Mosaïque @ Arcachon

# Which formal languages for natural languages?

(revision of a talk for Huet 60th birthday)

Christian Retoré
LaBRI (CNRS et Université de Bordeaux)
INRIA Bordeaux Sud-Ouest

## Survey with something new

- Formal syntax of natural language
- Natural language syntax with strings
- State of the art and discussion
- Tree languages for natural language
- The place of Ed Stabler's minimalist grammars in the hierarchy (very recent joint work with Gregory Kobele and Sylvain Salvati)

## Back to the origins of computational linguistics

Which formal languages
for natural language syntax?
(first strings, then trees)

## Two traditions

1. Logic and grammar
   - Denis from Thrax (Alexandria, Byzance)
   - Scholastics
   - Frege, Montague, Lambek
2. Grammar and computation
   - Panini
   - Chomsky, Schutzenberger
3. Mixed (new in Computational Linguitics) Model theoretic syntax
   - 60's TCS: Buchi, Doner, Thatcher,…
   - 90's CL: Mönnich, Rogers, Morawietz, Pullum, …

## Two traditions

1. Logic and grammar
   ++ connexion to semantics
   + learning
   - - efficiency, complexity
2. Grammar and computation
   ++ Complexity, (abstract) machines
   - Learning
   - - Connexion to semantics

Me: 1 visiting 2

## Some ideas from generative grammar

- Language ≠ corpus
  He believes that (longuest sentence)
- Language: set of unconscious rules
  evidence: learning overgeneralisation.
  Against learning by imitation.
  Why the child holded the baby rabbit
- Competence (rules) ≠ performance
  The wheat {that the rat [that the cat (that the dog chased) killed] ate} was poisonous.

## Some ideas from generative grammar

- Universal grammar / parameters
  explaining the acquisition paradox
- Movement / comparison between sentences
  Which book that Chomsky wrote did he like?
  He likes three books that Chomsky wrote.
- Syntax/semantics
  quantifiers
  possible impossible coreferences
  (affirmative: he and Chomsky non coreferent)

## Two principles from generative grammar

1. Fast (polynomial?) analysis
   Grammaticality is decided quickly by speakers
2. Learnable under some conditions
   - Knowing argument structure and root meaning
   - With interaction
   - With prosody
   - With positive examples only
   - Not that much positive examples
   - By iterated restrictions of the language

## Two mixable kinds of finite descriptions of a class of well-formed expressions.

- Formal Grammar
  – CFGs, TAGs, HPSGs, CGs,
- Logic, finite model theory
  Model Theoretic Syntax
  – CFGs, TAGs, CGs, CxGs, GP,…

## Two mixable kinds of finite descriptions of a class of well-formed expressions.

- Formal Grammar
  – Rules generating the potential infinity of sentences, structures,….
  – Computationally, Efficient,
  – Difficult to write and understand
    (especially if lexicalised)
- Logic, finite model theory
  Model Theoretic Syntax
  – The set of strings or terms satisfying a set of constraints -> degrees of grammaticality.
  – No natural underlying computational process.
  – Natural for linguistic descriptions, easy to write.

## String Grammars

Usual Hypotheses
and current State of the Art

## Formal grammars

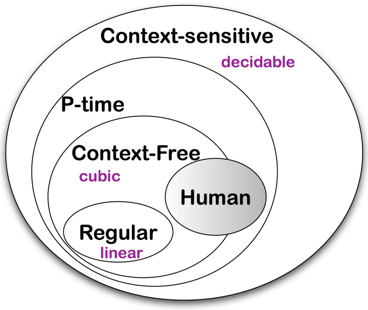
- T terminals, N non terminals
- Rules W –> W' (W: at least one N)

={
  – W=W1 Z W2 and W'= W1 W'' W2
    context sensitive
  – |W'|≥|W| length increasing
  – |W|=1 context-free
  – |W|=1 and W'=mZ regular

## Which string languages?

- Center-embedded relatives
  Pierre (que Pierre)$^n$ connaît$^n$ dort.
  at least context-free.
- Dutch (Swiss-German) completives
  …dat ik$_1$ Henk$_2$ haar$_3$ de nijlpaarden$_3$ zag$_1$ helpen$_2$ voeren$_3$
  … that I$_1$ see$_1$ Henk$_2$ help$_2$ her$_3$ to feed$_3$ the hippopotamuses

## The current hypothesis on human string languages




Challenged from time to time:

Michaelis & Kracht 96 old Georgian is not semi-linear

Kobele 06 Yoruba involves unbounded copying

## Mildly context sensitive languages

- First notion:
  - Tree Adjoing Grammars 1975 come back 1991
  - Combinatorial Categorial Grammars
- A larger one:
  - Multi-Component-TAG Weir
  - Minimalist grammars Stabler 1996
  - LCFRS Vijay-Shankar,Weir, Joshi
    Seki, Matsumura, Fujii, Kasimi
- Large classe = P-time
  Range Concatenation Grammars Boullier

## Discussion: complexity

- Recursion limited to two (or say five)
  - Computer = finite state automaton??
  - Speakers (with extra processing time) accept nested sentences
  - Rules are stated like this by speakers, books, …
  - Economy of the description

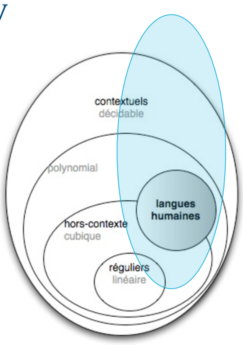
## Discussion: word order

- Models of strict word orders, what about more free word order (e.g. with rich morphology, Latin, Russian, Sanskrit)
  - Standard answer: there is a canonical order from which other are derived and it induces semantic nuances
  - A hidden answer: it is much simpler to work with total orders then with partial orders!!

## Discussion: acquisition

- Acquisition condition left out… but very important
  - for understanding human language faculty
  - for building large grammars from corpora.
- Exception: categorial grammars can be learnt:
  - lexicalized
  - structured types -> unification

## Learnable languages in the Hierarchy




## Discussion: local state of the art

- Richard Moot MMCG: extraction, parsing
  - NWO Dutch Spoken Corpus (spontaneous conversation, annotated transcript)
    - 1.002.098 word occurrences
    - 114.801 phrases (7,6 words per sentence)
    - 44.306 different word forms
  - Multi-Modal Categorial Grammar, acquired from the corpus (average 100 trees per word!)
  - Supertagging (n-most likely sequences of trees corresponding to the words in the sentence)
  - Results on test corpus 19.237 sentences 146.497 words (supertagging >> parsing):
    - 1 supertag 2'53'' 40% correct (9 ms/sent., 1.18 ms/wd)
    - 10 best supertags 48'34'' 70% correct (151ms/sent., 20ms/wd)

## Discussion: local state of the art

- Benoît Sagot, Eric de la Clergerie LFG parsing
  - Corpus EASy (Evaluation des Analyseurs Syntaxiques) Newspapers, web, mail, political speeches, literature,…
    - 87177 word occurrences
    - 4322 sentences (20,2 words per sentence)
  - Handwritten LFG grammar
  - Selects one parse per sentence
  - Parsing time: total 152s, 35ms/sentence 1,7ms/word
    - Correct chunks: 86%
    - Correct relations: 49%

## Discussion:how to compare two different practical states of the art

1. Mainly written
2. Rather long sentences ~ 20 words
3. Flat annotations
4. Hand written grammar
5. Lexical Functional Grammar
6. Correctness measure: results on chunks

1. Spoken
2. Very short but tricky sentences <10 words
3. Deeply annotated
4. Automatically acquired grammar
5. MultiModal Categorial Grammar
6. Correctness results on whole parse structure

## Tree grammars

- Strings are not enough:
  - For learning
  - For interpreting sentences
- Graphs (proof-nets of categorial grammars, dependency graphs) would be much welcome …….but let's start with trees.

## Tree grammars

(that I am just discovering, be indulgent)

## Context-free tree grammars (Engelfriet after Fisher)

- A ranked signature of terminals
- A ranked signature of non-terminals
- Productions rules of the form

$$A(x_1,\ldots,x_n) \quad \rightarrow \quad t(x_1,\ldots,x_n)$$

- where $A$ non terminal of arity $n$
- where $t$ tree over terminals and non terminals with variables $x_1,\ldots,x_n$

## Regular Tree Grammars Thatcher, Doner, 1967

- Rules only for non-terminals of rank 0 rewrite (ONLY LEAVES rewrite)
- These tree languages exactly are the ones definable in monadic second order logic
- Their yields are context free strings languages

## Context Free Tree Grammars Fisher 1968, Engelfriet 1977

- **OI** (~ unrestricted) only the highest non terminal undergo rewriting.
  Strings: indexed languages
- **IO** only the lowest non terminals undergo rewriting.
  Strings: LCFRS (incomparable)
- Monadic (always a single NT)
  **CFTG (IO=OI) ~ TAG derived trees**
  **Mönnich 1996**

## Context free Hyper Edge Replacement Grammars Courcelle 1987, Engelfriet

- Non terminal: hyper edges (ordered with possible repetitions)
- External vertices
- Replace an hyper edge with one with the same external vertices, possibly with new hyperedges linking them

## Where are the tree languages that I like?

Categorial grammars
Minimalist grammars

## Categorial grammars

- Old notion: parse tree: any proof tree any bracketting is possible…
- Normal natural deduction only (Tiede)
- Non associative Lambek grammars
  - RTG Tiede (?), Kandulski
  - ACG encoding Salvati Retoré
- Associative Lambek grammars
  - RTG are not enough (despite CFL only)
  - CFTG Salvati september 2007

## Stabler’s minimalist grammars

- Close to categorial grammars or linear logic but much richer
- Implements Chomsky’s minimalist program
- Lexicalised
- Two operations
  - Merge (binary)
  - Move (unary)

## Minimalist grammars

- Trees with a head “<“ or “>” on internal nodes, indicating where the head is.
- Complete trees: a single c on the head, only words on other leaves
- Sequences of features on the leafs
  - Selection
    - d n v .......
    - =d =n =v .....
  - Movement
    - +wh +k ....
    - -wh -k ...

Lexical items sequence of features associated with a word, possiby empty

## Minimalist grammars

- Merge
  - a tree t with head =x w
  - Another tree t’ with head xw’
- Result
  suppress the x and =x yielding t and t’
  the selector si the head
  the selected is not

  <( t ; t’ ) if t is lexical (a leaf)

  >( t’; t ) if t is a real tree

## Minimalist grammars

- Move
  - a tree t[t’] with head +f w and a subtree t’ with head -f w
- Result
  supress the +f and -f yielding t and t’
  the context is the head

  >( t’ ; t[ε])

## Minimalist grammars: lexicon

Jon : d

aime : =d =d v

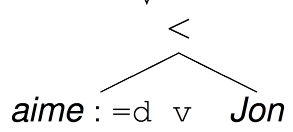
qui : d −WH

ϵ : =v +WH c

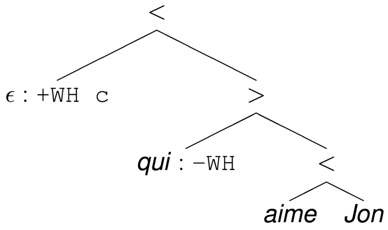
## Minimalist grammars: merge

*aime* : =d =d v + *Jon* : d

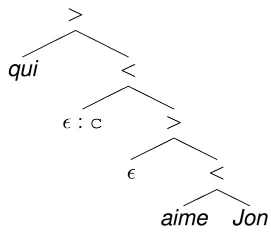
⇓



## Minimalist grammars: merge




## Minimalist grammars: move




## Shortest move condition SMC

- Chomsky: whenever two subtrees (-f) are competing for a movement triggered by (+f), the one closest to the attractor (+f) moves.
- Stabler: whenever two subtrees (-f) are competing for a movement triggered by (+f), the derivation crashes. Strong SMC !

## Minimalist tree languages in the hierarchy

As the image by a transducer of a regular language

## Two step description

de Mönnich, Morawietz, Michaelis

- If minimalist tree languages are complicated, can we describe them as the image by a simple mechanism of a simple set of tree languages.
- MG->MCFG
- Lift -> RTG (derivation trees)
- Walking Tree Automaton computing dominance, precedence of the MG derived trees

## A simpler and lower description

Kobele, Retoré, Salvati

- Derivation trees (regular set):
  lexical, move(_) merge (_,_)
  Tree tuples
  [main tree, (-$f_1$ subtree), …., (-$f_n$ subtree)]
  Strong SMC at most one subtree per $f_i$
- Eliminate the derivations that fail (still regular)
- Defined move and merge on tuples of trees
- Can be done with a Linear Deterministic Mult. Bottom-Up Tree Transducer

## Merge with tuples of trees

$$(t_0[=xw], t_1, \ldots, t_n) \quad (t'_0[xm], t'_1, \ldots, t'_n)$$

- Compute $<(\underline{t_0}, t'_0)$ or $>(t'_0, \underline{t_0})$
- Put the trees in the tuple, and if there are two trees whose head starts with the same -f, the derivation crashes. (Strong Shortest Move Condition)

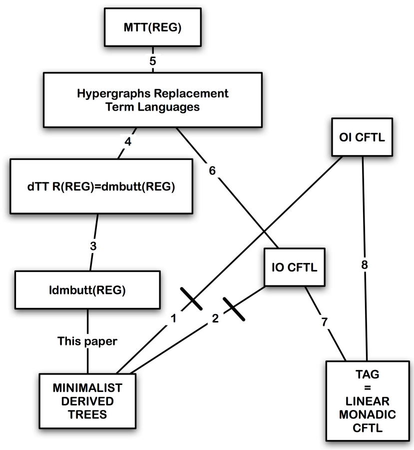
## Move with tuples of trees

$$(t_0[+f_i w], t_1, .., t_i[-fm], \ldots, t_n)$$

- Compute $>(t_i, \underline{t_0})$
- Put the trees in the tuple, and if there are two trees who's head starts with the same -f, the derivation crashes. (Strong Shortest Move Condition)

## Interpreting this result

- Filtering the wrong derivation tree is linear (bottom up automaton)
- The computing of the derived tree ensures to be included into HR CFG (technical horrible reason: a top-down tree transducer with regular look-ahead and finite copying can do what a linear deterministic multi bottom up tree transducer does)




## Conclusion

- Admittedly, little is know, but we're learning and starting to clear the picture.
- At least we know where stand a foramlisation of a/the main linguistic theory
- Improving the connexion between logical formalisms and rewrite formalisms
  - Syntax / Semantics correspondence
  - Parsing efficiency (kind of compilation)
- The need for two kinds of descriptions:
  - Model Theoretic Syntax: linguistic description
  - Derivational syntax: processing

## Some references


- Edward Stabler A derivational approach to minimalism. LACL Springer 1996
- James Rogers *A descriptive approach to language complexity* CSLI 1998
- Frank Morawietz *Two step approaches to natural language formalism* Mouton de Gruyter 2003
- Greg Kobele, Christian Retoré, Sylvain Salvati: An automata -theoretic approach to minimalism in *Model Theoretic Syntax at 10.* 2007
- Christian Retoré Les mathématiques de la linguistique computationnelle. Premier volet: la théorie des langages. *La gazette des mathématiciens,* Société mathématique de France. January 2008