :

$$\forall u.\ \mathit{enfants}(u) \Rightarrow \exists.x\ \mathit{pizza}(x) \wedge \mathit{prendront}(u,x)$$

D – The Montagovian Generative Lexicon

D.1. Examples of Lexical Issues

Short roadmap:

- Restriction of selection, polysemy, felicity
- System-F and our framework
- Determiners and quantification
- Classical GL constructions
- Co-predication and constraints
- Deverbals
- Fictive motion
- Integrating plurals and their readings
- Specific issues

D.2. Restriction of Selection and Polysemy

Selection

- Predicates (syntactically) select arguments
- The lexical field of those arguments is restricted
- Other arguments can be forced to behave as expected

Differences in acceptability

- The dog barked.
- The chair barked.
- The drill sergeant barked.
- The hawker barked.

Contrastive ambiguity

- Different lexemes, homophonic / homographic
- Bank, Bar, Pen. . .

Relational polysemy

- A single word, different uses and related meanings
- The bank killed my account.
- The school is on strike.

D.3. Acceptability and Felicity: Semantics, Pragmatics or Both ?

Montague: everything is acceptable

- All syntactically valid items have the same semantic "meaning"
- We have to rely on pragmatics or interpretation

Too strong restriction from lexical semantics

- **e** is replaced by many sorts
- Barking dogs are licensed, everything else is blocked
- No language works that way

Creative uses and semantic licenses

- Fast runners, cars, computers… and phones
- A delicious game (Cooper)
- Expertly built (Adams)

Pragmatic licenses

- Variations from the lexicon in specific contexts external to the utterances
- The father becomes the son of the uncle on the left (Tree manipulation)
- The analyser cannot be expected to get a correct meaning without the external context

Our position on the integration of “Pragmatics”

- Creative use should be included
- Anything that can be comprehended in a self-contained text should be included
- We should not try to account for unknown contextual information
- World knowledge, background, social and universal contexts can be integrated
- Pragmatics should not be a pretext to give up on critical variations of meaning

D.4. System F

Types:

- $\mathbf{t}$ (prop)
- many entity types $\mathbf{e}_i$
- type variables α, β, ...
- $\Pi\alpha.\ T$
- $T_1 \to T_2$

Terms

- Constants and variables for each type
- $(f^{T \to U} a^T) : U$
- $(\lambda x^T.\ u^U) : T \to U$
- $t^{(\Lambda\alpha.\ T)}\{U\} : T[U/\alpha]$
- $\Lambda\alpha.u^T : \Pi\alpha.T$ — no free α in a free variable of u.

The reduction is defined as follows:

- $(\Lambda\alpha.\tau)\{U\}$ reduces to $\tau[U/\alpha]$
  (remember that α and U are types).
- $(\lambda x.\tau)u$ reduces to $\tau[u/x]$ (usual reduction).

D.9. Examples of second order usefulness

Arbitrary modifiers: $\Lambda\alpha\lambda x^{A} y^{\alpha} f^{\alpha\to R}.((\mathsf{read}^{A\to R\to t}\ x)\ (f\ y))$

Polymorphic conjunction:

> Given predicates $P^{\alpha\to\mathbf{t}}$, $Q^{\beta\to\mathbf{t}}$ over respective types α, β,
>
> given any type ξ with two morphisms from ξ to α and to β
>
> we can coordinate the properties P, Q
> of (the two images of) an entity of type ξ:

The polymorphic conjunction $\&^{\Pi}$ is defined as the term

$$
\begin{aligned}
\&^{\Pi} = \Lambda\alpha\Lambda\beta\lambda P^{\alpha\to\mathbf{t}}\lambda Q^{\beta\to\mathbf{t}} \\
\Lambda\xi\lambda x^{\xi}\lambda f^{\xi\to\alpha}\lambda g^{\xi\to\beta}. \\
(\mathrm{and}^{\mathbf{t}\to\mathbf{t}\to\mathbf{t}}\ (P\ (f\ x))(Q\ (g\ x)))
\end{aligned}
$$

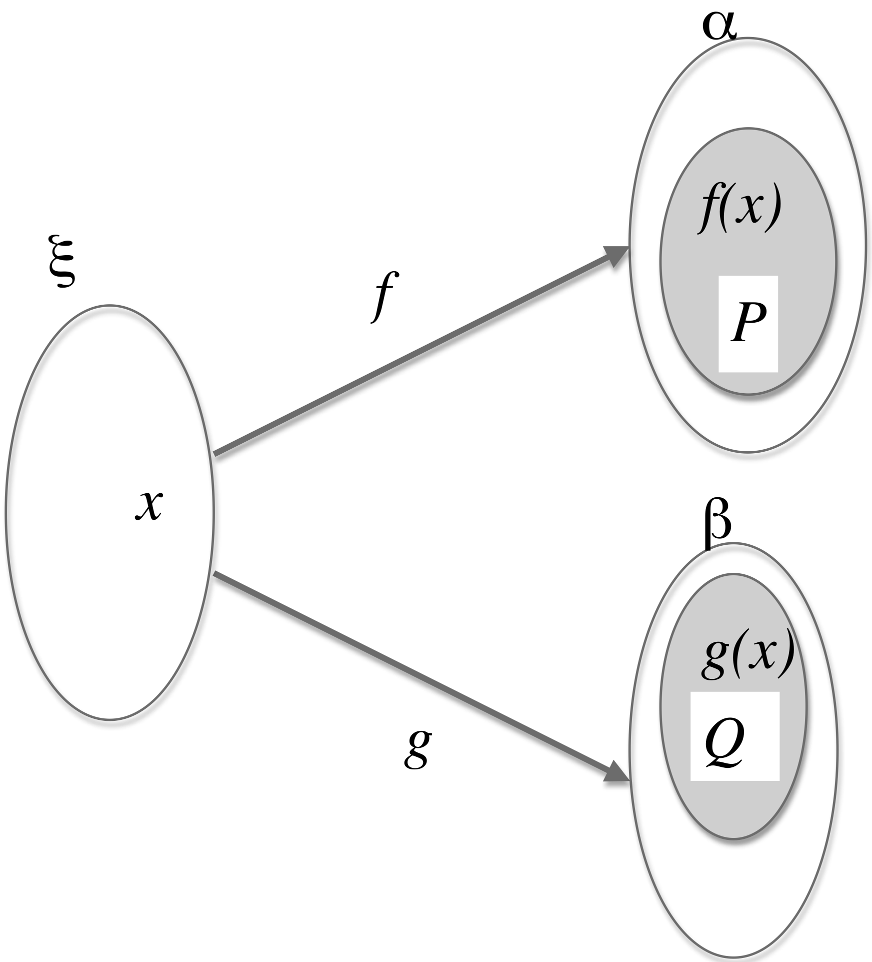


Figure 1: Polymorphic conjunction: $P(f(x))\&Q(g(x))$ with $x : \xi$, $f : \xi \to \alpha$, $g : \xi \to \beta$.

D.12. In a Nutshell

The Generative Lexicon

- Pustejovsky, 1995 (and precursors)
- Discussed and refined by Asher, Cooper, Luo. . .
- Idea: the lexicon provides enough data to generate word meanings in context

Framework sketch

- Types (and terms) from System-F
- Lexical entries are typed with many sorts
- Each word has a single main λ-term
- Each word can have any number of optional λ-terms
- Those terms are transformations, and are word-based
- Normal application is the same
- Transformations are used when types clash
- Types guide the selection of transformations

D.13. Lexicon v. Type

Why do we think transformations are lexical ?

(Rather than type-driven)

In a word: idiosyncrasy.

Consider:

(5) La classe est finie. (Event)
(6) La classe est fermée. (Location)
(7) La classe est de bon niveau. (People)
(8) La promotion est de bon niveau. (univoque: People)

In French, the two words do not have the same possible uses, but represent exactly the same group of people.

(This also seems to be the case in American English.)

D.14. Strong Idiosyncrasy

Linguistic constructs are not independent of the language

(Pleonasm ?)
Idioms and specific constructs are illustrations of this.

Differences of language
I have punctured

Differences of dialect
Un demi-fraise

Differences of jargon
Redresse la #16

D.15. Toy Example

Named towns are examples of highly polysemous words that can be referred to for their location, population, and many other aspects.

- Types : T (town), Pl (place), P (people)
- Usual predications:

  1. Birmingham is spread out
  2. Birmingham voted labour
  3. 1 & 2

Lexical item	Main λ-term	Modifiers
Birmingham	$birmingham^T$	$Id_T : T \rightarrow T$ $t_2 : T \rightarrow P$ $t_3 : T \rightarrow Pl$
is_spread_out	$spread\_out : Pl \rightarrow \mathbf{t}$	
voted	$voted : P \rightarrow \mathbf{t}$	

1. Type mismatch in $\mathit{spread\_out}^{Pl\to\mathbf{t}}(\mathit{Birmingham}^{T}))$, resolved using t_3:

$$\mathit{spread\_out}^{Pl\to\mathbf{t}}({t_3}^{T\to Pl}\mathit{Birmingham}^{T}))$$

2. The same, using t_2:

$$\mathit{voted}\,P \to \mathbf{t}({t_2}^{T\to Pl}\ \mathit{Birmingham}^{T})$$

3. We use a polymorphic conjunction operator, $\&^{\Pi}$.

   –as seen.

   $$\Lambda\xi\lambda x^{\xi}\ \lambda f^{\xi\to\alpha}\lambda g^{\xi\to\beta}(\mathrm{and}^{(\mathbf{t}\to\mathbf{t})\to\mathbf{t}}\ (\mathit{spread\_out}\ (f\ x))(\mathit{voted}\ (g\ x)))$$

   After application, we have:

   $$(\mathrm{and}\ (\mathit{spread\_out}^{Pl\to\mathbf{t}}\ ({t_3}^{T\to Pl}\ \mathit{Birmingham}^{T}))$$
   $$(\mathit{voted}^{Pl\to\mathbf{t}}\ ({t_2}^{T\to P}\ \mathit{Birmingham}^{T})))$$

Complex example

Lexicon

word	principal λ-term	optional λ-terms	rigid/flexible
Birmingham	$Birmingham^T$	$Id_T : T \to T$	(F)
		$t_1 : T \to F$	(R)
		$t_2 : T \to P$	(F)
		$t_3 : T \to Pl$	(F)
is_spread_out	$spread\_out : Pl \to \mathbf{t}$		
voted	$voted : P \to \mathbf{t}$		
won	$won : F \to \mathbf{t}$		

where the base types are defined as follows:

T	town
F	football club
P	people
Pl	place

Z Conclusion

Z.1. Exemple de questions d’actualité

Comment acquérir un lexique sémantique sophistiqué ?

Comment obtenir des préférences entre les différentes interprétations logiques d’un même énoncé ?

Comment bien modéliser la quantification (portée, liage "in situ") faire la différence entre "tout" et "chaque" ? Comment interpréter les quantificateurs généralisés (la moitié, beaucoup, peu, "les",....).

Quelle notions naturelle de sous typage qui corresponde aux relations ontologiques ?

Z.2. Un domaine interdisciplinaire

Il est possibles d’étudier de jolies questions de logique
qui aient un sens du point de vue linguistique,
et qui permettent d’analyser automatiquement des textes.

Réciproquement, les linguistes ou les philosophes du langage
ou les besoins en traitement automatique des langues
posent de bonnes questions au logicien.