Discursive analysis of itineraries in a historical regional corpus of travels

syntax, semantics & pragmatics in a unified type theoretical framework

Richard Moot[Bx], Laurent Prévot[Aix], Christian Retoré[Bx]

Bx: LaBRI, INRIA, Université de Bordeaux

Aix: LPL Université de Provence

Providing a type theoretical frame work for a treatment from syntax to discourse (CG syntax, compositional DRT semantics, meaning transfers, discourse relations)

1. corpus and objective, the virtual traveller problem
2. categorial parser with DRS outputs
3. extending the type system for lexical pragmatics
4. a lexicon and an analysis involving a virtual traveller

No interpretation of the semantic representation.

Faithful modelling of the linguistic analyses of motion verbs done by others.

Part I
Data, question and outline

1. A case study, a field for semantic experiments

Corpus: French, XVII-XX centuries (mainly XIX), travel stories through the Pyrenees (576.334 words).

Goal: given a part of text, can we reconstruct the itinerary followed by the traveller?

2. From parsing to semantics within type theory

- Multimodal categorial grammar
  - syntactic lexicon acquired form corpora
    Moortgat Moot style MMCG
  - semantic/lexicon hand written (much smaller prototype)
    lambda DRT
- lexicon designed for semantics pragmatics
  meaning transfer functions, second order lambdas
- ? finding out narration and elaboration relations
  S DRT
- ?? itinerary reconstruction
  spatial reasoning using our geographic data base

During this process advocate for the use of variable types (system F).

3. A particular phenomenon: virtual traveller, fictive travel

Sometimes one ought to consider a virtual travel(ler) (cf. Talmy) in particular because of the duration indication:

(1) The path descended abruptly.

(2) The road runs along the coast for two hours.

This does not mean that someone actually follows the path.
How do we model this?

4. When do we know whether the path is actually followed?

(3) La route est bordée à droite par les montagnes, à gauche par la plaine qui va en se resserrant jusqu'à Pierrefite,

(4) les routes de Lux et de Cauterets séparent.

(5) Celle de Lux entre dans une gorge qui vous mène au fond d'un précipice et traverse le gave de Pau.

(6) Ici commence la route qui est taillée dans le rocher, elle est encaissée ;

(7) les parois du rocher sont à une hauteur prodigieuse et le gave qui gronde à plus de 100 mètres au dessous,

(8) cette route monte jusqu'à Lux où l'on arrive par une jolie avenue de peupliers.

(9) Ici tout a changé, ce n'est plus ces montagnes arides, ni ce chemin tortueux, mais une jolie vallée entourée de montagnes couvertes de verdure ;

(10) c'est là que la petite ville de Lux est placée.

5. Some particularities of fictive motion

(11) Nous coupons ici un sentier qui vient du port de Barroude (...)
Here, we cross a path which comes from the pass of Barroude

(12) La route suit le gave qui vient de Gavarnie.
The road follows the mountain stream coming from Gavarnie.

(13) Plus loin, de nobles hêtres montent sur le versant (...)
Further away, noble beeches climb the slope

(14) (...) cette route qui monte sans cesse pendant deux lieues
this road which climbs incessantly for two miles

(15) Le chemin pavé de calcaire et de pierres luisantes (...) serpente à travers fourrés de buis et de noisetiers
The road paved with limestone and shining stones winds across buxus and hazels shrubbery

Part II
Categorial syntax and semantics

6. Grail categorial parser and French grammar

- Grail is a general-purpose parser for (multimodal) categorial grammars.
- A wide-coverage French grammar has been semi-automatically extracted from the French Treebank.
- On the basis of the 382.145 words and 12.822 sentence of the treebank, the extraction algorithm extracts 883 different formulas, of which 664 occur more than once.
- Many frequent words are assigned *many* different formulas.
- Standard statistical methods (*supertagging*) help with lexical disambiguation.

7. Number of entries for common words and POS tags

et	CONJ	71
,	PONCT	62
à	PRP	55
plus	ADV	44
ou	CONJ	42
est	VERB	39
être	INF	36
en	PRP	34
a	VERB	31

ADV	206
VERB	175
PRP	149
CONJ	92
PONCT	89

8. Syntactic categories

Atomic categories: *np* (noun phrase), *n* (common noun), *s* (sentence) and *pp* (prepositional phrase) (actually some more are needed).

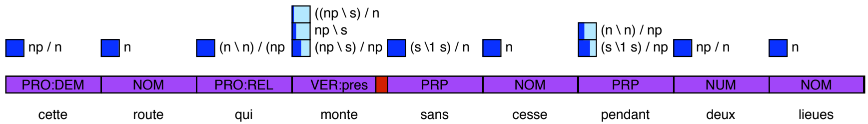
Categories:

- atomic categories
- A/B whenever A and B are categories
- $B\backslash A$ whenever A and B are categories

Rules:

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

9. Example



- The supertagger assigns supertag with a confidence level (indicated by the darker blue part of the square before the formula)
- when it is less sure it proposes more alternatives
- Most words only have a single formula is proposed by the supertagger, the verb “monte” (which does not occur at all in the training corpus) has three possible formulas and the preposition “pendant” has two.

10. Semantics with lambda-DRT

We use a λ-DRT entities/individuals divided into several sorts Ty_n.

We also use event variables for reification (this is common, but possibly harmful).

Extension to TY_n without difficulty nor suprise: e can be divided in several kind of entities.

It's a kind of flat ontology: objects, concepts, events,...

Let us show an example of a lexicon, and the treatment of an example.

$\oplus$ is the DRS merge operation.

word phrase syntactic type	lambda-DRS
descend $np\backslash s$	$\lambda x^{person} \lambda e^{event} \begin{array}{\|l\|} \hline p^{path} \\ \hline travel(e, x, p) \\ height(source(p)) \\ \quad > height(destination(p)) \\ \hline \end{array}$
Jean $s/(np\backslash s)$	$\lambda P^{person \rightarrow event \rightarrow t} \lambda e^{event} \begin{array}{\|l\|} \hline y^{person} \\ \hline Jean(y) \\ \hline \end{array} \oplus ((P\ y)\ e)$

11. Syntactic / semantic analysis

Syntactic analysis is proving

Jean descend $\vdash s$ ....

and it tells us that the semantic is the application of the semantic λ-term of *Jean* to the one of *descend*

$$\left((\lambda P^{person \to event \to t} \lambda e^{event} \begin{array}{|c|} \hline y^{person} \\ \hline Jean(y) \\ \hline \end{array} \oplus ((P\ y)\ e)) \right.$$

$$\left. \lambda x^{person} \lambda e^{event} \begin{array}{|l|} \hline p^{path} \\ \hline travel(e, x, p) \\ height(source(p)) \\ \quad > height(destination(p)) \\ \hline \end{array} \right)$$

$$\to_\beta$$

$$\lambda e^{event} \begin{array}{|l|}\hline y^{person} \\ \hline Jean(y) \\ \hline \end{array} \oplus \Bigg(\big(\lambda x^{person} \lambda e^{event} \begin{array}{|l|}\hline p^{path} \\ \hline travel(e, x, p) \\ height(source(p)) \\ \quad > height(destination(p)) \\ \hline \end{array} y\big)$$

$$\rightarrow_\beta$$

$$\lambda e^{event} \begin{array}{|l|}\hline y^{person} \\ \hline Jean(y) \\ \hline \end{array} \oplus \big(\lambda e^{event} \begin{array}{|l|}\hline p^{path} \\ \hline travel(e, y, p) \\ height(source(p)) \\ \quad > height(destination(p)) \\ \hline \end{array} e\big)$$

$$\rightarrow_\beta$$

$$\lambda e^{event} \begin{array}{|l|}\hline y^{person} \\ \hline Jean(y) \\ \hline \end{array} \oplus \begin{array}{|l|}\hline p^{path} \\ \hline travel(e, y, p) \\ height(source(p)) \\ \quad > height(destination(p)) \\ \hline \end{array}$$

$\longrightarrow$ DRS merge

$$\lambda e^{event} \begin{array}{|l|}\hline y^{person} \; p^{path} \\ \hline Jean(y) \\ travel(e, y, p) \\ height(source(p)) > height(destination(p)) \\ \hline \end{array}$$

$\longrightarrow$ Existential closure

$e^{event}\ y^{person}\ p^{path}$
$Jean(y)$ $travel(e, y, p)$ $height(source(p))$ $\quad > height(destination(p))$

Part III

Extending the type system:

ΛTy_n, $F\!-\!DRT$

12. Second order types (Girard's F).

Ty_n (several base types) filters the sort of the argument according to lexical constraints, but....

uniform operations on all terms of all types are useful.

For co-predication, for (generalized) quantification, and all operation that act uniformly upon all types, one can also add type variables and quantification over types.

13. More general types and terms. Second order types (Girard's F).

- Constants e and t, as well as any type variable α in P, are types.
- Whenever T is a type and α a type variable which may but need not occur in T, $\Pi.\alpha.\ T$ is a type.
- Whenever T_1 and T_2 are types, $T_1 \rightarrow T_2$ is also a type.

14. More general types and terms. Second order terms (Girard's F).

- A variable of type T i.e. $x : T$ or x^T is a *term*. Countably many variables of each type.
- $(f\ \tau)$ is a term of type U whenever $\tau : T$ and $f : T \rightarrow U$.
- $\lambda x^T.\ \tau$ is a term of type $T \rightarrow U$ whenever $x : T$, and $\tau : U$.
- $\tau\{U\}$ is a term of type $T[U/\alpha]$ whenever $\tau : \Lambda\alpha.\ T$, and U is a type.
- $\Lambda\alpha.\tau$ is a term of type $\Pi\alpha.T$ whenever α is a type variable, and $\tau : T$ without any free occurrence of the type variable α.

15. More general types and terms. Second order reduction.

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).

16. Remarks: system F

- used for the syntax of semantics (a.k.a. metalogic, glue logic)
- the formulae of semantics are the usual ones
- a single constant, e.g. for the quantifier $\forall$ or the choice function ι which is specialized for each type
- less types (constrained) than formulae with a free variable

It is also the type system of the polymorphic functional programming languages ML, CaML,...

17. More general types and terms. A second order example.

Given two predicates $P^{\alpha\to t}$ and $Q^{\beta\to t}$
over entities of respective kinds α and β
when we have two morphisms from ξ to α and to β
we can coordinate entities of type ξ:
$\Lambda\xi\lambda x^{\xi}\lambda f^{\xi\to a}\lambda g^{\xi\to b}.(and\ (P\ (f\ x))(Q\ (g\ x)))$

One can even quantify over the predicates P, Q and the types α,β to which they apply:
$\Lambda\alpha\Lambda\beta\lambda P^{\alpha\to t}\lambda Q^{\beta\to t}\Lambda\xi\lambda x^{\xi}\lambda f^{\xi\to\alpha}\lambda g^{\xi\to\beta}.(\mathrm{and}\ (P\ (f\ x))(Q\ (g\ x)))$

Part IV

The theoretical lexicon at work: the virtual traveller

18. Principles of our lexicon

- Remains within realm of Montagovian compositional semantics (possibly we are not so keen on possible worlds semantics).
- Allows both predicate and argument to contribute lexical information to the compound.
- Works with λ-DRT.

We advocate a system based on *optional modifiers* which can account for lexical idiosyncrasies.

Second-order typing, like Girard's F system is needed for arbitrary modifiers:

$$\Lambda\alpha\lambda x^{A} y^{\alpha} f^{\alpha\to R}.((\mathrm{read}^{A\to R\to t}\ x)\ (f\ y))$$

19. A lexical entry

- syntactic category x
- A standard Λ-term of type x^* attached to the main sense:
  - Used for compositional purposes
  - Comprising detailed typing information
  - Including slots for optional modifiers
- several Λ-terms modifiers (type changes), that can be:

rigid the same modifier applies once for every occurrence. (blocks impossible copredications)

flexible modifier: a different modifier can be used for each occurrence.

20. Rules

$$\frac{A/B : f^{U \to T} \quad B : x^U}{A : (fx)^T} \; /E$$

$$\begin{array}{c} \ldots [B : x^U] \\ \vdots \\ \dfrac{A : t^T}{A/B : \lambda x^U t} \; /I \end{array}$$

$$\frac{A/B : f^{\Pi\alpha . U[\alpha] \to T} \quad B : x^{U[V]}}{A : (f\{V\}x)^T} \; /E^*$$

Correspondence: syntactic rule / semantic counterpart.

Instantiation and application are combined. The funciton only partly specifies it argument. Thereafter *le* is an instance of this process.

21. Specialised Lexical Semantics

A rather minimal and schematic model.

Base types *region* and *path*

- cognitively motivated (Jackendoff)
- linguistically motivated (some verbs require one).

height is a function from regions to their vertical coordinate.

Functions *source* and *destination*: they convert a path p to its source region and its destination region.

Predicate *middle*(p, r) where p is a path and r a region $\rightarrow$ distinction between Initial, Median and Final verbs as in Asher an Sablayrolles

Spatial variable *here* position and orientation of the spatial reference point not necessarily the narrators's place, implemented as a succession of values.

Motion verbs: relations between one or more entities and a *path* (possibly implicit)

Verbs specify lexically which of their arguments follow this path (subject, object or both, see e.g. Nam).

22. A lexicon

word/phrase syntactic type	lambda-term
chemin n	$\lambda x^{immobile\_object} \begin{array}{\|l\|}\hline \\ \hline chemin(x) \\ \hline\end{array}$
g n/n	$\lambda P^{immobile\_object \rightarrow t} \lambda p^{path} \begin{array}{\|l\|}\hline x^{immobile\_object} \quad q^{path} \quad here^{region} \\ \hline path\_of(y, p) \\ subpath(q, p) \\ source(q) = here \\ \hline\end{array} \oplus (P\ x)$

chemin *n*	$\lambda x^{immobile\_object}$ $\begin{array}{\|c\|}\hline \\ \hline \textit{chemin}(x) \\ \hline \end{array}$

“chemin” is true of entities of type immobile object for which *chemin*(x) holds

$$
\begin{array}{l|l}
g & \lambda P^{immobile\_object \rightarrow t} \lambda p^{path} \begin{array}{|l|}\hline x^{immobile\_object} \quad q^{path} \quad here^{region} \\ \hline path\_of(y,p) \\ subpath(q,p) \\ source(q) = here \\ \hline \end{array} \oplus (P\ x) \\
n/n &
\end{array}
$$

coercion g: from an immobile object x to a path p, correspondence indicated by the predicate *path_of*

and selecting a sub-path q of p going forward from *here*, which may or may not go to the end of the path p.

both x (immobile) and p (path) are DRT referents. Indeed, both aspect can be used “a brick road to Pau” and anaphors can refer to either aspect.

(16) The street was completed in 1825 (...)

(17) It runs from the Regent’s residence at Carlton House (...) to All Souls Church.

le $(s/(np\backslash s)/n$	$\Lambda\alpha\lambda P^{\alpha\to t}\lambda Q^{\alpha\to event\to t}\lambda e^{event}\begin{array}{\|c\|}\hline x^{\alpha} \\ \hline \\ \hline \end{array} \oplus (P\ x) \oplus ((Q\ x)\ e)$

"Le" is viewed as a generalised quantifier. It selects a noun (subset of α) and a VP which applies to α to produce a sentence.

<table>
<tr><td>descend
$np\backslash s$</td><td>$\lambda x^{person}\lambda e^{event}\begin{array}{|l|}\hline p^{path} \\ \hline \mathit{travel}(e,x,p) \\ \mathit{height}(\mathit{source}(p)) > \mathit{height}(\mathit{destination}(p)) \\ \hline \end{array}$</td></tr>
<tr><td>h
$(np\backslash s)/(np\backslash s)$</td><td>$\lambda P^{person\to event\to t}\lambda p^{path}\lambda e^{event}\begin{array}{|l|}\hline \\ \hline \begin{array}{|l|}\hline x^{person} \\ \hline \mathit{travel}(e,x,p) \\ \hline \end{array} \Rightarrow ((P\ x)\ e) \\ \hline \end{array}$</td></tr>
</table>

“descend” main term:

given a person argument x and an event argument e,

the DRS checks that there exists a path p such that x follows p and that the height at the start of this path p is greater than his height at the end of it.

What about coercion?

The coercion *h* (for “descend” but not for any motion verb) applied to *descend* yields if a person follows the path p, then he descends.

h descend
$np\backslash s$

$$\lambda p^{path} \lambda e^{event} \left[\begin{array}{l} \\ \hline \left[\begin{array}{l} x^{person} \\ \hline travel(e,x,p) \end{array}\right] \Rightarrow \left[\begin{array}{l} \\ \hline height(source(p)) > \\ height(destination(p)) \end{array}\right] \end{array}\right]$$

Note that “h descend” does not commit us to concluding that anyone actually takes the path. This must be deduced separately.

le{path} (g chemin) $\lambda P^{path \rightarrow event \rightarrow t} \lambda e^{event}$

$s/(np\backslash s)$

$y^{immobile\_object}$ p^{path} q^{path} $here^{region}$
$chemin(y)$
$path\_of(y, p)$
$subpath(q, p)$
$source(q) = here$

" le chemin" with type assignment $np - \iota x^{immobile\_object}.chemin(x)$ does not combine with “descend” which requires a person as its argument, $np\backslash s - \lambda y^{person}$ ...

Both “chemin” and “descend” permit lexically anchored type coercions, which solves the type mismatch:

- “chemin” has a lexical lambda term g which coerces it in such a way that “le chemin” obtains type assignment $np - \iota x^{path}.chemin(x)$
- whereas “descend” has a lexical lambda term h which coerces its lexical semantics to $np \backslash s - \lambda y^{path}$....
- With both coercions “le chemin descend” is a correctly typed term, with “le chemin” being a term of type $path$ and “descend” a term of type $path \rightarrow t$.

23. Two remarks

variable *here* (place + orientation)

no incoherence between “le chemin monte” and “le chemin descend” (reversed orientation)

“pendant deux heures” simple Davidsonian analysis: the duration of the corresponding event is two hours.

pendant 2 h.	$\lambda s^{event \to t} \lambda e^{event} (s\ e) \oplus \begin{array}{\|c\|}\hline \\ \hline \mathit{duration}(e, 2h) \\ \hline \end{array}$
$s \backslash s$	

Part V

Towards Segmented Discourse Representation Structures

24. Objective

Finding discourse relations,

- in particular *narration* and *elaboration* for the itinerary
- others, to skip what is irrelevant.

25. Some phenomena

(18) nous descendons, pendant un quart d'heure, la vallée de l'Esera.
we descend, for a quarter of an hour, the Esera valley.

(19) La lune, qui éclaire notre marche, nous fait découvrir sur la droite un sentier qui serpente.
The moon, which lightens our steps, allows us to discover a winding path on our right.

(20) Il nous conduit sur un petit plateau, au milieu de sapins, au-dessus et à quelque distance du torrent de Ramun.
It leads us to a small plateau, surrounded by firs, at some distance of and above the Ramun torrent.

"Il" (it) in sentence 20 refers to "un sentier qui serpente"

imposes anaphora resolution before coercion.

Constraints on the possible interpretations *Background*(18,19) and *Narration*(19,20).

26. Examples, yet other remarks

Rhetorical structure: important but hard to infer.

(21) Nous partimes pour Barèges à 8 heures du matin par une fort jolie route qui nous conduisit à Lourdes.
We left (PS) for Barèges at 8 in the morning, taking a very pretty road which led (PS) us to Lourdes.

(22) (...) qui va en se resserrant jusqu'à Pierrefite, où les routes de Lux et de Cauterets séparent.
(...) which goes shrinking along the way, up to Pierrefite, where the roads to Lux and to Cauterets split.

(23) Celle de Lux entre dans une gorge qui vous mène au fond d'un précipice et traverse le gave de Pau.
The one to Lux enters a gorge which leads you to the bottom of a precipice and traverses the Gave de Pau.

(24) (...) Après une longue marche, l'on arrive à Barèges à 6 heures du soir.
(...) After a long walk, we arrive in Barèges at 6 in the evening.

21 introduces the destination and therefore the whole spatio-temporal extension route. The following will therefore constitute an

Elaboration relation between this sentence and the sequence of 22-24.

It is (at first sight) difficult to decide on the discourse relation of

Sentence 23: it would certainly be possible to have a later phrase beginning with “Celle de Cauterets” (the road leading to Cauterets) and a number of the following sentences (omitted here for space reasons) give further background information about the road to Lux.

However, at sentence 24, it suddenly becomes evident that the author has been describing the road while following it.

27. Conclusions and Future Work

Model of “virtual movement” in a type-logical grammar by extending Montague-style semantics

- DRT
- Generative Lexicon

Using system F for varying types and flexibility when it is lexically allowed.

Implementation of coercion in a prototype by Emeric Kien.

Open (common) problems

- anaphora resolution
- determining the appropriate discourse relations between segments of text

Suppressing DRS who are not relevant to the itinerary question (comparison, reason why they acted that way).