Test Suite Generation

Sylvain Schmitz

LORIA, INRIA Nancy - Grand Est, Nancy, France

NaTAL Workshop, Nancy, June 25, 2008

Issues with Surface Generation

*Jean que cherches-tu est grand.
Jean qui baille s'endort.
Le chat noir est grand.
Il le faut.
Beaucoup de chats noirs se lavent.
Jean qu'il arrive agit.
*Le chat est avec beaucoup de poils grand.
Jean qui agit est grand.
*Le repas que attendez-vous arrive.
*Jean est avec chat grand.

Overgeneration

Definition

Equivalently, if a grammar

- assigns incorrect structure to grammatical sentences
- accepts agrammatical sentences
- generates agrammatical sentences

Error Mining

van Noord [2004], Sagot and Éric de la Clergerie [2006]

Applied to undergeneration:

1. parse a large corpus of correct sentences
2. failures indicate coverage issues
3. statistical analysis identifies a probable culprit for each failure
4. might attempt to provide corrections

Error Mining

van Noord [2004], Sagot and Éric de la Clergerie [2006]

Applied to undergeneration:

1. parse a large corpus of correct sentences
2. failures indicate coverage issues
3. statistical analysis identifies a probable culprit for each failure
4. might attempt to provide corrections

For Overgeneration?

Which test suite for pass/failure?

- a TreeBank
- a corpus of incorrect sentences
- sentences generated from the grammar
  - which input?
  - generation from logic formulae is NP-complete

For Overgeneration?

Which test suite for pass/failure?

- a TreeBank
- a corpus of incorrect sentences
- sentences generated from the grammar
  - which input?
  - generation from logic formulae is NP-complete

For Overgeneration?

Which test suite for pass/failure?

- a TreeBank
- a corpus of incorrect sentences
- sentences generated from the grammar
  - which input?
  - generation from logic formulae is NP-complete

For Overgeneration?

Which test suite for pass/failure?

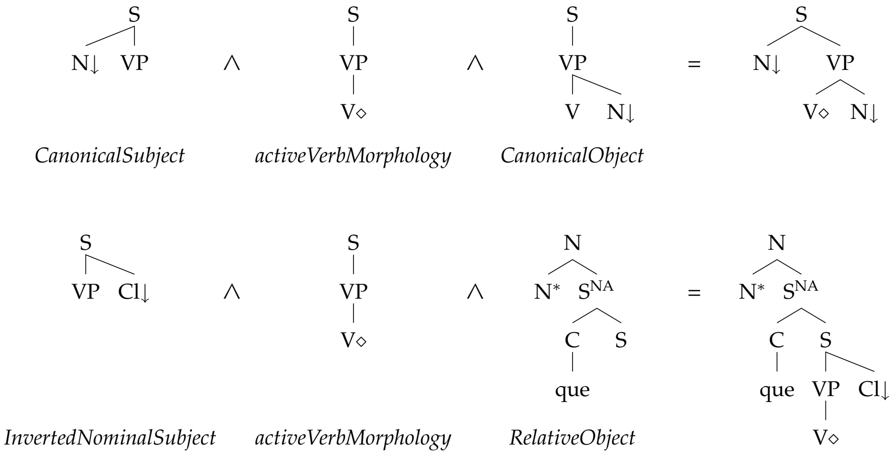
- a TreeBank
- a corpus of incorrect sentences
- sentences generated from the grammar
  - which input?
  - generation from logic formulae is NP-complete

"Exhaustive" Generation

- not in terms of elementary trees (about 6,000)
- in terms of *linguistic phenomena*
    - grammar compiled from a meta grammar
    - compilation traces
    - 87 classes match linguistic phenomena

Meta Grammar

XMG, Crabbé [2005] and many others

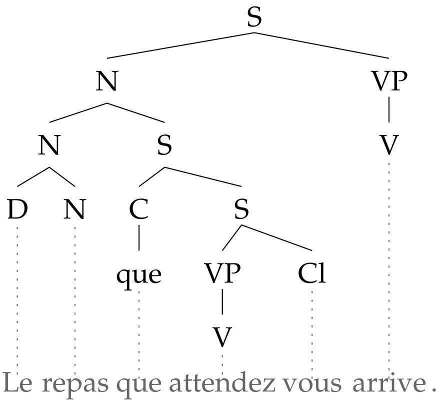


Guided Generation

Input:
bag of classes

{ *InvertedNominalSubject*, *RelativeObject* }

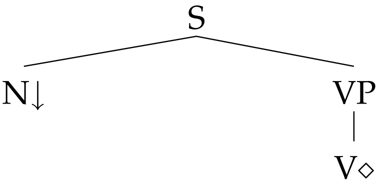
Output:
set of trees



Algorithm

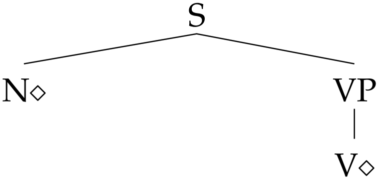
S

Algorithm

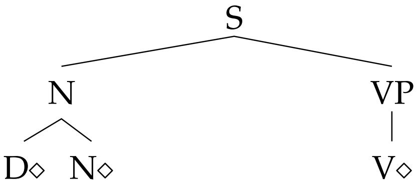


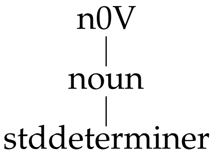
n0V

Algorithm

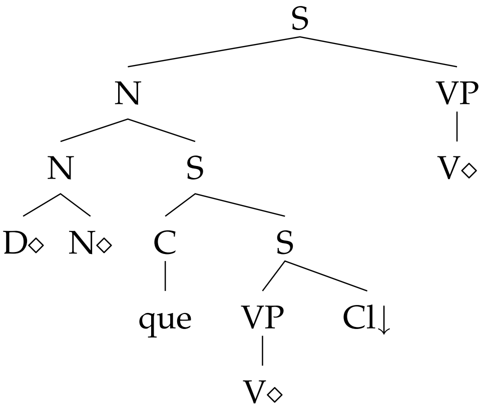


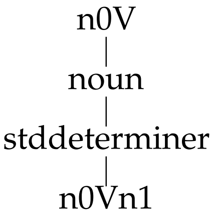
Algorithm



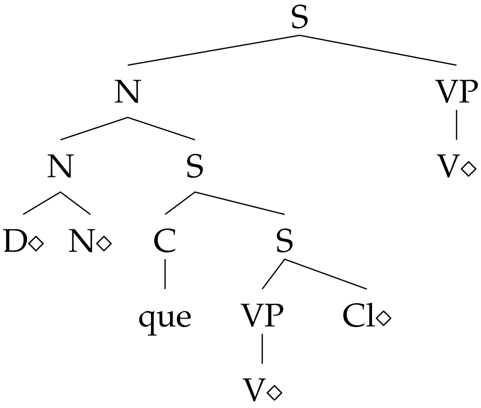


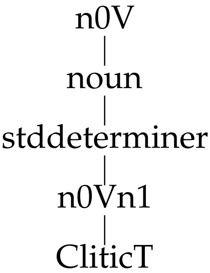
Algorithm



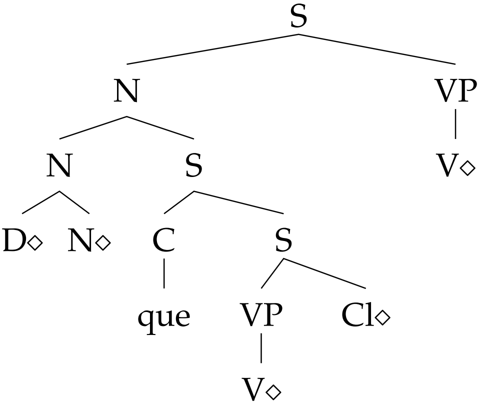


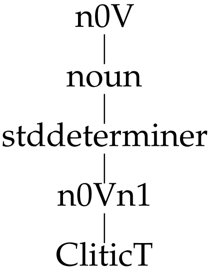
Algorithm


S
N
VP
N
S
V◇
D◇
N◇
C
S
que
VP
Cl◇
V◇


n0V
noun
stddeterminer
n0Vn1
CliticT

Algorithm

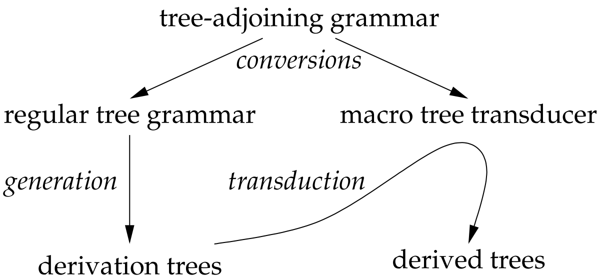




Le repas que attendez vous arrive .

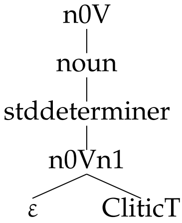
Digression: 2-level Syntax

Shieber [2006] and many others



Digression: Regular Tree Grammar

$$
\begin{aligned}
S_s &\rightarrow \mathrm{n0V}(N_s) \\
N_s &\rightarrow \mathrm{noun}(N_a) \\
N_a &\rightarrow \mathrm{stddeterminer}(N_a) \\
N_a &\rightarrow \mathrm{n0Vn1}(N_a, Cl_s) \\
N_a &\rightarrow \varepsilon() \\
Cl_s &\rightarrow \mathrm{ClíticT}()
\end{aligned}
$$



- need to account for feature structures
  see [S and Le Roux, 2008]

Algorithm, again

- derivation-tree centric
- distances computed using the accessibility relation in the regular tree grammar
- elementary tree selection uses distances
  - to the remaining target classes
  - to the globally accumulated classes
  - to the classes accumulated in the current derivation

Experiments with the Algorithm

- issue: non termination of the grammar
- experiments on small controlled subsets
- larger generation using GenI
  see [Gardent and Kow, 2007]

Experiments with Error Mining

Issues: Ordering Suspects

```
Worst Form Number: 1    0.220409692766
complexAdvDeDeterminer  s0Pv1post
d7   0.220780944532
Worst Form Number: 2    0.185227187873
AdjectivalPredicativeform  s0Pv1post
d7   0.185539179236
d11  0.185187092097
```

Experiments with Error Mining

Issues: Bigrams

```
Worst Form Number: 5    0.0836861365418
CanonicalSubject  InvertedNominalSubject
d10  0.0838670966961
d1   0.0838670966961
InvertedNominalSubject  RelativeObject
d10  0.0838670966961
d1   0.0838670966961
```

Random Concluding Remarks

- application of two-level syntax
- opens new issues with error mining
- entropy measures?

B. Crabbé. Grammatical development with XMG. In P. Blache, E. Stabler, J. Busquets, and R. Moot, editors, *LACL'05*, volume 3492 of *Lecture Notes in Computer Science*, pages 84–100. Springer, 2005. ISBN 978-3-540-25783-7. doi: 10.1007/11422532_6.

C. Gardent and E. Kow. Spotting Overgeneration Suspects. In *ENLG'07*. 2007. URL `http://hal.inria.fr/inria-00149372/`.

B. Sagot and Éric de la Clergerie. Error mining in parsing results. In *ACL'06*, pages 329–336. ACL Press, 2006. doi: 10.3115/1220175.1220217. URL `http://www.aclweb.org/anthology/P06-1042`.

S. Schmitz and J. Le Roux. Feature unification in TAG derivation trees. In C. Gardent and A. Sarkar, editors, *TAG+9*, 2008. URL `http://www.loria.fr/~schmitsy/pub/tagunif.en.pdf`.

S. M. Shieber. Unifying synchronous tree-adjoining grammars and tree transducers via bimorphisms. In *EACL'06*. ACL Press, 2006. ISBN 1-932432-59-0. URL `http://www.aclweb.org/anthology/E06-1048`.

G. van Noord. Error mining for wide-coverage grammar engineering. In *ACL'04*, pages 446–453. ACL Press, 2004. doi: 10.3115/1218955.1219012. URL `http://www.aclweb.org/anthology/P04-1057`.