Treebanks as parsed corpora: an introduction

Koenraad De Smedt, Paul Meurer, Victoria Rosén

University of Bergen and Uni Digital

Prague, December 13–14, 2010

Overview

- Introduction (*Koenraad*): Motivation, Approaches, Issues
- Methodology (*Victoria*): LFG Parsebanking with Discriminants, Infrastructure Projects
- Details and demos (*Paul*): Workflow, Search, Dependency Annotation, Parallel Parsebanking, Demos

Language technology needs

- English now accounts for less than 1/3 of language content on the web
- Europe with its variety of languages accounts for 50% of the worldwide language services market
- E-commerce and online public services are on the rise
- Better search, document retrieval and other content aware processing is needed for a variety of languages
- Content is available as never before, but cannot readily be used due to lack of detailed analyses
- What does this have to do with treebanks?

Deep syntactic and semantic analysis for information search

Example

Which athletes died this year?
— In my sophomore year of high school, I heard from the principal over the PA system ... I can't possibly honor every athlete who died too young, but I can certainly honor some.
— On Monday, the 17-year-old was dead, the victim of a .... about her son's death and the importance of more safeguards to protect athletes.

— 5/09/2010 Erica Blasberg, a promising young golfer on the LPGA tour, was found dead in her Las Vegas home.

Powerset (acquired by Microsoft in 2008)

Search engine based on deep syntactic and semantic analysis

Automatic parsing of training corpus, with manual disambiguation

↓

Training of stochastic parser

↓

Web turned into a big parsebank

↓

Question is parsed with same grammar as the web material to be searched

Pred-arg relations for anaphor resolution

Example

The police officer was searching for the suspect.
(a) He had been investigating the murder since Tuesday.
(b) He had committed the second murder on Tuesday.

From an analyzed corpus:

Police are first argument of *{take, supervise, investigate, expand on, agree, experience, indicate, keep, encourage, confirm}*

Deep analysis is needed to find predicate-argument relations

The KunDoc project has used such knowledge for detecting reference chains and for concept clustering

Need for treebanks

Data and tools

- Testing material for computational grammars
- Material for inducing computational grammars
- Training material for disambiguating parsers
- Frequencies for anaphor resolution, concept clustering etc.

Many applications

- Language technologies: content mining, search, question answering
- Theoretical and applied linguistics: grammar studies, study of language learning (L2 corpora), study of language variation and change, etc.
- Other applications?

Example theoretical use: Gradience of dative alternation

Bresnan and Nikitina (2008) investigate soft constraints on the realization of beneficiaries in treebank

Examples (found by us)

- And he gave me a sad smile. (*He gave a sad smile to me)
- And as I gave it to him my heart was torn. (*I gave him it)
- ...what you gave me to drink was like music. (*to* not a preposition)
- "Draw me a sheep!"
- So I drew for him one of the two pictures I had drawn so often.

(From *The Little Prince*)

Why deep analysis?

- Some constructions cannot be found in corpora with shallow annotation

Example (Relative clauses without complementizers)

The plane we wanted to take was canceled.

- Shallow annotation does not deal well with ambiguities (local or real)

Example

Suspected Islamic militants shot dead
...at least 22 Hindus
...at least 22 times

Levels of analysis

- Hierarchical relations (phrase structure)
- Functional relations (subject, direct object, modifier, etc.)
- Semantic relations (predicate-argument, scope, etc.)
- Discourse relations (topic, focus, etc.)
- Translational relations (in parallel treebanks)

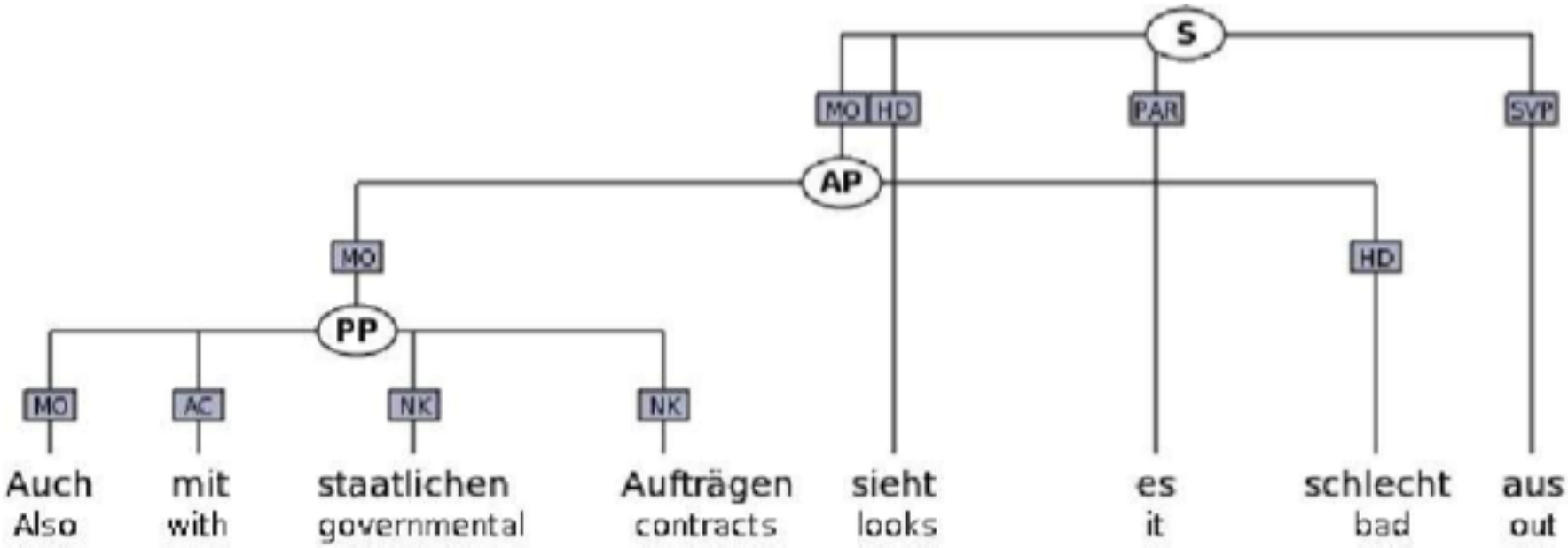
Constituent structure (phrase structure)

Labeled bracketing with indentation (from IBM Paris Treebank)

```
[N Vous_PPSA5MS N]
[V accedez_VINIP5
   [P a_PREPA
      [N cette_DDEMFS session_NCOFS N]
   P]
   [Pv a_PREP31 partir_PREP32 de_PREP33
      [N la_DARDFS fenetre_NCOFS
         [A Gestionnaire_AJQFS
            [P de_PREPD
               [N taches_NCOFP
                N]
            P]
         A]
      N]
   Pv]
V]
```

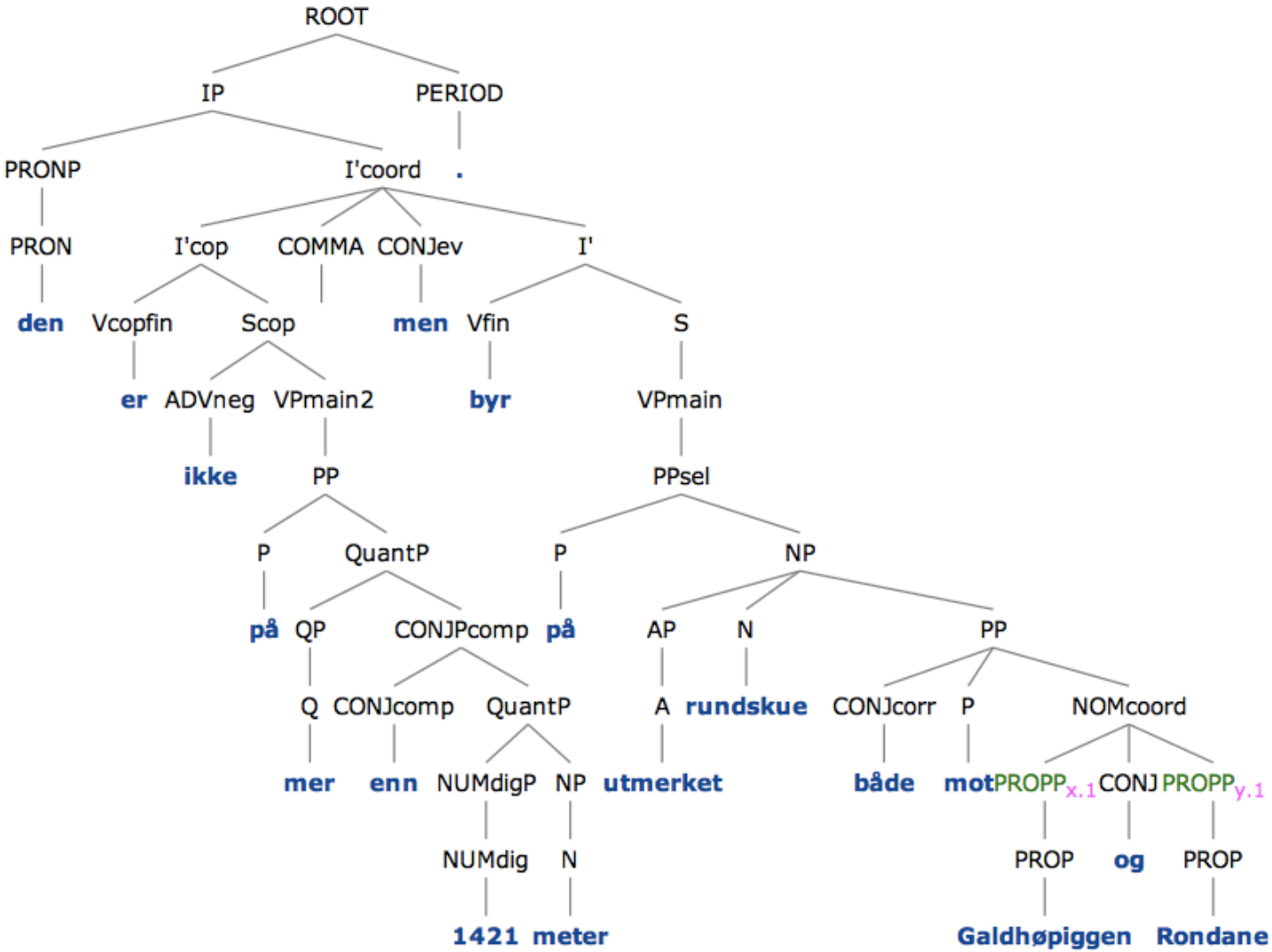
Tree structures

TIGER, with crossing branches (from Rehbein 2007)

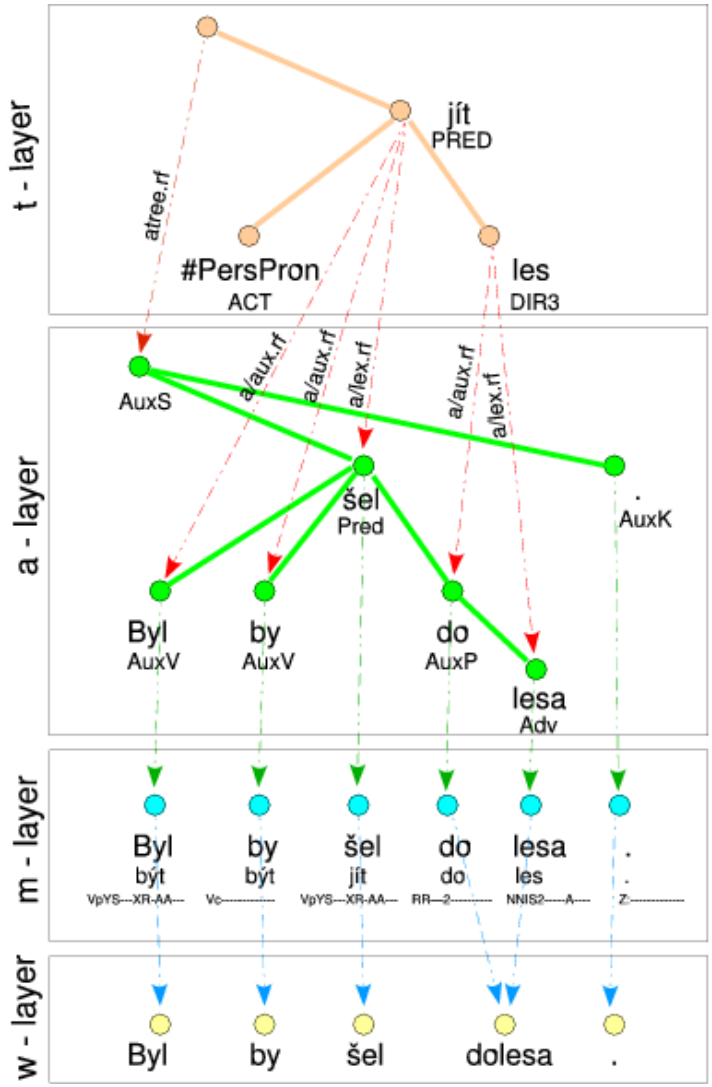


Tree structures

C-structure, TREPIL/LOGON (Scalable Vector Graphics)



Dependency structures (from Prague Dep. Treebank)



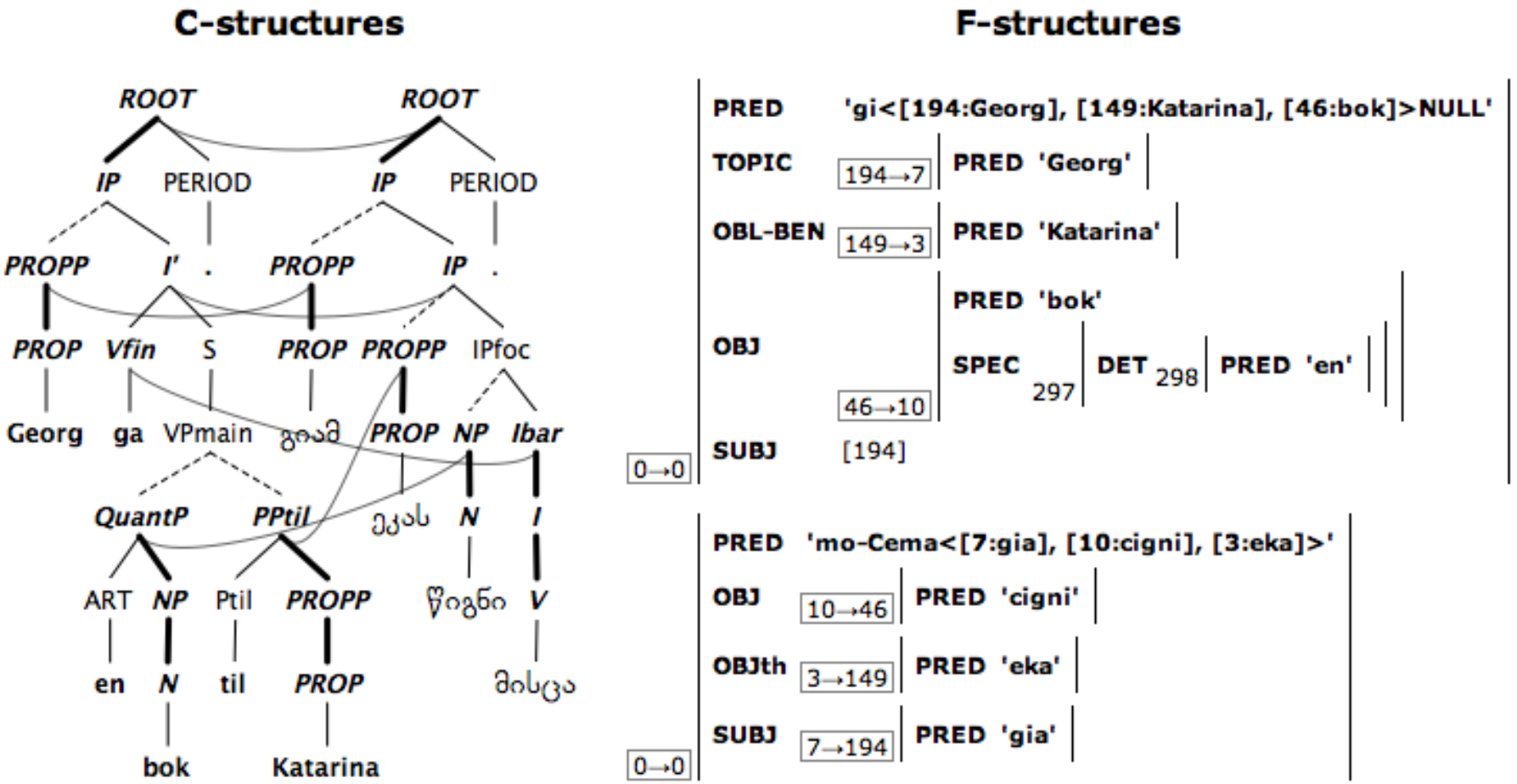
Feature-structure based representations

[*feature : value*]

- Functional structures [*subj* [*pred :* 'John']]
- Semantic structures e.g. MRS
- etc.

Notation and visualization differences between treebanks may be superficial, but sometimes they reflect somewhat different theoretical constraints (some formalisms allow crossing branches at phrase structure level, others require discontinuities to be handled at a different level).

Parallel treebanks



Annotation strategies

A treebank is a corpus of authentic language samples, annotated with grammatical structures

Few treebanks are either fully manually or automatically constructed

Options

- Machine-aided manual construction
- Human-aided machine construction

Advantages and disadvantages?

Manual methodology

Preprocessing/Tagging

↓

Manual annotation (machine-aided)

↓

Grammar induction

↓

Error detection and correction

Incremental process of grammar induction and correction

Manual annotation

Advantages: Does not require grammar, can be used for grammar induction

Problems:

- Much manual labor, does not scale up well
- Practical limit to detail and complexity of structures
- Errors and inconsistencies in manual annotation (thus: needs error detection)
- Large number of rules: 17,500 rules for 50,000 sentences in Penn Treebank (thus: needs compaction)
- Induced grammar may not capture linguistic generalizations

Parsebanking methodology (TREPIL project)

XLE/LFG Parsing (automatic)

↓

Efficient disambiguation (manual, with advanced tool)

↓

(Grammar revision, reparsing, automatic re-disambiguation)

↓

Training of stochastic disambiguator

Incremental process, manual revision of grammar but automated reparsing and re-disambiguation with previous disambiguation choices

Parsebanking

Advantages:

- Automated process, scales up well
- Allows high detail and complexity of structures
- Perfect consistency with grammar
- Rules designed to capture linguistic generalizations in the language beyond the corpus

Challenges:

- Requires existing grammar and good parser
- Requires disambiguation
- May not cover whole corpus
- Incremental approach is needed

Is coverage a problem?

- Not every item in a corpus can be parsed automatically
- But should every item be parsed?

Consider:

- Non-syntactic input
- Performance phenomena
- The creativity of language
- Difficult syntactic problems

Not every item is a sentence

- Typographically a sentence $\neq$ grammatically a sentence
- Many items headlines, lists, headers, etc.
- These may or may not have grammatical structure

Not every item is a sentence

Example (UPenn *Wall Street Journal* treebank)

8 13/16% to 8 11/16% one month; 8 13/16% to 8 11/16% two months; 8 13/16% to 8 11/16% three months; 8 3/4% to 8 5/8% four months; 8 11/16% to 8 9/16% five months; 8 5/8% to 8 1/2% six months.

What is the benefit of analyzing this as a grammatical sentence?

Not every item is a sentence

- Syntactic annotation may not be theoretically motivated for items that lack true syntactic structure
- Non-syntactic items may still be annotated, for instance with part of speech tags

Spoken language

- Spoken language characterized by dysfluencies
- False starts, repetitions, repairs, etc.
- No widespread consensus on how these should be annotated

Christine treebank

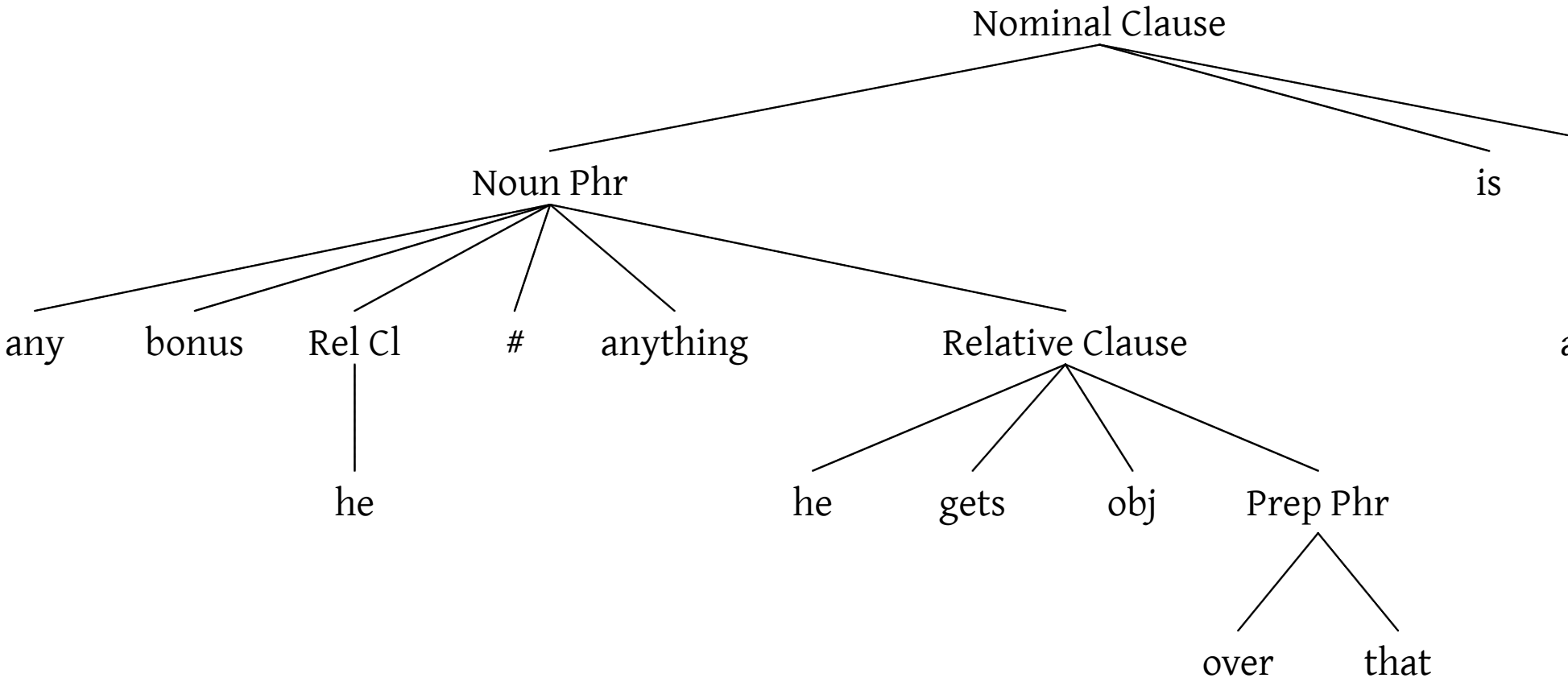
Example

any bonus he # anything he gets over that is a bonus

Geoffrey Sampson:

- "we need rules for deciding how to fit that symbol, and the words before and after it, into a coherent structure[...]"
- "Where in the tree do we attach the interruption symbol?"
- "[...] these guidelines have had to grow quite complex; but only by virtue of them can thousands of individual speech repairs be annotated in a predictable, consistent fashion"

Christine treebank example



Spoken language

- Consistent annotation of speech repairs is good for the study of speech repairs
- Better to have performance phenomena and syntactic structure on different annotation levels

Written language

- Also written language has numerous performance errors
- Spelling mistakes, typos, repetitions, omissions, etc.
- Questionable grammar use

Written language

Examples

He wants to among other things to go fishing.
Petter går til butikken, parken og til slottet.
“Petter goes to the store, the park and to the castle.”

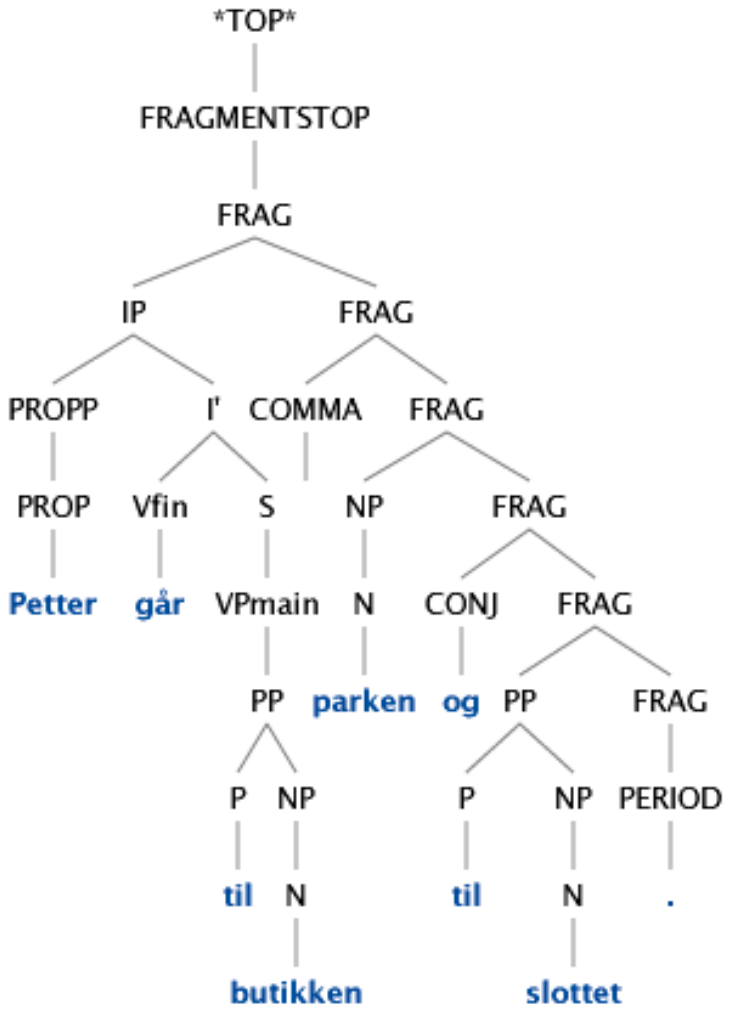
What would happen if we included all such possibilities in the grammar?

Written language

Including such errors in the grammar would lead to overgeneration; two better alternatives:

- We can parse only the grammatical parts (*fragment parse*)
- We can correct the error and produce a full analysis, retaining the information about what was corrected

Written language



Language is open-ended

- A handwritten grammar will always miss some constructions
- Not just because of accidental omissions, but because language is creative and open-ended
- Parsebanking is a methodology for incremental development and continuous testing of a grammar against a corpus

Language is open-ended

Examples (Resultatives)

He wiped the table clean.
She hammered the metal flat.
Det trekker til seg store mengder fugler som deretter skiter eiendommen full.
"This attracts large quantities of birds who subsequently poop the property full."

How to handle the open-endedness of language?

Handling open-endedness of language

- Add the appropriate subcategorization frame to the lexicon
- The sentence will get the intended analysis on the next reparsing of the corpus
- Creative uses are impossible to predict, but can be incorporated with an interactive and incremental approach to parsebanking

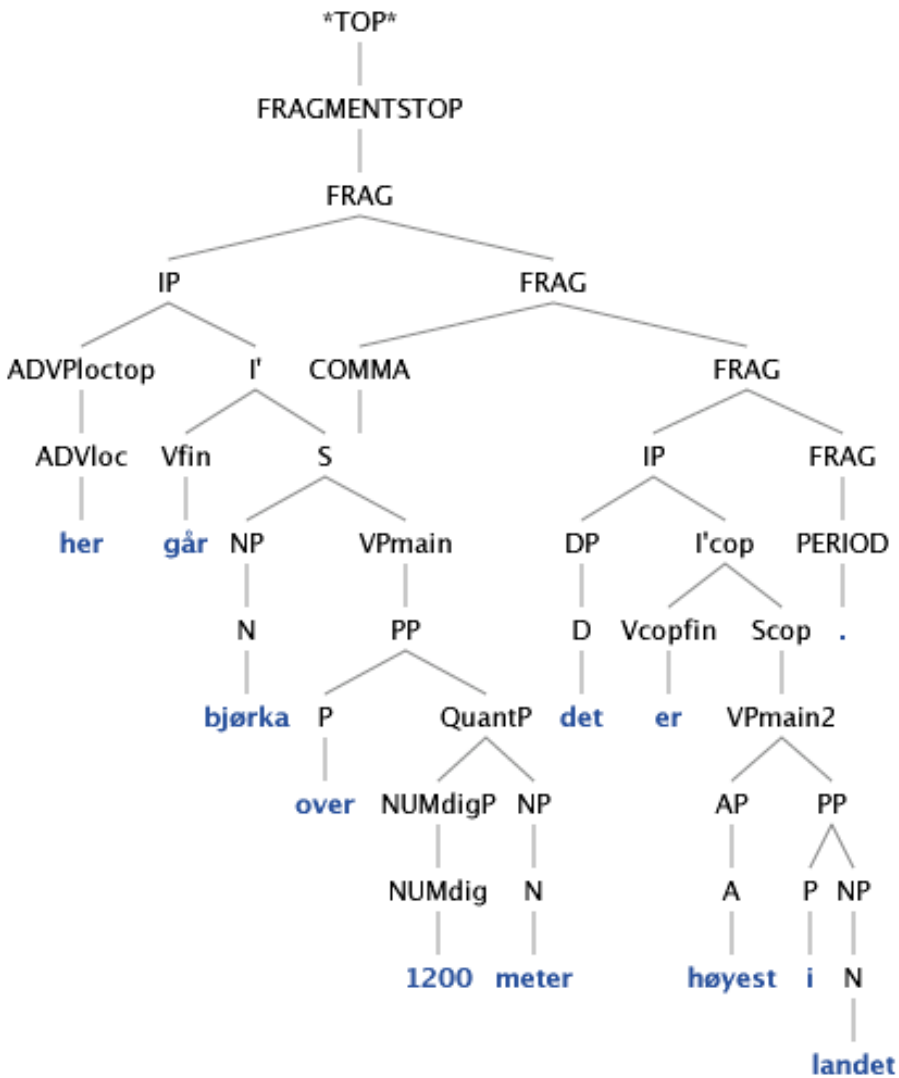
Handling open-endedness of language

- Some constructions are on the borderline of grammaticality
- Evidence from the corpus being parsed can help one decide whether they should be included in the grammar

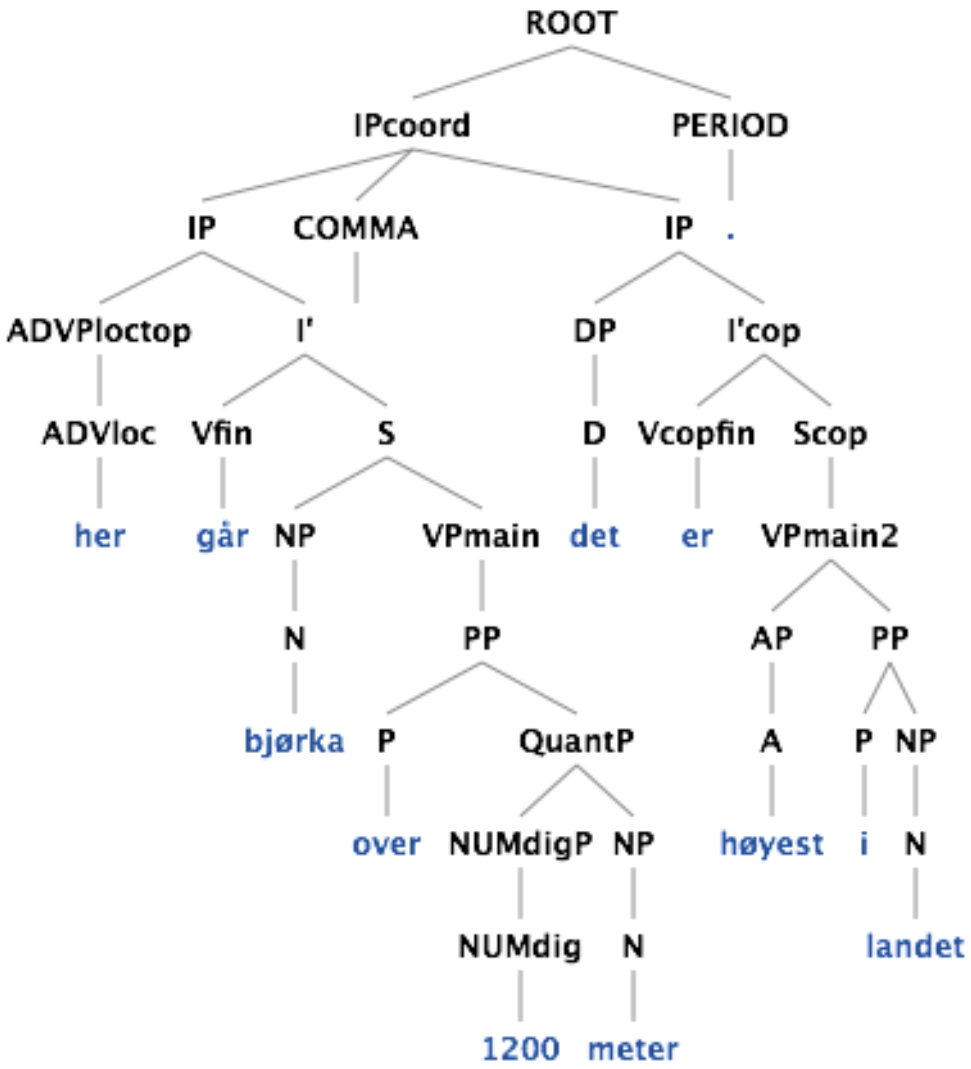
Examples (Unmarked coordination)

Her går bjørka over 1200 meter, det er høyest i landet.
"Here birches grow over 1200 meters, that is the highest in the country."

Handling open-endedness of language



Handling open-endedness of language



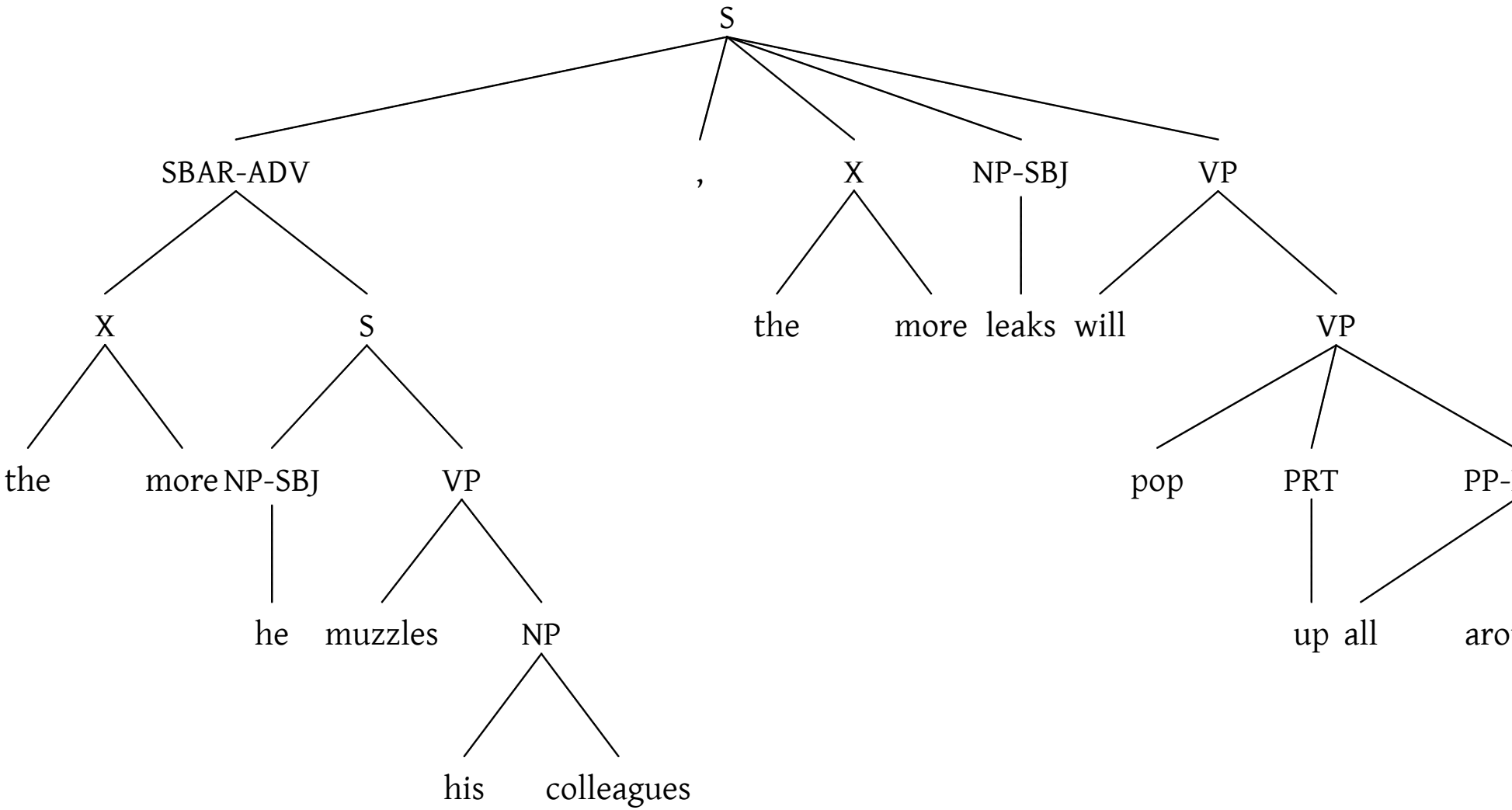
Difficult syntactic problems

There are genuinely difficult syntactic problems, but these are also difficult for approaches that use manual annotation.

e.g. *the more, the merrier*

Penn Treebank II's *Bracketing Guidelines*: "Unknown, uncertain or unbracketable. X is often used for bracketing typos and in bracketing *the ... the ...* constructions"

The more ... the more ...



Difficult syntactic problems

- This is actually a partial analysis in the guise of a full analysis
- Can also be achieved by automatic fragment analysis
- Rather than devising *ad hoc* structures, we think it is fairer to admit that certain constructions are simply not covered yet

Other parsing problems

Very long sentences can be a challenge to the parser

Possible solutions:

- Pruning the search space (cooperation with Aoife Cahill)
- Manual pre-bracketing if needed

Conclusion

Why cannot (in some cases should not) everything in a corpus be treated as a sentence with a full syntactic structure?

- Non-syntactic input
- Performance phenomena

Some cases we may want to annotate, but in a theoretically motivated way, by careful incremental grammar development

- Creative use of language
- Difficult syntactic problems

Conclusion

- Parsebanking is a feasible approach with good methods and tools
- Grammar fully consistent with parsebank, can be used directly for applications
- Formal coherence between grammar and lexicon, different projections
- Avoids errors and inconsistencies due to manual annotation
- Eliminates postprocessing for error detection and compaction
- May yields better quality, theoretically motivated grammars
- Coverage will never be perfect, but is it necessary/desirable?

However, regardless of approach you will not be able to find structures which are not annotated, e.g. if people use *clefts* in ways not annotated as such in the grammar, you cannot search for them.

Links

http://gandalf.aksis.uib.no/trepil

http://iness.uib.no