Treebanks, Linguistic Theories and Applications

Introduction to Treebanks

Lecture One

Petya Osenova and Kiril Simov

Sofia University “St. Kliment Ohridski”, Bulgaria
Bulgarian Academy of Sciences, Bulgaria

ESSLLI 2018

Plan of the Lecture

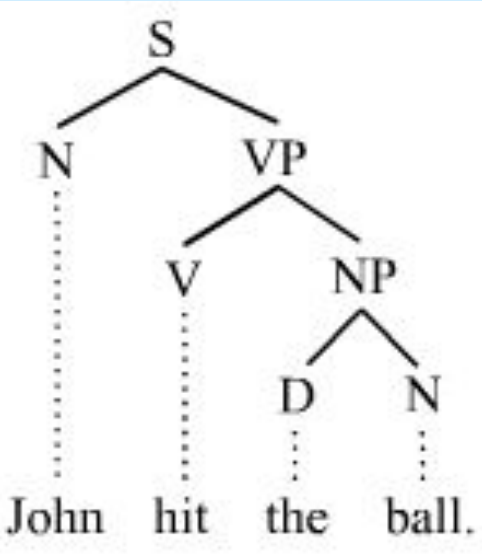
- Definition of a treebank
- The place of the treebank in the language modeling
- Related terms: parsebank, dynamic treebank
- Prerequisites for the creation of a treebank
- Treebank lifecycle
- Theory (in)dependency
- Language (in)dependency
- Tendences in the treebank development

Treebank Definition

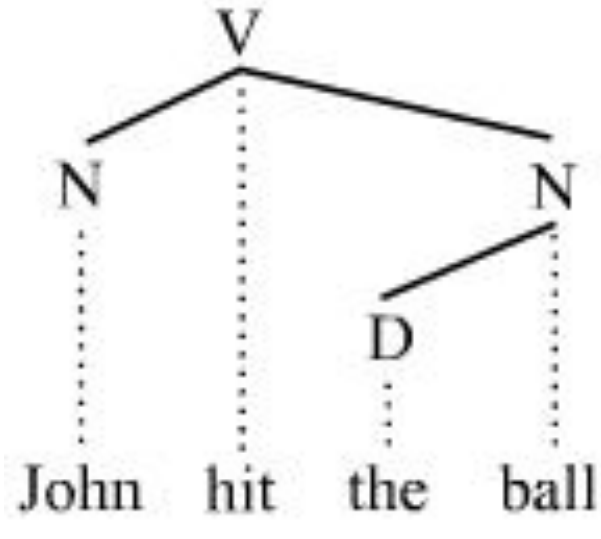
A corpus annotated with syntactic information

- The information in the annotation is added/checked by a trained annotator - manual annotation
- The annotation is complete - no unannotated fragments of the text
- The annotation is consistent - similar fragments are analysed in the same way
- The primary format of annotation - syntactic tree/graph

Example Syntactic Trees from Wikipedia



Constituency-based parse tree



Dependency-based parse tree

The two main approaches to modeling the syntactic information

Pros vs. Cons (*Handbook of NLP*, p. 171)

Constituency

- Easy to read
- Correspond to common grammatical knowledge (phrases)
- Introduce arbitrary complexity

Dependency

- Flexible
- Also correspond to common grammatical knowledge (grammatical functions)
- Consistent criteria for head identification needed

Trees and Graphs

- A graph: a set of vertices (nodes) and edges (arcs)
- A tree: a connected graph in which each node (except one) has exactly one parent node
- Labels - the nodes and arcs can be labeled

The term Treebank is based on the assumption that the syntactic information is represented as a tree

The reality shows that usually the syntactic information requires graph representation

Examples of Annotations in Treebanks (1)

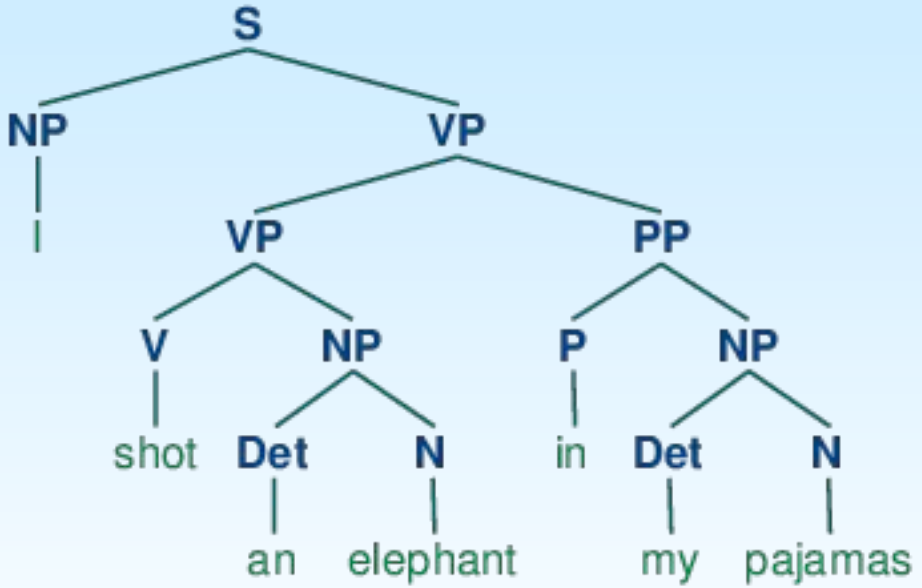
Penn Treebank Bracketing

```
( (S
    (NP-SBJ
      (NP (DT The) (NN tender) (NN offer) )
      (PP (IN by)
        (NP
          (NP (NNP West) (NNP Germany) (POS 's) )
          (JJ second-biggest) (JJ commercial) (NN bank) )))
    (VP (VBZ is)
      (PP-PRD (IN in)
        (NP (CD two) (NNS stages) )))
    (. .) ))
```

Examples of Annotations in Treebanks (2)

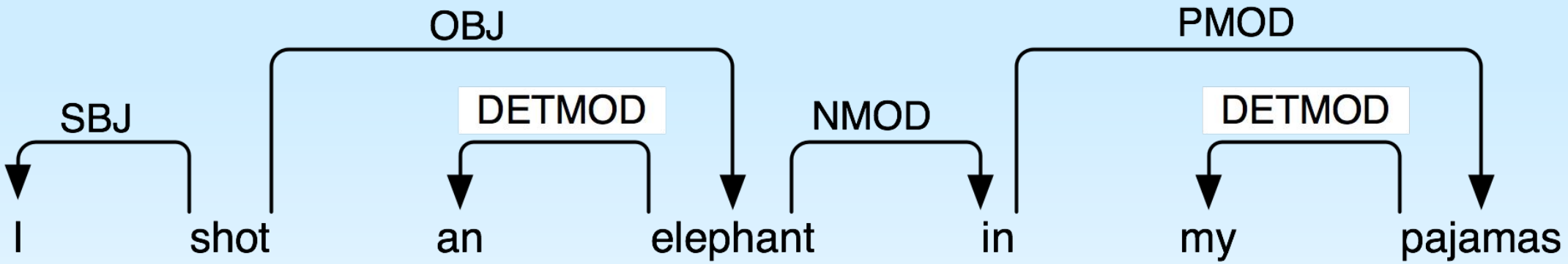
```
(S
  (NP I)
  (VP
    (VP (V shot) (NP (Det an) (N elephant)))
    (PP (P in) (NP (Det my) (N pajamas)))))
```

Graphical Representation (NLTK)



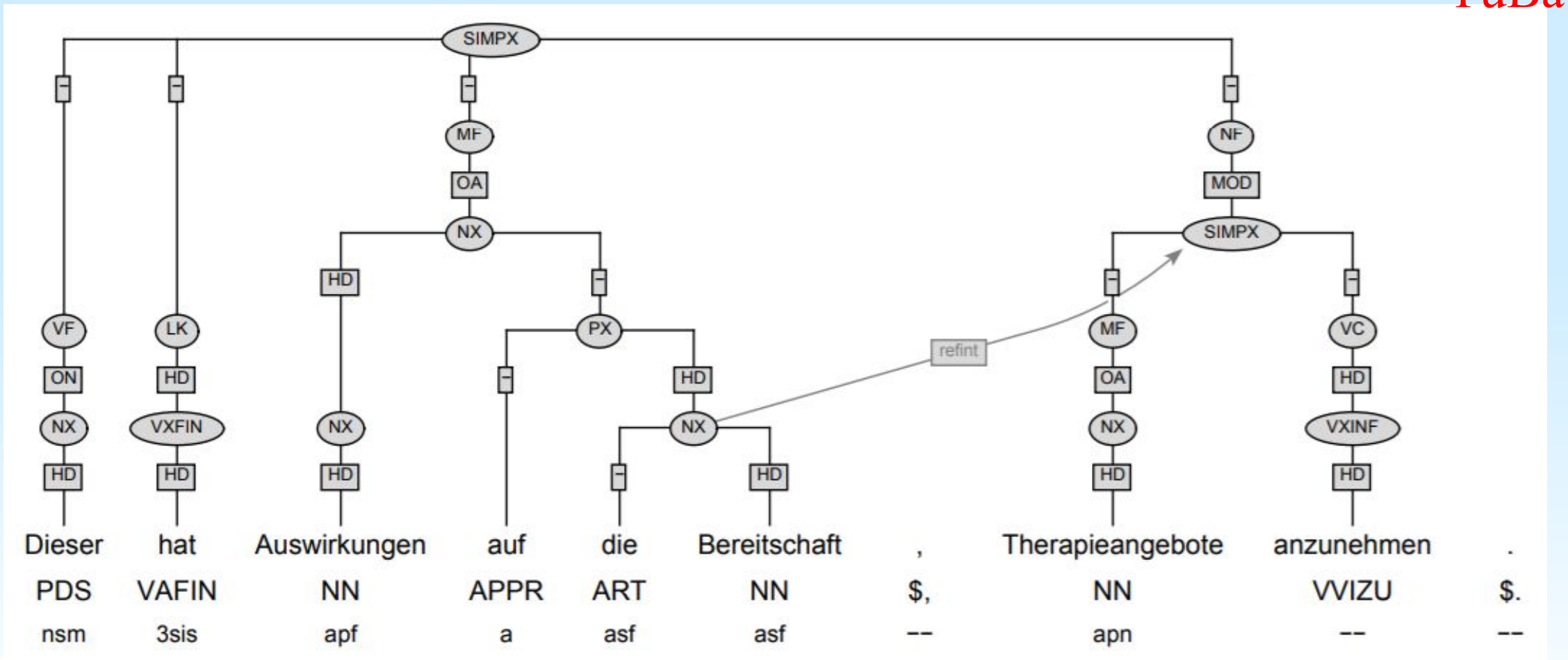
Examples of Annotations in Treebanks (3)

Dependency Tree

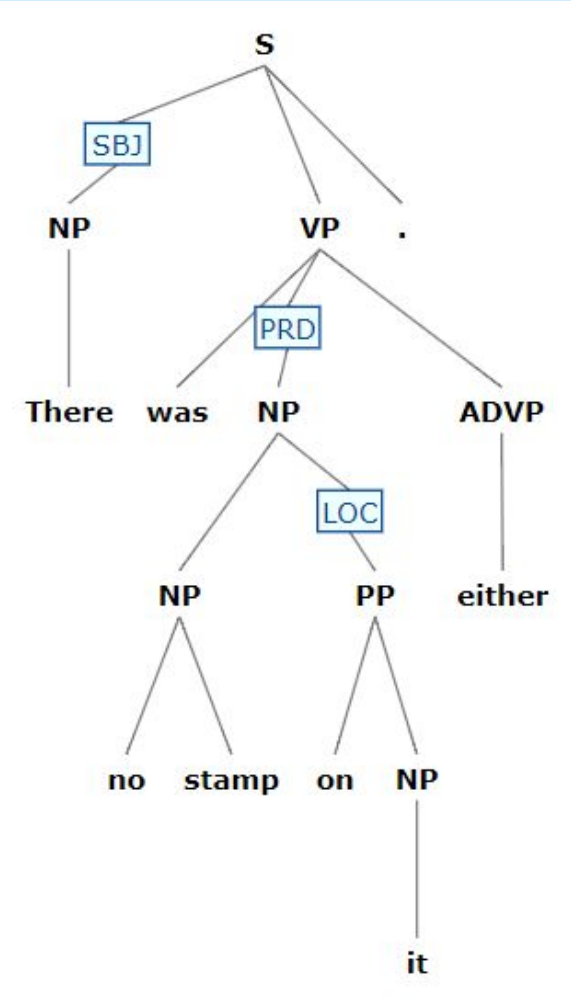


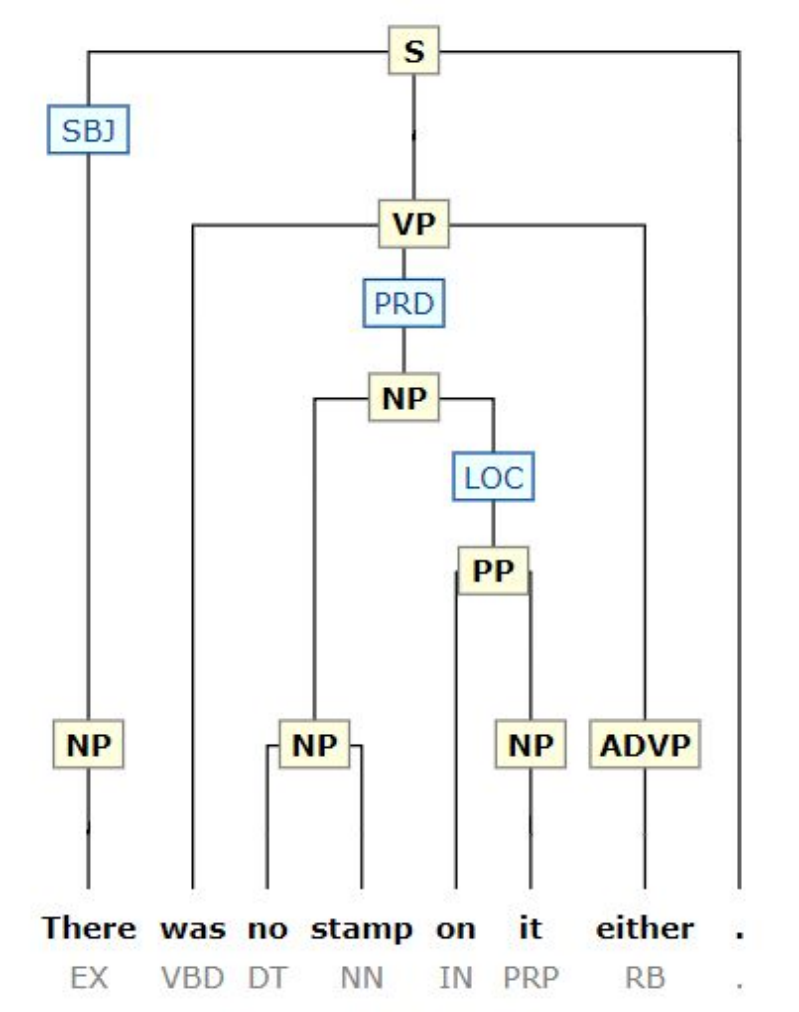
Examples of Annotations in Treebanks (4)

TüBa-D/Z



Examples of Annotations in Treebanks (5)

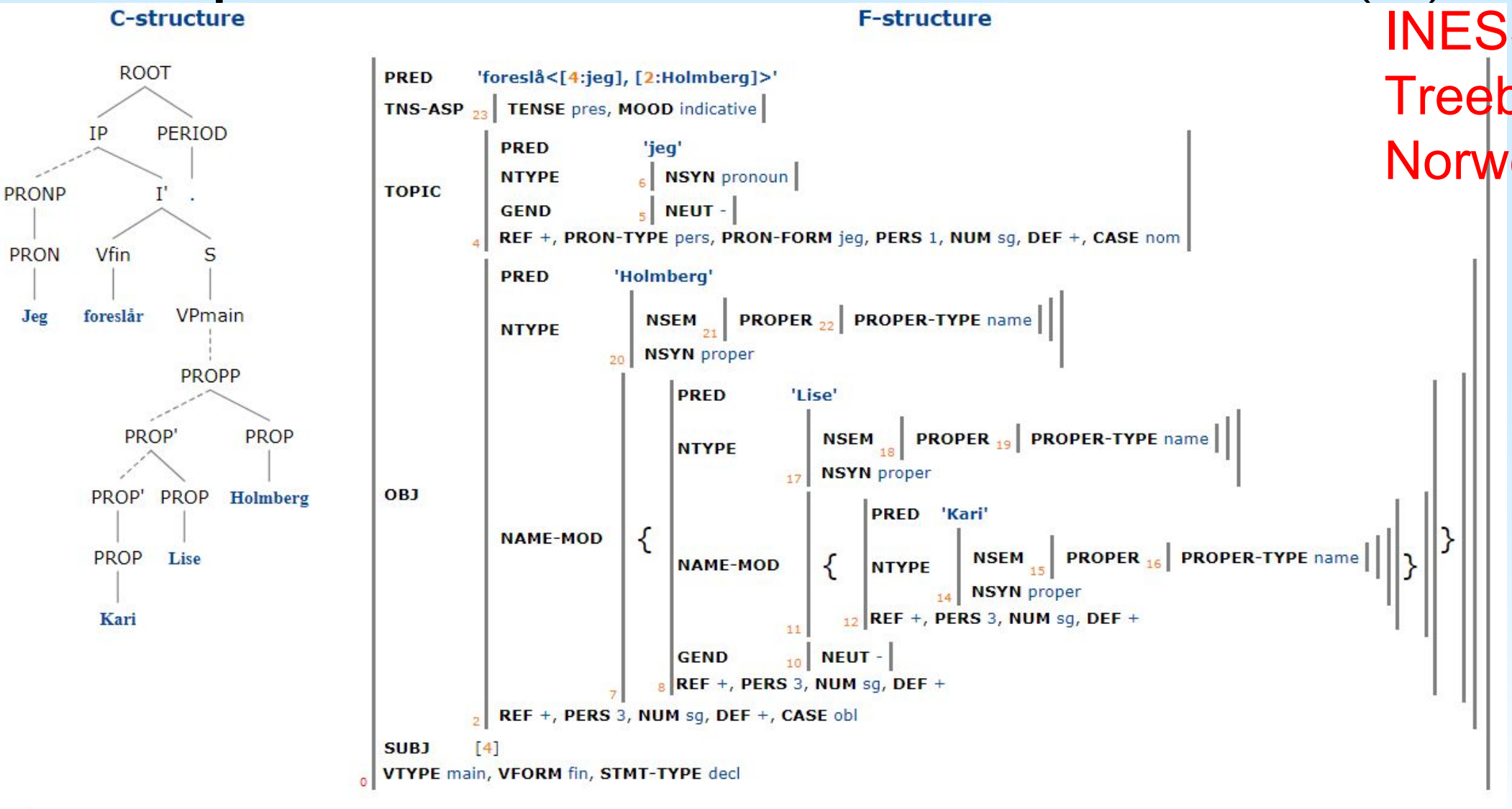




INESS - Sophie's World

Examples of Annotations in Treebanks (6)

INESS - LFG Treebank for Norwegian



Modeling of Language Knowledge

- Language as a natural phenomenon - language object and relations between them defined by grammaticality
- Language model - mathematical abstraction over language comprises mathematical objects and relations among them
- Linguistic theory - it constrains the linguistic model to correspond to the language
- A treebank is a subset of a language model with respect to unknown linguistic theory

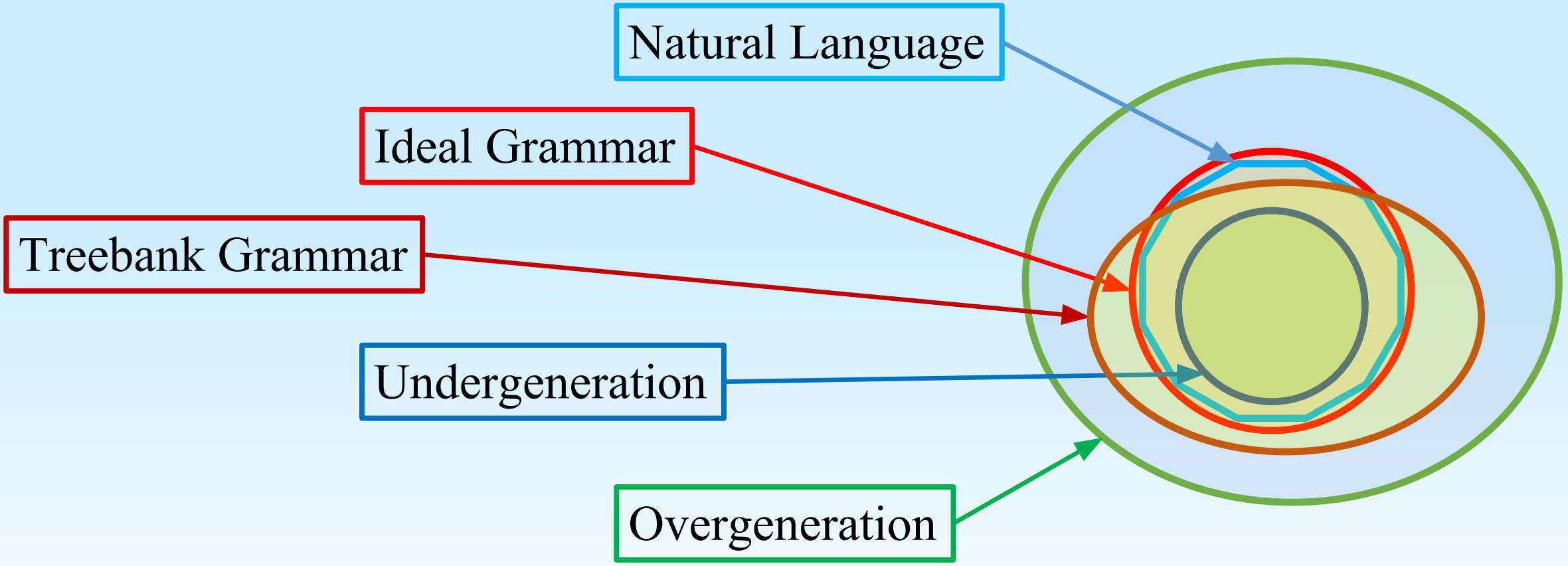
Language Model

- Language Model (LM) as an abstraction over Natural Language (NL) is usually represented as a set of trees
- A sentence representation in LM is a sequence of words associated with a tree
- A treebank is a set of graphs representing sentences from NL

Ideal Grammar and Treebank Grammar

- A Formal Grammar (FG) is a formal device that for each sentence predicts whether it is grammatical in the natural language or not assigning a representation of the sentences
- An Ideal Grammar is a grammar Γ that predicts/recognises the language with minimum over- or undergeneration with respect to the natural language
- We have not invented the Ideal Grammar Γ yet
- A Treebank Grammar is a formal grammar that generalizes over the sentences in the treebank

Natural Language, Grammars, Treebanks



Treebank in a Grammar Formalism

A treebank C in a given grammatical formalism G is a sequence of analyzed sentences where each analyzed sentence is a member of the set of structures defined as a *strong generative capacity* of a grammar Γ in this grammatical formalism:

$$\forall\ S.\ S \in C \Rightarrow S \in SGC(\Gamma)$$

and

$$\forall\ S.\ S \in C \Rightarrow \forall S'.(S' \in \Gamma(\boldsymbol{\delta}(S)) \Rightarrow S' \in C)$$

Annotation Schema

- The annotation schema for a treebank can be considered a treebank grammar
  - The annotation schema could be an implemented grammar like English Resource Grammar (HPSG) or NorGram (LFG)
  - The annotation schema could be a set of tools for preprocessing of the sentences and supporting the manual annotation
- Thus:
  - Tree selection vs. Tree creation

Treebank Lifecycle

- Design phase
- Implementation phase
- Validation phase
- Documentation phase
- Exploitation phase
- Reclassification phase

Design Phase

The *design phase* determines:

- The goal of the treebank
- The width and depth of the linguistic knowledge, and
- The treebank format

The result from the design phase is a stylebook describing the underlying annotation scheme for the treebank

- In case of the tree selection annotation it describes discriminators used for selection of the correct trees
- In case of the tree creation annotation it describes the treatment of language phenomena

Implementation Phase

During the implementation phase the following tasks are addressed:

- Identifying the sources of information, which support the annotation process;
- The architecture of the processing, the software tools and their environment;
- The organization of the actual annotation

The sources of information can be: text corpora, partial grammars, lexicons, general linguistic analyses (grammar books)

Validation Phase

The *validation phase* requires consistency checks over the whole treebank. These checks determine the conformance of the treebank with:

- The principles of the style book, and
- The agreement among annotators on their decisions when the style book is too general or does not provide any solutions to certain cases

Documentation Phase

The *documentation phase*:

- Registers the current status of the treebank and
- Enables both: its further development and its usage

Documentation of the treebank includes the stylebook of analyzed phenomena types, extended with the description of the whole process of annotation, the preprocessing steps, etc.

Exploitation Phase

The *exploitation phase* includes the actual usages of the treebank for different tasks such as:

- Linguistic research
- Language resources creation
- NLP processing tasks
- Others

Reports on usefulness of the treebank

Reclassification Phase

The reclassification phase revises the current state of the treebank based on the results from the validation and exploitation phases.
It includes:

- Creation of a new annotation schema (redesign)
- Annotation of new data
- Adaptation (reclassification) of the represented linguistic knowledge

Encoded Linguistic Knowledge

Besides the syntactic information the treebanks encodes other linguistic knowledge that plays important role in the implementation and exploitation phases such as:

- Morphosyntactic information
- Named Entities
- Coreferential chains
- Semantic roles
- Senses

Prerequisites for the Creation of a Treebank

Despite of being grammar-based, or NLP-based, it has to handle:

- Tokenization/segmentation (words, numbers, symbols, punctuation, etc.)
- Sentence boundary
- Named Entity Recognition (names and abbreviations)
- Morphology (especially for morphologically rich languages)
- Lemmatization (detection of the base form)
- MWEs (fixed, idioms, collocations, etc.)

Sentence Boundary

It accepted as an easy task, but there are many problematic cases like:
“It’s going to be difficult. Croatia have a generation of outstanding players who have just won,” Sampaoli said.

vs.

“When he has two or three opponents trying to block him, somewhere on the pitch a teammate is free, as happened against Iceland,” he said. “We need to take advantage of that.”

What is a Syntactic Annotation?

- Primarily, texts, annotated with syntactic annotation with the help of a certain language grammar model

- The syntactic information might include:
  - Syntactic domains - NP, VP, PP…
  - Functional labels - Pragmatic, DiscA…
  - Semantic roles, discourse information….

Syntactic Domains

- Full syntactic analysis vs. Chunking (Abney 1987)
  - "Islands of certainty" or "ill-formed sentences include well-formed chunks"
  - Non-recursive structures;
  - Precision is favoured to coverage
  - Delayed attachment decisions

[*The man*] *observed* [*the boy*] *with* [*the telescope*] *in* [*the garden.*]

Treebank, Parsebank, Dynamic Bank

Treebank: manually constructed resource with syntactically analysed texts.

Parsebank: automatically parsed texts whose output are syntactic structures.

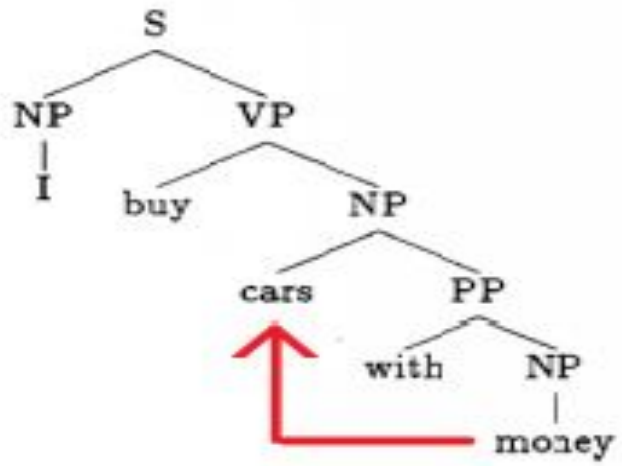
Dynamic Bank: simultaneous construction of a grammar and a treebank in a continuous loop of grammar correction and re-parsing.

Challenges

- Syntactic Domains and Ambiguity: differ per language and theory
- Linguistic Phenomena: differ per language and theory
- Relation to other linguistic levels: monostratal or multistratal representation

Ambiguity

Example from Ratnaparkhi (1999)



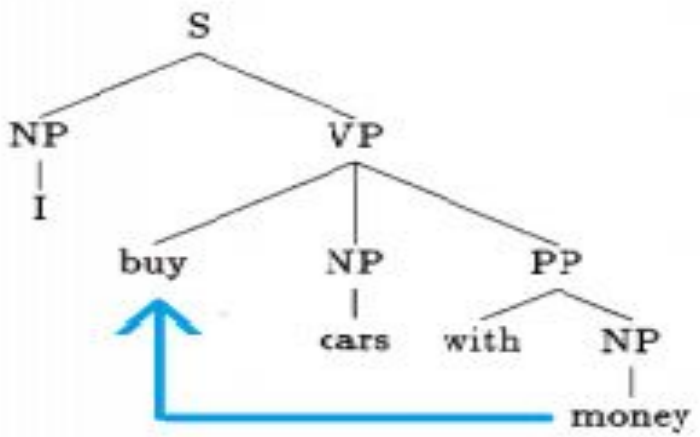


Figure 3: Unlikely parse (left); Likely parse (right)

- *Money* refers to *buy* (verb-phrase), not *cars* (noun-phrase)
- Both parses are legal, but we want the one that is more likely

source: https://cs.uwaterloo.ca/~mli/statistical-parsing

Annotation with a Precise Grammar

The annotation is done by the following steps:

- A new sentence is processed by the grammar (different from parser)
- The result is usually many different analyses (or none)
- Discriminators are determined (for example agreement)
- The annotator selects the correct analysis on the basis of the correct discriminators
- The correct analysis and the selected discriminators are stored

Annotation without a Grammar

- Pre-processing steps are performed. They includes:
  - The steps mentioned above (tokenization, POS tagging, …)
  - Partial syntactic analysis
- The Annotation Scheme is encoded in the annotation tool
- Annotators annotate (semi)automatically the text according to the Annotation Scheme
- The annotator could modify all suggested information
- Application of constraints (rules) to check the consistency of the result

Constituency vs. Dependency Linguistic Models

Constituency: phrase-based

Dependency: word-based

Both: monostratal and multistratal

Both: the notion of HEAD and related issues (syntactic vs. semantic head, non-headed phrases). Types: NP, VP, PP, AP, AdvP, (NumP, DetP)

Constituency

Classic: PenTreeBank

Constraint-based:

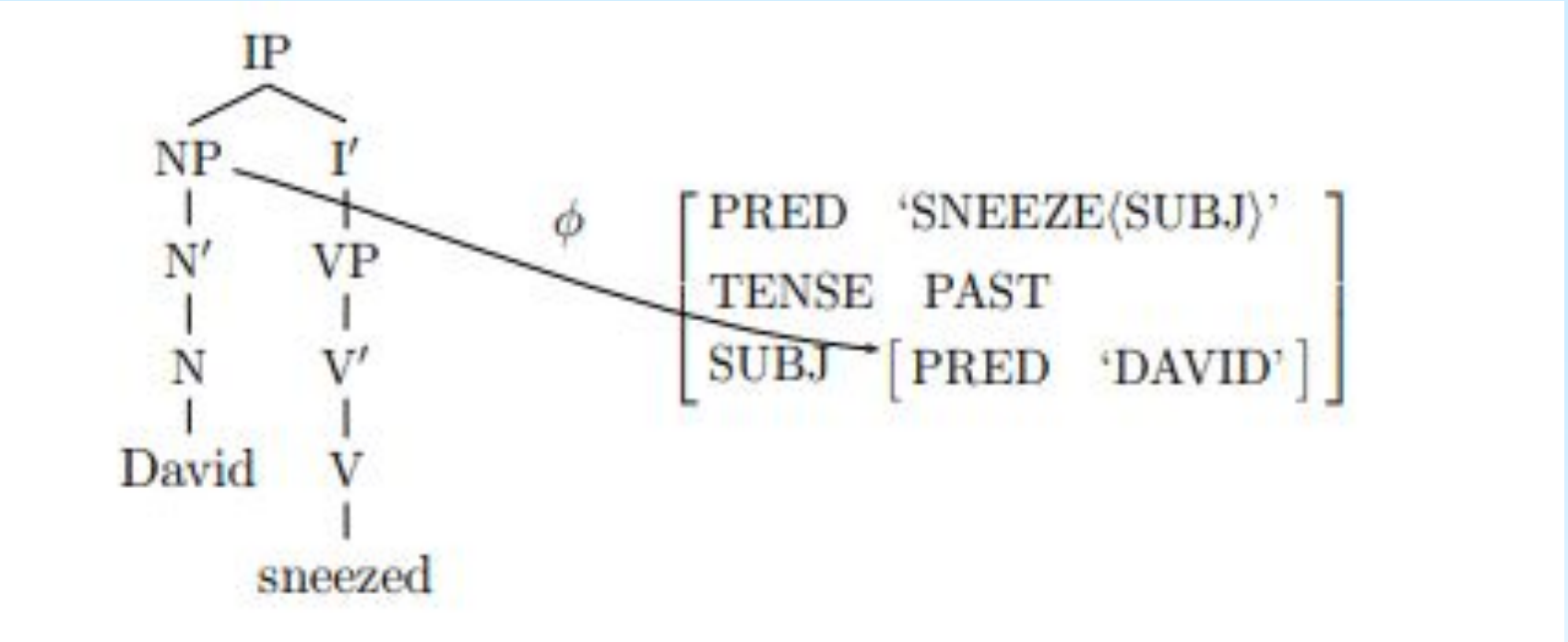
- Lexicalized: HPSG (LinGO Redwoods, Deepbank), LFG (NorGramBank)
- Categorial Grammar (Thai Categorial Grammar Treebank)

Penn Treebank

```
(S (NP-SBJ Sandy)
   (VP (ADVP-MNR sneakily)
       threw
       (NP a curve)))
```

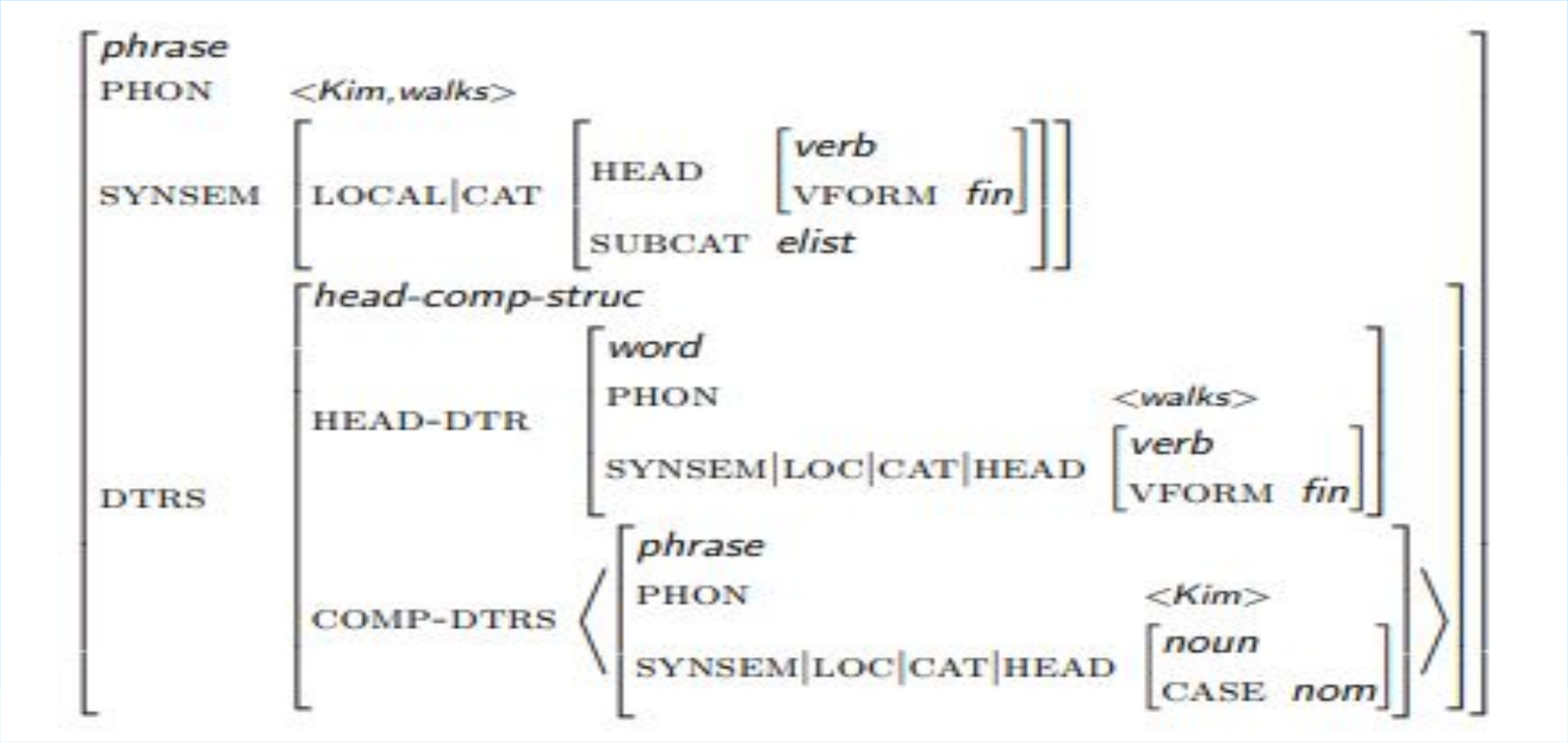
(source: http://cs.jhu.edu/~jason/465/hw-parse/treebank-manual.pdf)

LFG Representation



(source: http://users.ox.ac.uk/~cpgl0015/lfg.pdf)

HPSG Representation



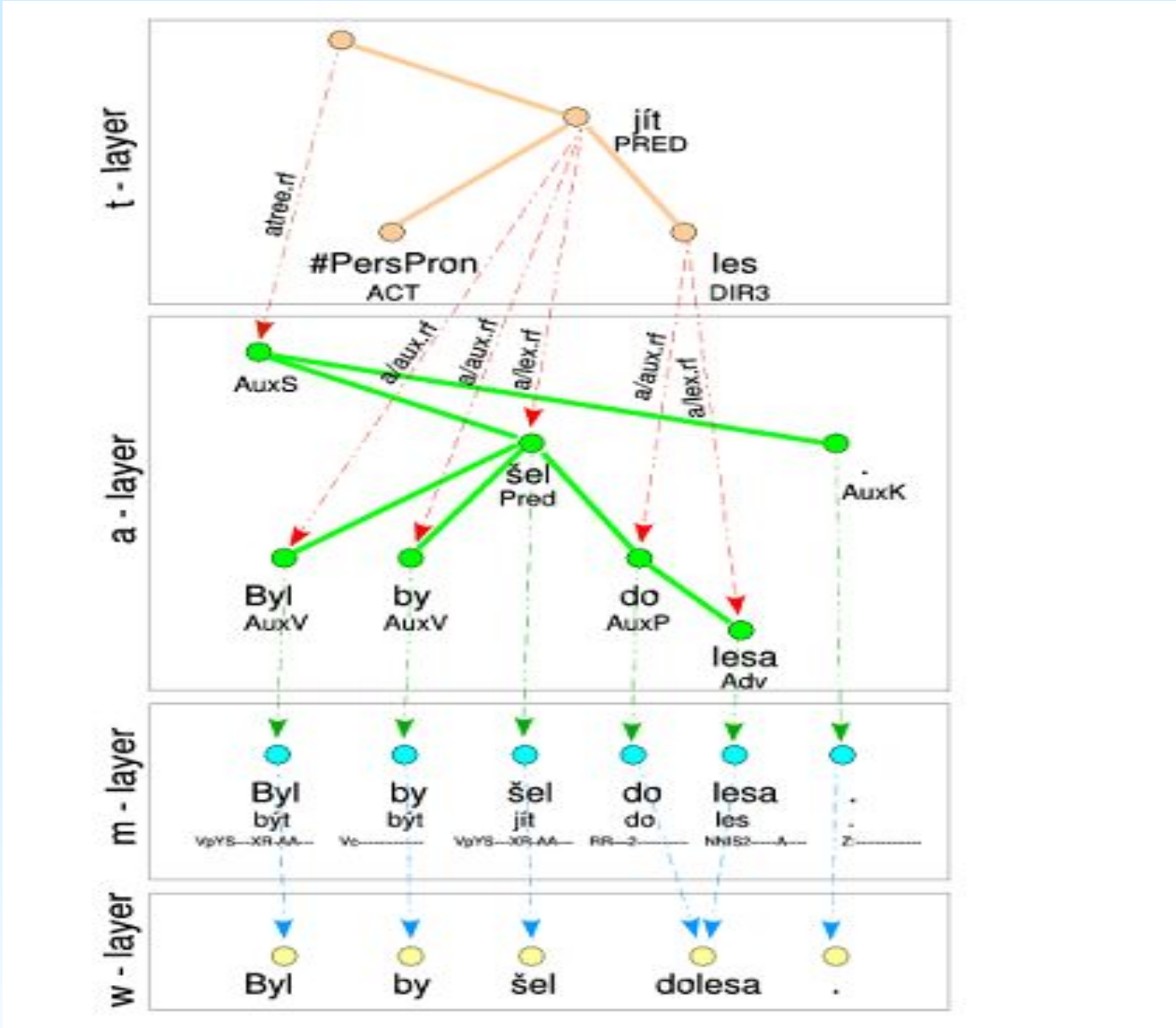
(source: http://cl.indiana.edu/~md7/15/614/slides/06-hpsg/06-hpsg.pdf)

Dependency-based Treebanks

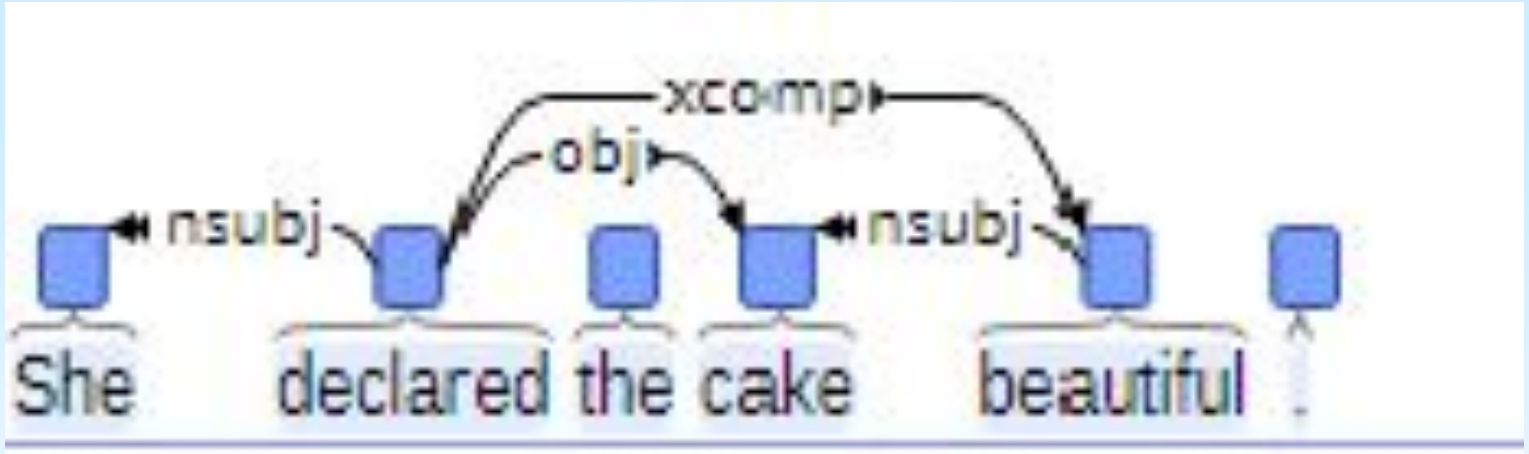
Generative Functional Grammar: Prague Dependency Treebank

Universal Dependencies: Multilingual setting

Prague Dependency Treebank



Universal Dependency



Theory (in)dependency: Myths and Reality

- Myth: A treebank can be theory-neutral
- Reality: A treebank is always theory-dependent
- Myth: We can always convert safely the resource from one theory to another
- Reality: We do that with problems and additional work. Information is lost or distorted
- Myth: Theory complexity does not influence the treebank presentation
- Reality: Sometimes the theory is powerful but too complicated for humans

Theory (in)dependency / Language (in)dependency

The term ‘theory neutral’ does not mean without any theory behind the annotation scheme.

It means that an underlying level of annotation is developed that is applicable to ‘any’ language (or at least to languages that are typologically close)

(From “*Handbook of Natural Language Processing*”, p. 180)

Tendencies in Treebank Development

- From constituency to dependency
- From small treebanks to huge treebanks (treebanks -> parsebanks)
- From syntactic resources to semantic and discourse ones
- From monolingual perspective to cross-lingual and multi-lingual ones

Initiatives in Treebank Development

- Universal Dependencies: a framework for cross-linguistically consistent grammatical annotation
- CLARIN-related:
  - INESS: integrated environment for building, accessing, searching and visualizing treebanks.
  - GrETEL: Greedy Extraction of Trees for Empirical Linguistics. It is a user-friendly search engine for the exploitation of syntactically annotated corpora or treebanks.

Some references

Handbook of Natural Language Processing 2010 (second edition), Nitin Indurkhya, Fred J. Damerau (eds.). CRC Press.
Treebanks: Building and Using Parsed Corpora 2003. Anne Abeille (ed.). Kluwer Academic Publishers.