# IN2110: Språkteknologiske metoder

## *Dependenssyntaks*

Lilja Øvrelid

Språkteknologigruppen (LTG)

(with thanks to Stephan Oepen and Joakim Nivre)

24 mars, 2025



# From Linear Order to Hierarchical Structure

- NLP approaches we have considered this far:
  - Distributional representations of documents or words:

    *Cisco acquired Tandberg* $\equiv$ *Tandberg acquired Cisco*

  - Sequence labeling: HMMs.
    - One layer of abstraction: BIO-labels as hidden states.
    - Still only sequential in nature.

# From Linear Order to Hierarchical Structure

- NLP approaches we have considered this far:
  - Distributional representations of documents or words:

    *Cisco acquired Tandberg* $\equiv$ *Tandberg acquired Cisco*

  - Sequence labeling: HMMs.
    - One layer of abstraction: BIO-labels as hidden states.
    - Still only sequential in nature.
- Syntax adds hierarchical structure:
  - In NLP, being a sub-discipline of AI, we want our programs to *'understand'* natural language (on some level).
  - Finding the grammatical structure of sentences is an important step towards 'understanding'.
  - Shift focus from bags or sequences to hierarchical structure.

# Syntax

- Syntax: study of the structure of sentences
- "Who does what to whom?"
- Wealth of theories: some differences, a lot in common
  - Government and Binding (GB)
  - Minimalist Program (MP)
  - Head-driven phrase structure grammar (HPSG)
  - Lexical Functional Grammar (LFG)
  - Categorial Grammar
  - Dependency Grammar
  - ...

# Why bother?

- Theoretical syntacticians concerned with **grammaticality**
  - *The President nominated a new Supreme Court justice*
  - **President the new Supreme justice Court nominated*
- Relevant for some NLP applications:
  - text generation
  - grammar checking
- But mostly want systems that are robust and can handle realistic (noisy) language

# Why bother?

- Theoretical syntacticians concerned with **grammaticality**
  - *The President nominated a new Supreme Court justice*
  - **President the new Supreme justice Court nominated*
- Relevant for some NLP applications:
  - text generation
  - grammar checking
- But mostly want systems that are robust and can handle realistic (noisy) language

```
Great job @justinbieber!  Were SOO PROUD of what youve
accomplished!  U taught us 2 #neversaynever & you
yourself should never give up either <3
```

# Why bother?

- Parsing provides “scaffolding” for semantic analysis
- Direct, down-stream usage of syntactic information
  - opinion mining
  - information extraction
  - syntax-informed statistical machine translation
  - sentence compression
  - etc.

# Today's lecture

- Very brief repetition of basic principles of syntax:
  - form vs function
  - constituents and phrases
  - context-free grammars
- Dependency Grammar
  - basic concepts: head, dependent
  - comparison to constituent structure
  - formal properties
- Treebanks

# Constituents

- The words in a sentence are organized into groupings
- function as a whole
- relate to other words as a unit
  - The dog ate my homework
- linguistic tests of constituency
  - The dog ate it

# Constituents

- The words in a sentence are organized into groupings
- function as a whole
- relate to other words as a unit
  - The dog ate my homework
- linguistic tests of constituency
  - The dog ate it
  - My homework the dog ate

# Form and function

- **Syntactic form** - constituents are described using parts of speech and phrases
  - phrases - larger constituents above word level
  - phrases named after the **head** - central, obligatory member
  - e.g. NP, VP

# Form and function

- **Syntactic form** - constituents are described using parts of speech and phrases
  - phrases - larger constituents above word level
  - phrases named after the **head** - central, obligatory member
  - e.g. NP, VP

| NP | VP | |
|---|---|---|
| NP | | NP |
| The dog | ate | my homework |

- **Syntactic function** - constituents are described by their role in the sentence as a whole
  - Subject
  - (Direct and Indirect) Object
  - Adverbial

# Form and function

- **Syntactic form** - constituents are described using parts of speech and phrases
  - phrases - larger constituents above word level
  - phrases named after the **head** - central, obligatory member
  - e.g. NP, VP

| NP | VP | |
|---|---|---|
| NP | | NP |
| The dog | ate | my homework |

- **Syntactic function** - constituents are described by their role in the sentence as a whole
  - Subject
  - (Direct and Indirect) Object
  - Adverbial

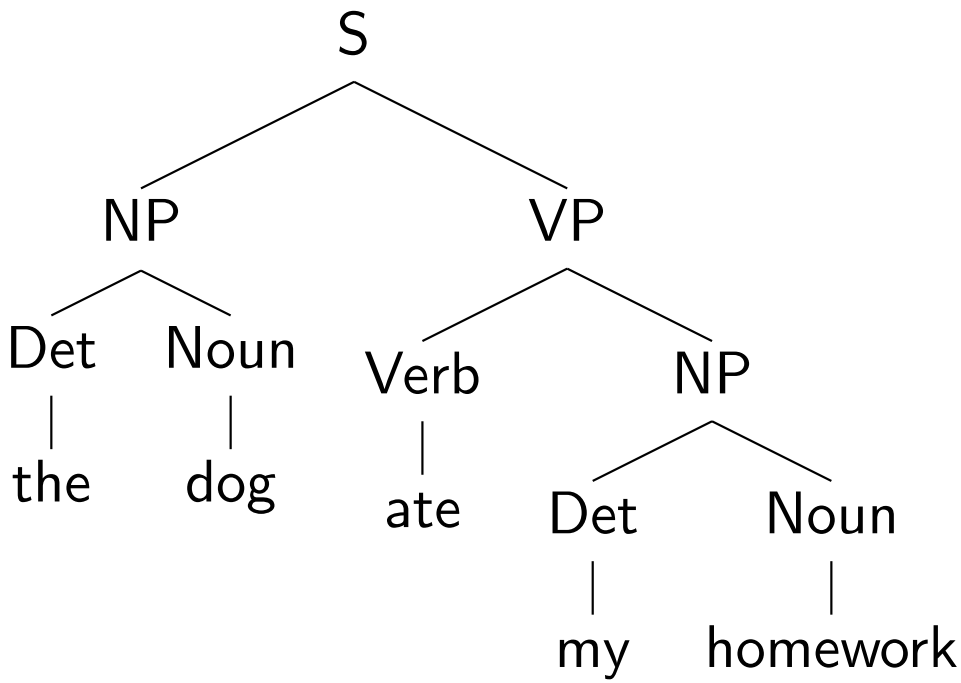
| Subject | Predicate | Object |
|---|---|---|
| The dog | ate | my homework |

# Phrase structure grammar (PSG)

- Capture constituent status and ordering
- Formal model: context-free grammar
  1. S $\rightarrow$ NP VP
  2. NP $\rightarrow$ D N
  3. VP $\rightarrow$ V NP
- Syntactic structure as phrase structure **trees**

# Syntactic categories

- Phrase Structure (PS) tree

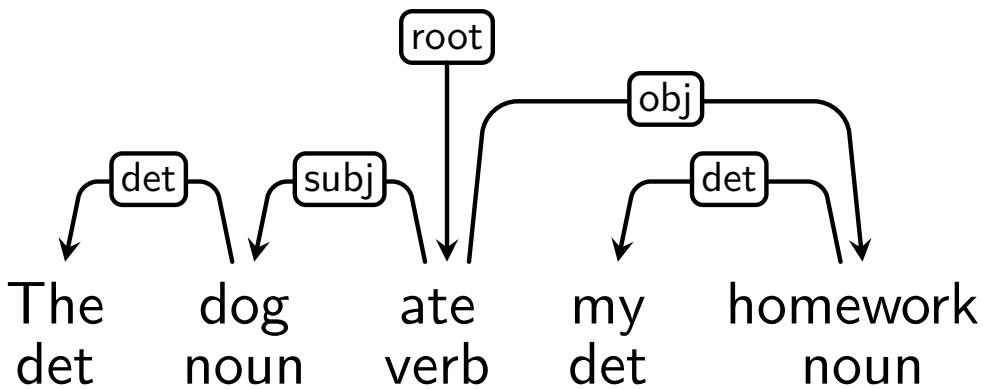


# Today's lecture

- Very brief repetition of basic principles of syntax:
  - form vs function
  - constituents and phrases
  - context-free grammars
- Dependency Grammar
  - basic concepts: head, dependent
  - comparison to constituent structure
  - formal properties
- Treebanks

# Now: Dependency Grammar (DG)

- An alternative to phrase structure representations
- Syntactic functions are central
- Claimed to be closer to semantic analysis
- The basic idea:
  - Syntactic structure consists of lexical items, linked by binary asymmetric relations called dependencies.



# Dependency Grammar

Dependency grammar is important for those interested in NLP:

- Increasing interest in dependency-based approaches to syntactic parsing in recent years (e.g., CoNLL shared tasks)
- Currently dominant approach
- Downstream applications: relation extraction, question answering, ontology learning, sentiment analysis, etc.

# Constituency vs. Relations

- DG is based on relationships between words, i.e., **dependency relations**
  - A $\rightarrow$ B means *A governs B* or *B depends on A* ...
  - Dependency relations can refer to syntactic properties, semantic properties, or a combination of the two
  - These relations are generally things like subject, object/complement, (pre-/post-)adjunct, etc.
    - Subject/Agent: *John* fished.
    - Object/Patient: Mary hit *John*.
- PSG is based on groupings, or constituents
  - Grammatical relations are not usually seen as primitives, but as being derived from structure

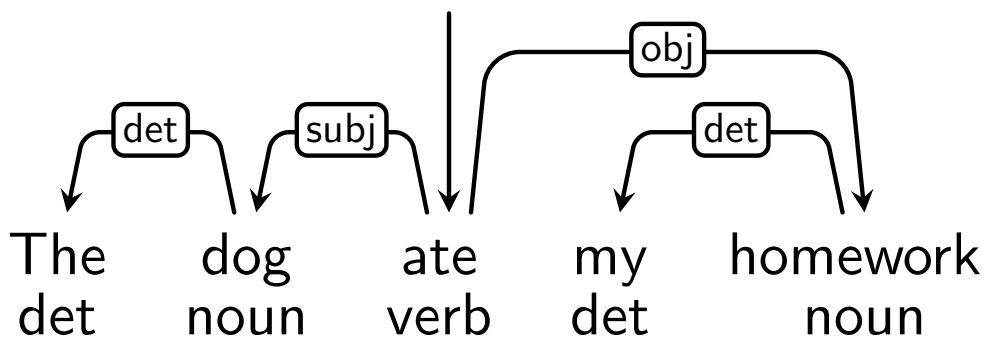
# Simple relation example

For the sentence *The dog ate my homework*, we have the relations:

- ate $\rightarrow_{\text{subj}}$ The dog
- ate $\rightarrow_{\text{obj}}$ my homework

Both *The dog* and *my homework* depend on *ate*, which makes *ate* the head, or **root**, of the sentence (i.e., there is no word that governs *ate*)

- The structure of a sentence, then, consists of the set of pairwise relations among words.

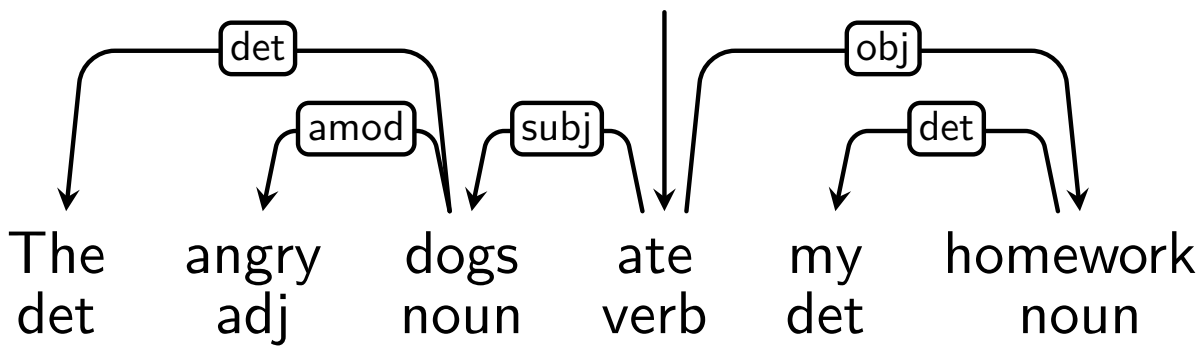


# Comparison

- Dependency structures explicitly represent
  - head-dependent relations (directed arcs),
  - functional categories (arc labels),
  - possibly some structural categories (parts-of-speech).
- Phrase structures explicitly represent
  - phrases (nonterminal nodes),
  - structural categories (nonterminal labels),
  - possibly some functional categories (grammatical functions).

# Criteria for Heads and Dependents

- Criteria for a syntactic relation between a head $H$ and a dependent $D$ in a construction $C$:
  1. $H$ determines the syntactic category of $C$; $H$ can replace $C$.
  2. $H$ determines the semantic category of $C$; $D$ specifies $H$.
  3. $H$ is obligatory; $D$ may be optional.
  4. The form of $D$ depends on $H$ (agreement or government).
  5. The linear position of $D$ is specified with reference to $H$.

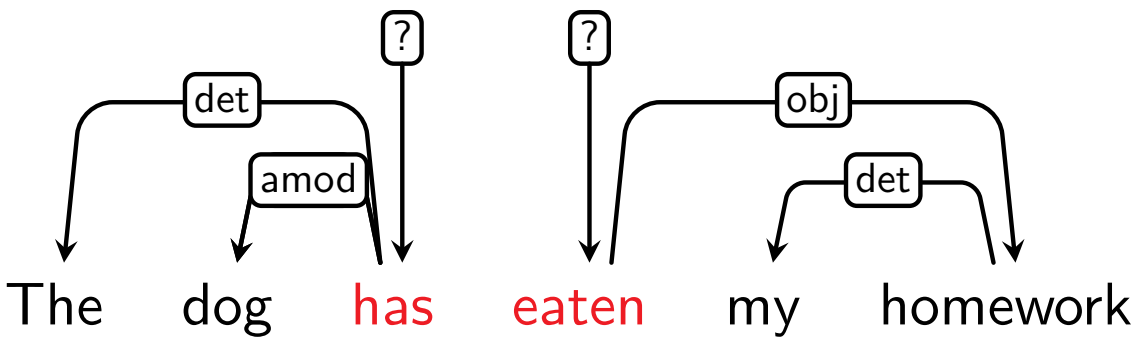


# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

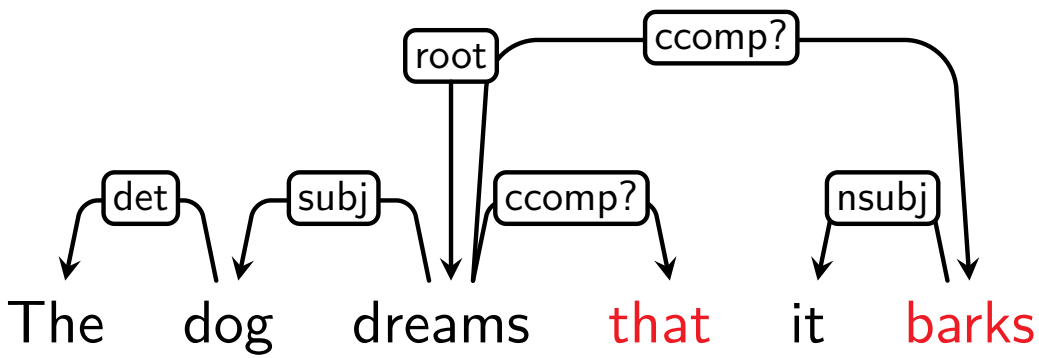


# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

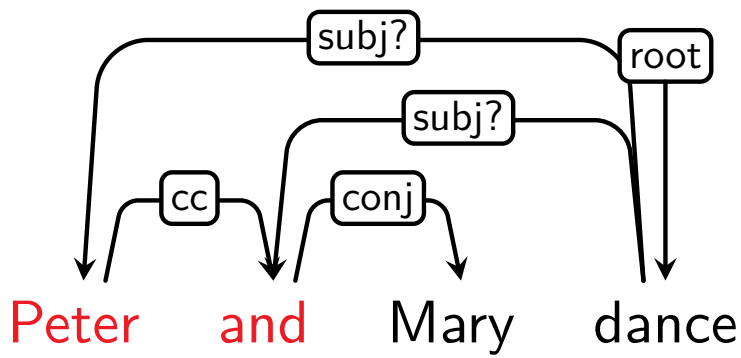


# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

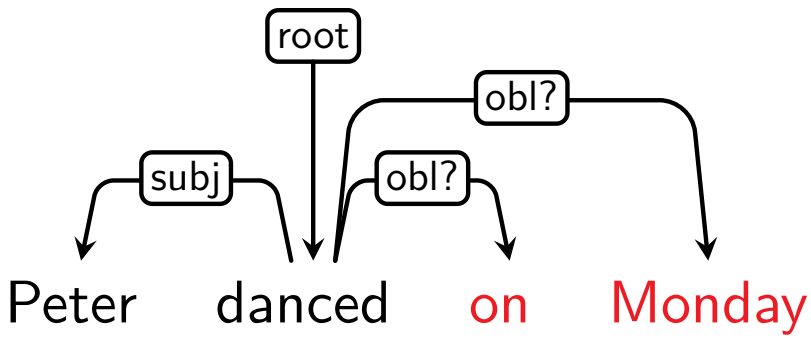


# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation



# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

# Some Tricky Cases

- Complex verb groups (auxiliary $\leftrightarrow$ main verb)
- Subordinate clauses (complementizer $\leftrightarrow$ verb)
- Coordination (coordinator $\leftrightarrow$ conjuncts)
- Prepositional phrases (preposition $\leftrightarrow$ nominal)
- Punctuation

# Dependency Graphs

- A dependency structure can be defined as a directed graph $G$, consisting of
  - a set $V$ of nodes,
  - a set $E$ of arcs (edges),
- Labeled graphs:
  - Nodes in $V$ are labeled with word forms (and annotation).
  - Arcs in $E$ are labeled with dependency types.
- Notational conventions ($i, j \in V$):
  - $i \rightarrow j \;\; \equiv \;\; (i, j) \in E$

# Formal Properties of Dependency Graphs

- **antisymmetric**: if A $\rightarrow$ B, then B $\nrightarrow$ A
  - If A governs B, B does not govern A
- **antireflexive**: if A $\rightarrow$ B, then B $\neq$ A
  - No word can govern itself.
- **antitransitive**: if A $\rightarrow$ B and B $\rightarrow$ C, then A $\nrightarrow$ C
  - These are *direct* dependency relations
  - cf. *a usually reliable source*: source $\rightarrow$ reliable & reliable $\rightarrow$ usually, but source $\nrightarrow$ usually
- **labeled**: $\forall \rightarrow$, $\rightarrow$ has a label $(r)$

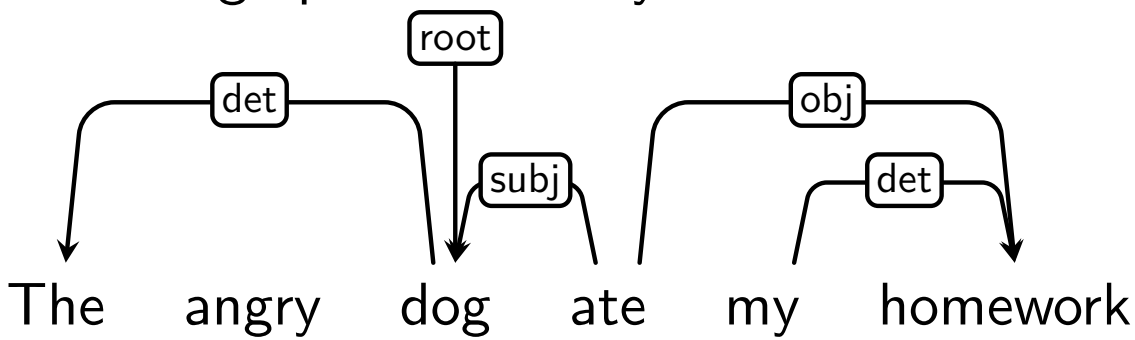
# Formal Conditions on Dependency Graphs

- $G$ is (weakly) connected:
  - For every node $i$ there is a node $j$ such that $i \rightarrow j$ or $j \rightarrow i$.
- $G$ is acyclic:
  - If $i \rightarrow j$ then not $j \rightarrow^* i$.
- $G$ obeys the single-head constraint:
  - If $i \rightarrow j$, then not $k \rightarrow j$, for any $k \neq i$.

# Formal Conditions on Dependency Graphs

- $G$ is (weakly) connected:
  - For every node $i$ there is a node $j$ such that $i \rightarrow j$ or $j \rightarrow i$.
- $G$ is acyclic:
  - If $i \rightarrow j$ then not $j \rightarrow^* i$.
- $G$ obeys the single-head constraint:
  - If $i \rightarrow j$, then not $k \rightarrow j$, for any $k \neq i$.

Does the graph below obey the formal conditions?

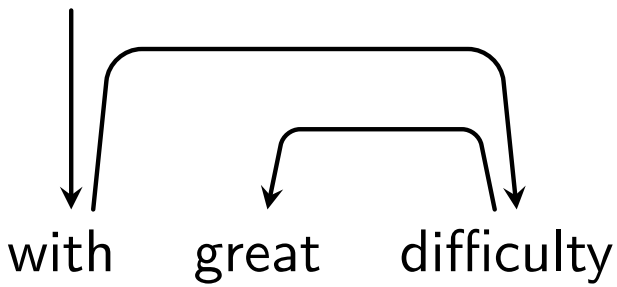


## Projectivity

- A projective graph: If $i \rightarrow j$ then for any $k$ such that $i < k < j$ or $j < k < i$, $i \rightarrow^* k$,

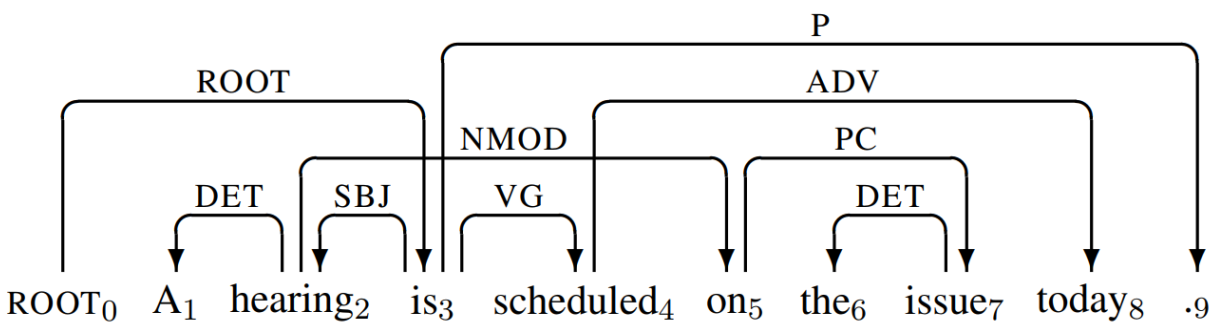
## Projectivity

- A projective graph: If $i \rightarrow j$ then for any $k$ such that $i < k < j$ or $j < k < i$, $i \rightarrow^* k$,



# Projectivity

- Most theoretical frameworks do not assume projectivity.
- Non-projective structures are needed to account for
  - long-distance dependencies,
  - free word order.



# Today's lecture

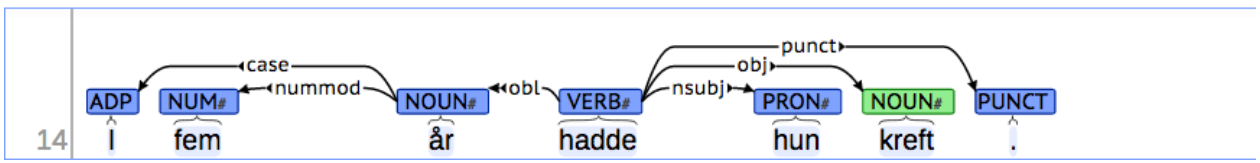
- Very brief repetition of basic principles of syntax:
  - form vs function
  - constituents and phrases
  - context-free grammars
- Dependency Grammar
  - basic concepts: head, dependent
  - comparison to constituent structure
  - formal properties
- Treebanks

# Treebanks

- Collection of sentences manually annotated with syntactic analysis $\Rightarrow$ a **treebank**
- Treebanks are used to train data-driven NLP tools (taggere, parsere)
- Treebanks for a number languages
  - Penn Treebank
  - Prague Dependency Treebank (czech)
  - Negra/Tuba-DZ (German)
  - Penn (Chinese)
  - Norwegian Dependency Treebank
  - Universal Dependencies

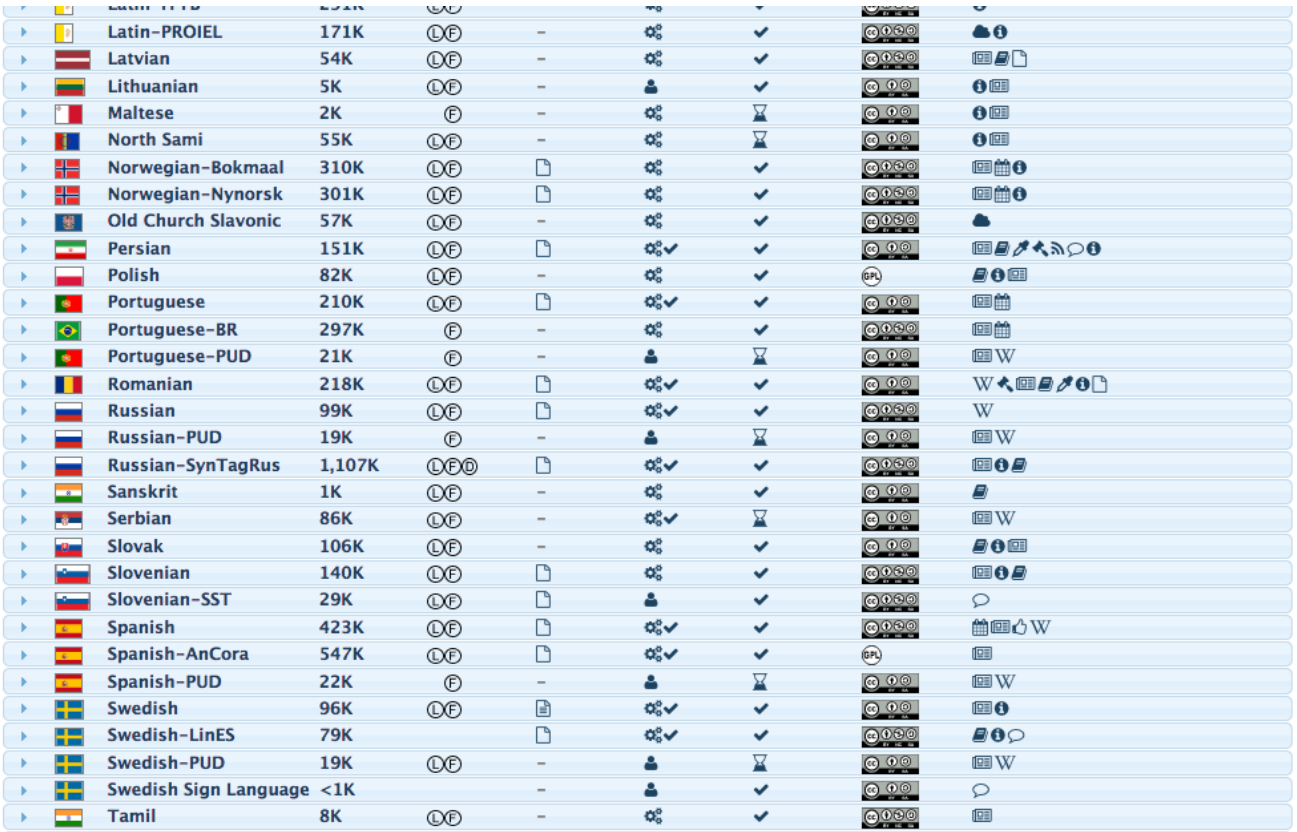
# Norwegian Dependency Treebank (NDT)

- NDT was completed in 2014 (Solberg et al, 2014) by Språkbanken, National Library
- Ca 600,000 tokens of manually annotated Bokmål and Nynorsk text (news, blogs, stortingsmeldinger)
- Enables training of taggers and parsers for Norwegian (Øvrelid & Hohle, 2016; Hohle et al, 2017; Velldal et al, 2017)
- Freely available so others can do the same (and better!)
- Converted to Universal Dependencies (Øvrelid & Hohle, 2016)



# Universal Dependencies

- Harmonized dependency treebanks for more than 100 languages (including Norwegian)
- Norwegian models in spaCy and Stanza
- `http://universaldependencies.org/`

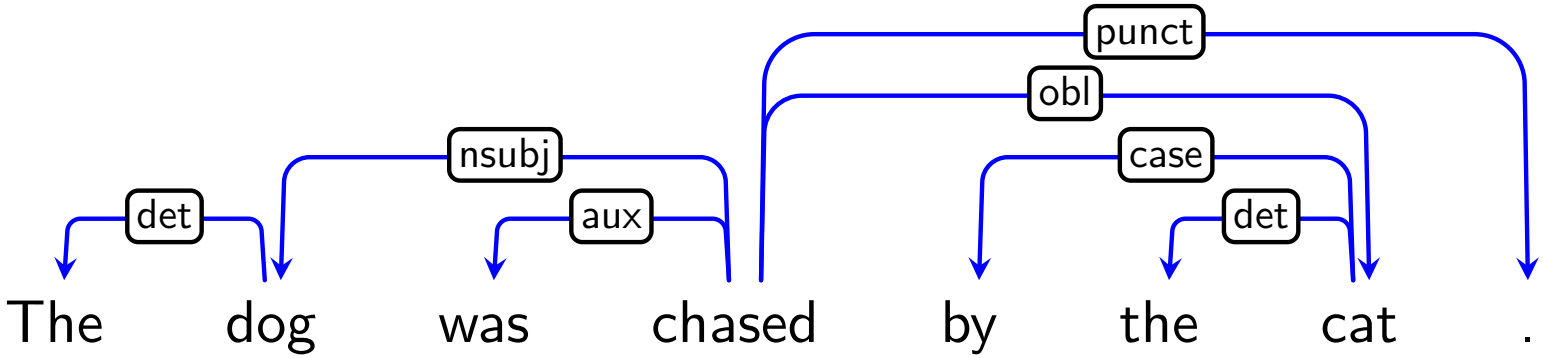


| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | Latin-PROIEL | 171K | LF | – | | | |
| | Latvian | 54K | LF | – | | | |
| | Lithuanian | 5K | LF | – | | | |
| | Maltese | 2K | F | – | | | |
| | North Sami | 55K | LF | – | | | |
| | Norwegian-Bokmaal | 310K | LF | | | | |
| | Norwegian-Nynorsk | 301K | LF | | | | |
| | Old Church Slavonic | 57K | LF | – | | | |
| | Persian | 151K | LF | | | | |
| | Polish | 82K | LF | – | | | |
| | Portuguese | 210K | LF | | | | |
| | Portuguese-BR | 297K | F | – | | | |
| | Portuguese-PUD | 21K | F | – | | | |
| | Romanian | 218K | LF | | | | |
| | Russian | 99K | LF | | | | |
| | Russian-PUD | 19K | F | – | | | |
| | Russian-SynTagRus | 1,107K | LFD | | | | |
| | Sanskrit | 1K | LF | – | | | |
| | Serbian | 86K | LF | – | | | |
| | Slovak | 106K | LF | – | | | |
| | Slovenian | 140K | LF | | | | |
| | Slovenian-SST | 29K | LF | | | | |
| | Spanish | 423K | LF | | | | |
| | Spanish-AnCora | 547K | LF | | | | |
| | Spanish-PUD | 22K | F | – | | | |
| | Swedish | 96K | LF | | | | |
| | Swedish-LinES | 79K | | | | | |
| | Swedish-PUD | 19K | LF | – | | | |
| | Swedish Sign Language | <1K | | – | | | |
| | Tamil | 8K | LF | – | | | |

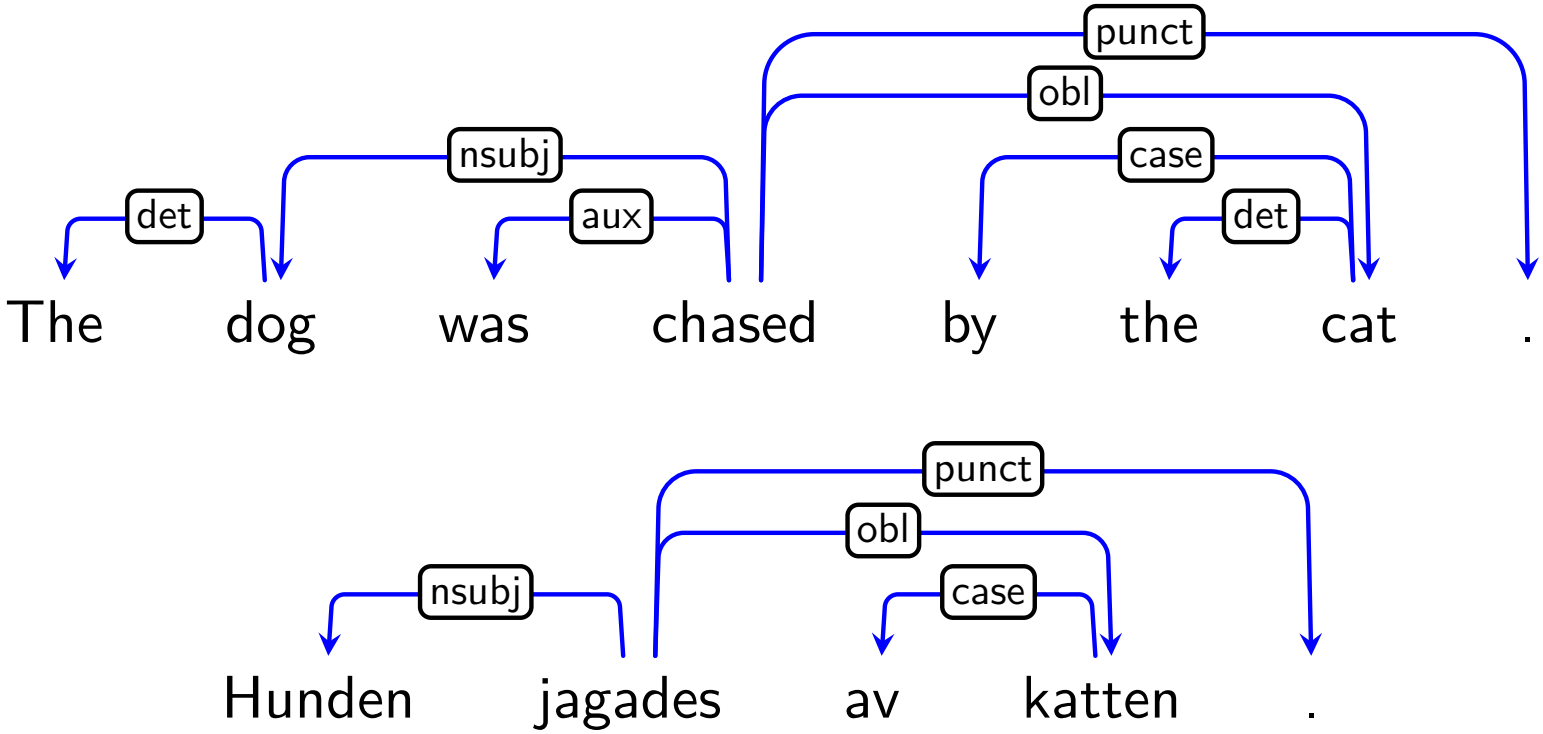
# Example 'Universal' Dependency Types

| | | |
|---|---|---|
| nsubj | nominal subject | She arrived. |
| csubj | clausal subject | That she arrived surprised me. |
| obj | (direct) object | My mother called me. |
| iobj | indirect object | She teaches my daughter maths. |
| ccomp | clausal complement | She knew that she arrived. |
| xcomp | open clausal complement | She promised to sing. |
| obl | oblique nominal | She arrived on Monday |
| obl | oblique nominal | She depends on me. |
| nmod | nominal modifier | the office of the chair is empty. |
| amod | adjectival modifier | the fierce dog barks. |
| acl | adjectival clause | the dog that barks arrived. |
| conj | conjunct | Kim and Sandy arrived. |
| cc | coordinating conjunction | Kim and Sandy arrived. |

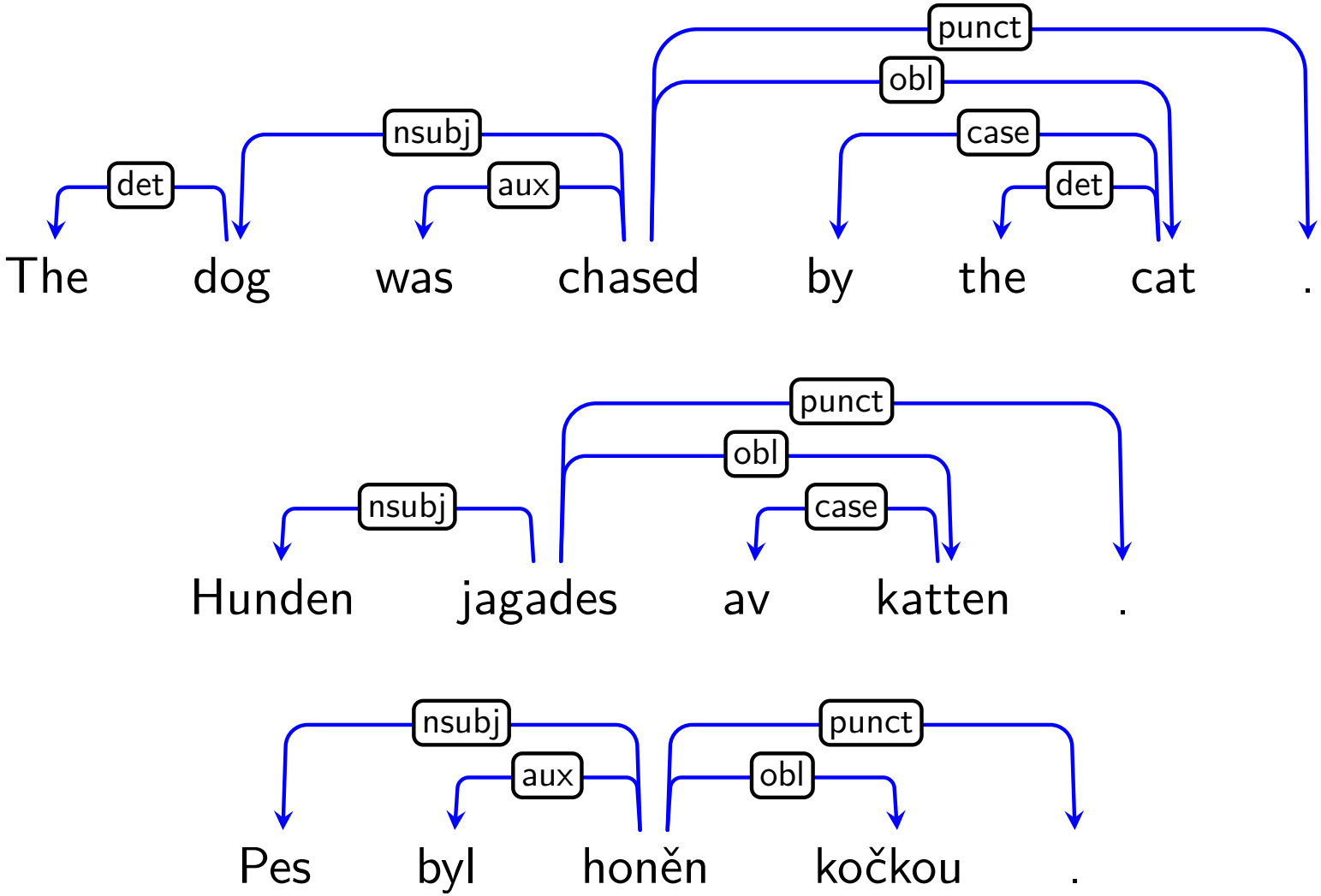
# (Degrees of) Cross-Linguistic Consistency



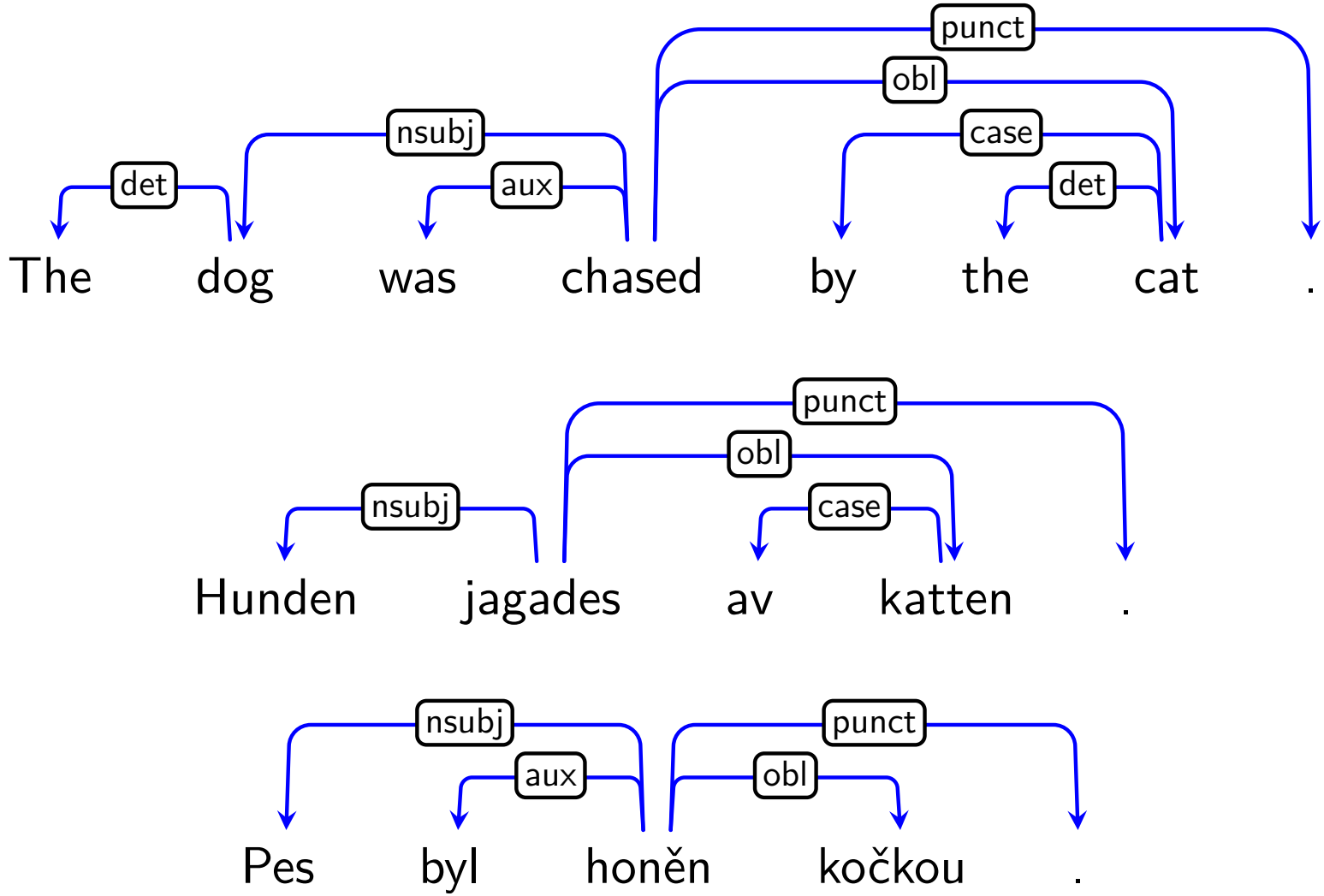
# (Degrees of) Cross-Linguistic Consistency



# (Degrees of) Cross-Linguistic Consistency



# (Degrees of) Cross-Linguistic Consistency



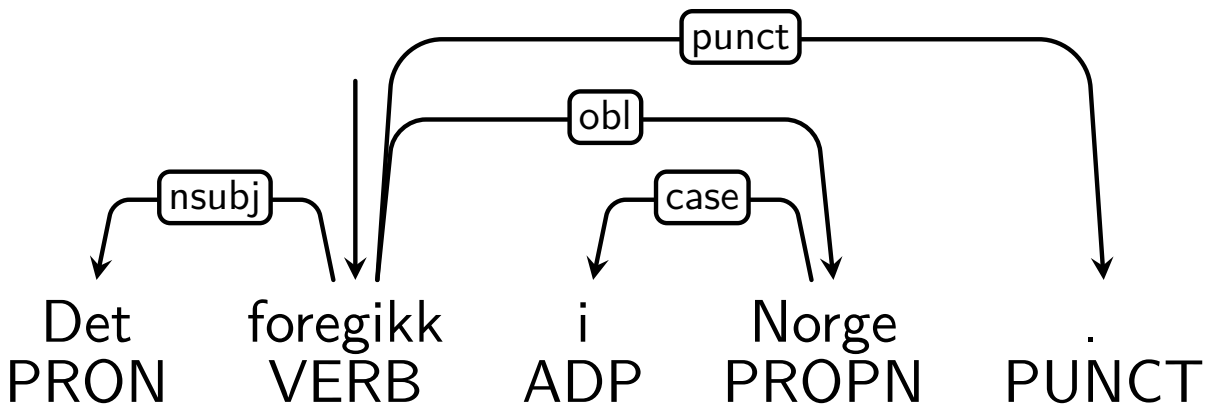

- Capitalize on content words, e.g. demote case-marking prepositions.

# CoNLL-U format

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | Det | det | PRON | Gender=Neut\|... | 2 | nsubj |
| 2 | foregikk | foregå | VERB | Mood=Ind\|... | 0 | root |
| 3 | i | i | ADP | _ | 4 | case |
| 4 | Norge | Norge | PROPN | _ | 2 | obl |
| 5 | . | . | PUNCT | _ | 2 | punct |

# CoNLL-U format

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | Det | det | PRON | Gender=Neut\|... | 2 | nsubj |
| 2 | foregikk | foregå | VERB | Mood=Ind\|... | 0 | root |
| 3 | i | i | ADP | _ | 4 | case |
| 4 | Norge | Norge | PROPN | _ | 2 | obl |
| 5 | . | . | PUNCT | _ | 2 | punct |



# Next week

- Syntactic parsing
- Data-driven parsing
- Data-driven dependency parsing
- Evaluation