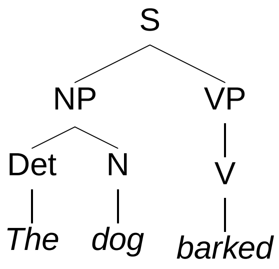


$$\begin{bmatrix} \mathtt{LTOP} & h_1 \\ \mathtt{INDEX} & e_2 \\ \mathtt{RELS} & \left\langle \begin{bmatrix} \textit{prpstn\_m\_rel} & \\ \mathtt{LBL} & h_1 \\ \mathtt{MARG} & h_3 \end{bmatrix} \begin{bmatrix} \textit{def\_q\_rel} & \\ \mathtt{LBL} & h_4 \\ \mathtt{ARG0} & x_5 \\ \mathtt{RSTR} & h_6 \\ \mathtt{BODY} & h_7 \end{bmatrix} \begin{bmatrix} \textit{“dog\_n\_rel”} & \\ \mathtt{LBL} & h_8 \\ \mathtt{ARG0} & x_5 \end{bmatrix} \begin{bmatrix} \textit{“bark\_v\_rel”} & \\ \mathtt{LBL} & h_9 \\ \mathtt{ARG0} & e_2 \\ \mathtt{ARG1} & x_5 \end{bmatrix} \right\rangle \\ \mathtt{HCONS} & \langle h_3 =_q h_9,\, h_6 =_q h_8 \rangle \end{bmatrix}$$

Computational Linguistics (INF2820 — Overview)

The Second Steep Road Against Bergen is a Card

Stephan Oepen

Universitetet i Oslo & CSLI Stanford

`oe@ifi.uio.no`

So, What Exactly is Computational Linguistics?

... *teaching computers our language.* (Alien Researcher, 2000)

So, What Exactly is Computational Linguistics?

... teaching computers our language. (Alien Researcher, 2000)

We Understand$^{\text{TM}}$. Unlike other solutions based on keyword or phrase recognition, YY Software's product actually understands customer e-mails and Web interaction. (Start-Up Marketing Blurb, 2000)

So, What Exactly is Computational Linguistics?

... teaching computers our language. (Alien Researcher, 2000)

We Understand$^{\text{TM}}$. Unlike other solutions based on keyword or phrase recognition, YY Software's product actually understands customer e-mails and Web interaction. (Start-Up Marketing Blurb, 2000)

... the scientific study of human language—specifically of the system of rules and the ways in which they are used in communication—using mathematical models and formal procedures that can be realized and validated using computers; a cross-over of many disciplines. (Stanford Linguistics Professor, 1980s)

So, What Exactly is Computational Linguistics?

... teaching computers our language. (Alien Researcher, 2000)

We Understand™. Unlike other solutions based on keyword or phrase recognition, YY Software's product actually understands customer e-mails and Web interaction. (Start-Up Marketing Blurb, 2000)

... the scientific study of human language—specifically of the system of rules and the ways in which they are used in communication—using mathematical models and formal procedures that can be realized and validated using computers; a cross-over of many disciplines. (Stanford Linguistics Professor, 1980s)

... a sub-discipline of our Artificial Intelligence programme.

(MIT CS Professor, 1970s)

Yes, Great, But Why Should Anyone Care?

In the next three to five years, voice over IP and mobile devices [...] will become prevalent. [...] Desired technologies will soon replace menus and graphic user interfaces with natural-language interfaces. — People so much want to speak English to their computer. (Steve Ballmer, December 2005)

Yes, Great, But Why Should Anyone Care?

In the next three to five years, voice over IP and mobile devices [...] will become prevalent. [...] Desired technologies will soon replace menus and graphic user interfaces with natural-language interfaces. — People so much want to speak English to their computer. (Steve Ballmer, December 2005)

FRAMTIDSFORSKERNES DØDSLISTE [...] Datamaskinen vil mer og mer bli noe vi snakker med. Tastaturet vil nok ikke forsvinne helt, men vi vil definitivt bruke det mindre enn i dag. (Dagsavisen, January 2006)

Yes, Great, But Why Should Anyone Care?

In the next three to five years, voice over IP and mobile devices [...] will become prevalent. [...] Desired technologies will soon replace menus and graphic user interfaces with natural-language interfaces. — People so much want to speak English to their computer. (Steve Ballmer, December 2005)

FRAMTIDSFORSKERNES DØDSLISTE [...] Datamaskinen vil mer og mer bli noe vi snakker med. Tastaturet vil nok ikke forsvinne helt, men vi vil definitivt bruke det mindre enn i dag. (Dagsavisen, January 2006)

Computational Linguistics

→ (young) interdisciplinary science: language, cognition, computation;

→ (once again) commercial growth potential due to ‘information society’.

Some Traditional Applications of CL

Machine Translation

- Traditional: analyse source to some degree, transfer, generate target.

Information Extraction & Text ‘Understanding’

- Email auto- (or assisted-) response: interpret customer requests;
- Semantic Web: annotate WWW with structured, conceptual data.

(Spoken) Dialogue Systems

Grammar & Controlled Language Checking

Summarization & Text Simplification

What Makes Natural Language a Hard Problem?

```
|< |Den andre veien mot Bergen er kort.| --- 16 x 52 x 112 = 112
|> |That other path towards Bergen is short.| [0.70] <0.03> (0:0:0).
|> |That other path against Bergen is short.| [0.70] <0.03> (0:1:0).
|> |That second path towards Bergen is short.| [0.65] <0.03> (2:0:0).
|> |That second path against Bergen is short.| [0.65] <0.03> (2:1:0).
|> |That other road towards Bergen is short.| [0.62] <0.03> (0:2:0).
|> |That other road against Bergen is short.| [0.62] <0.03> (0:3:0).
...
|> |The second path towards Bergen is short.| [0.18] <0.03> (3:0:0).
|> |The second path against Bergen is short.| [0.18] <0.03> (3:1:0).
|> |That second path towards Bergen is a card.| [0.17] <0.02> (8:0:0).
|> |That second path against Bergen is a card.| [0.17] <0.02> (8:1:0).
|> |That other path towards Bergen is cards.| [0.17] <0.03> (5:0:0).
|> |That other path against Bergen is cards.| [0.17] <0.03> (5:1:0).
...
|> |Short is that second road, towards Bergen.| [-0.42] <0.03> (2:2:2).
|> |Short is that other road, against Bergen.| [-0.37] <0.03> (0:3:2).
```

A Tool Towards Understanding: (Formal) Grammar

Wellformedness

- *Kim was happy because ____ passed the exam.*
- *Kim was happy because ____ final grade was an A.*
- *Kim was happy when she saw ____ on television.*

A Tool Towards Understanding: (Formal) Grammar

Wellformedness

- *Kim was happy because ____ passed the exam.*
- *Kim was happy because ____ final grade was an A.*
- *Kim was happy when she saw ____ on television.*

Meaning

- *Kim gave Sandy the book.*
- *Kim gave the book to Sandy.*
- *Sandy was given the book by Kim.*

A Tool Towards Understanding: (Formal) Grammar

Wellformedness

- *Kim was happy because ____ passed the exam.*
- *Kim was happy because ____ final grade was an A.*
- *Kim was happy when she saw ____ on television.*

Meaning

- *Kim gave Sandy the book.*
- *Kim gave the book to Sandy.*
- *Sandy was given the book by Kim.*

Ambiguity

- *Kim saw the astronomer with the telescope.*
- *Have her report on my desk by Friday!*

A Grossly Simplified Example

The Grammar of Spanish

S → NP VP

VP → V NP

VP → VP PP

PP → P NP

NP → “nieve”

NP → “Juan”

NP → “Oslo”

V → “amó”

P → “en”

Juan amó nieve en Oslo

A Grossly Simplified Example

The Grammar of Spanish

S → NP VP

VP → V NP

VP → VP PP

PP → P NP

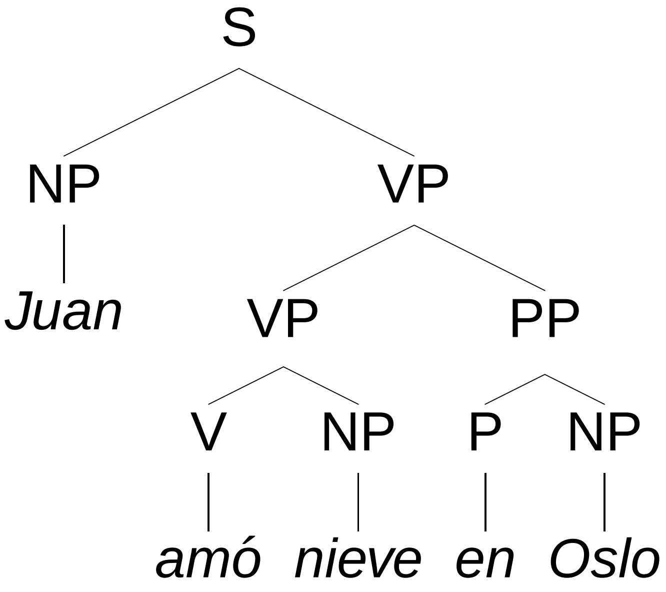
NP → “nieve”

NP → “Juan”

NP → “Oslo”

V → “amó”

P → “en”

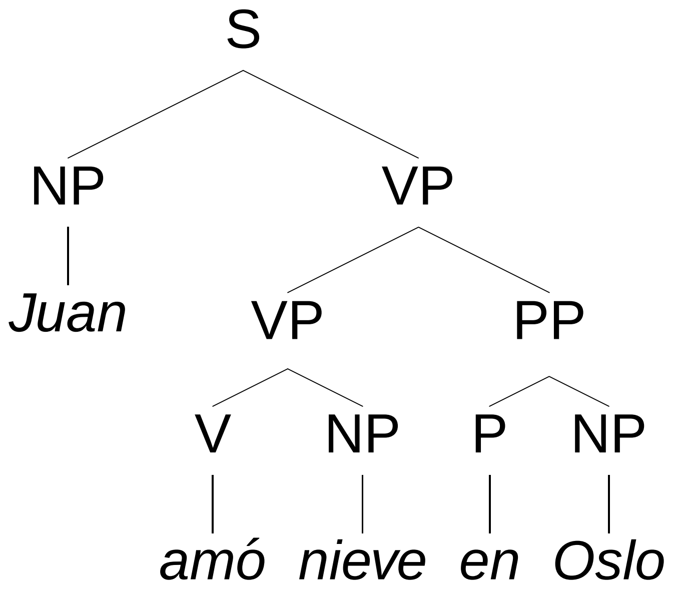


Juan amó nieve en Oslo

A Grossly Simplified Example

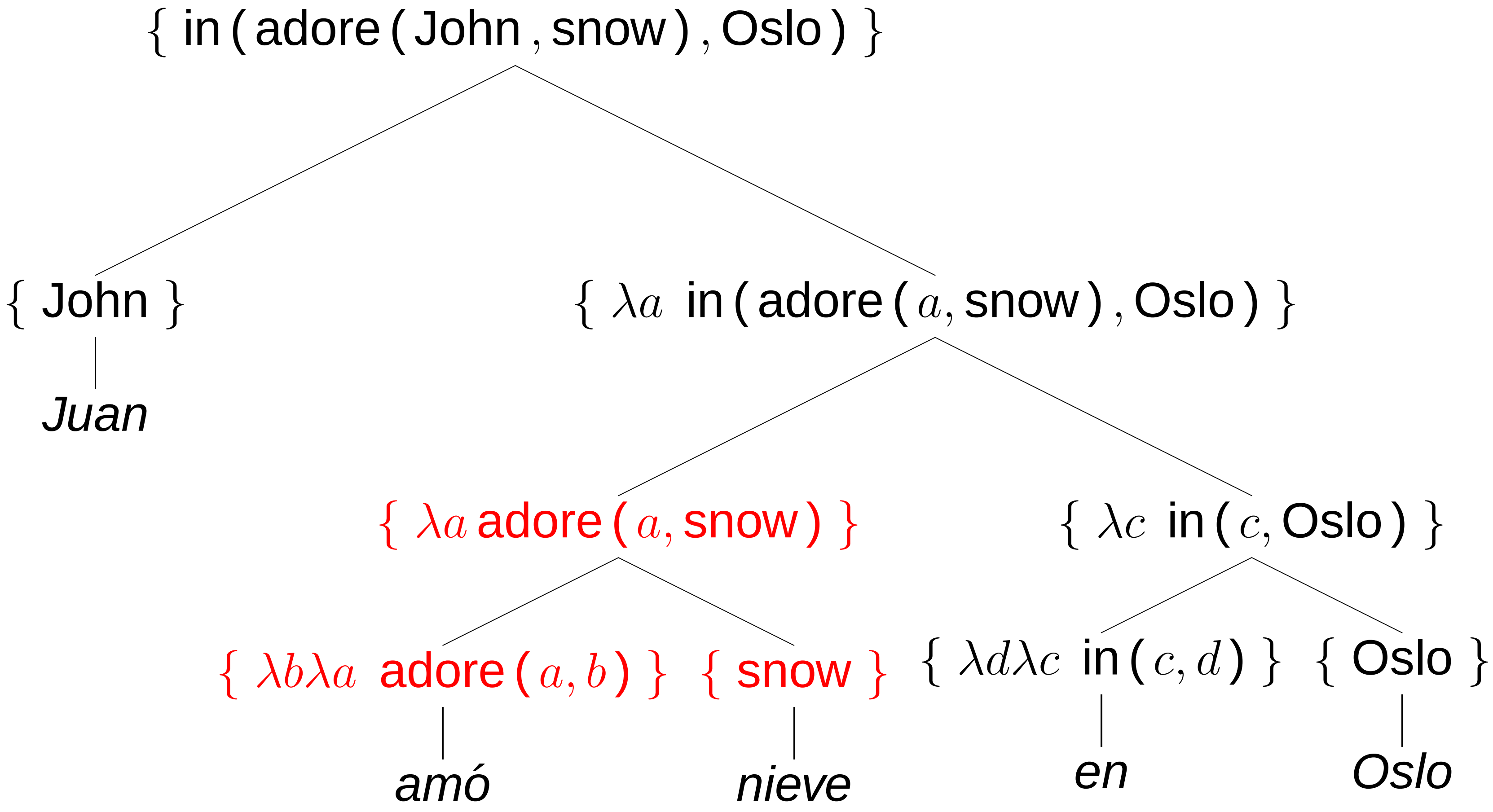
The Grammar of Spanish

S → NP VP	{ VP (NP) }
VP → V NP	{ V (NP) }
VP → VP PP	{ PP (VP) }
PP → P NP	{ P (NP) }
NP → “nieve”	{ snow }
NP → “Juan”	{ John }
NP → “Oslo”	{ Oslo }
V → “amó”	{ $\lambda b \lambda a$ adore (a, b) }
P → “en”	{ $\lambda d \lambda c$ in (c, d) }



Juan amó nieve en Oslo

Meaning Composition (Grossly Simplified, Still)



{ in (adore (John , snow) , Oslo) }

{ John }

Juan

{ λa in (adore ($a,$ snow) , Oslo) }

{ λa adore ($a,$ snow) }

{ λc in ($c,$ Oslo) }

{ $\lambda b \lambda a$ adore (a, b) }

{ snow }

{ $\lambda d \lambda c$ in (c, d) }

{ Oslo }

amó

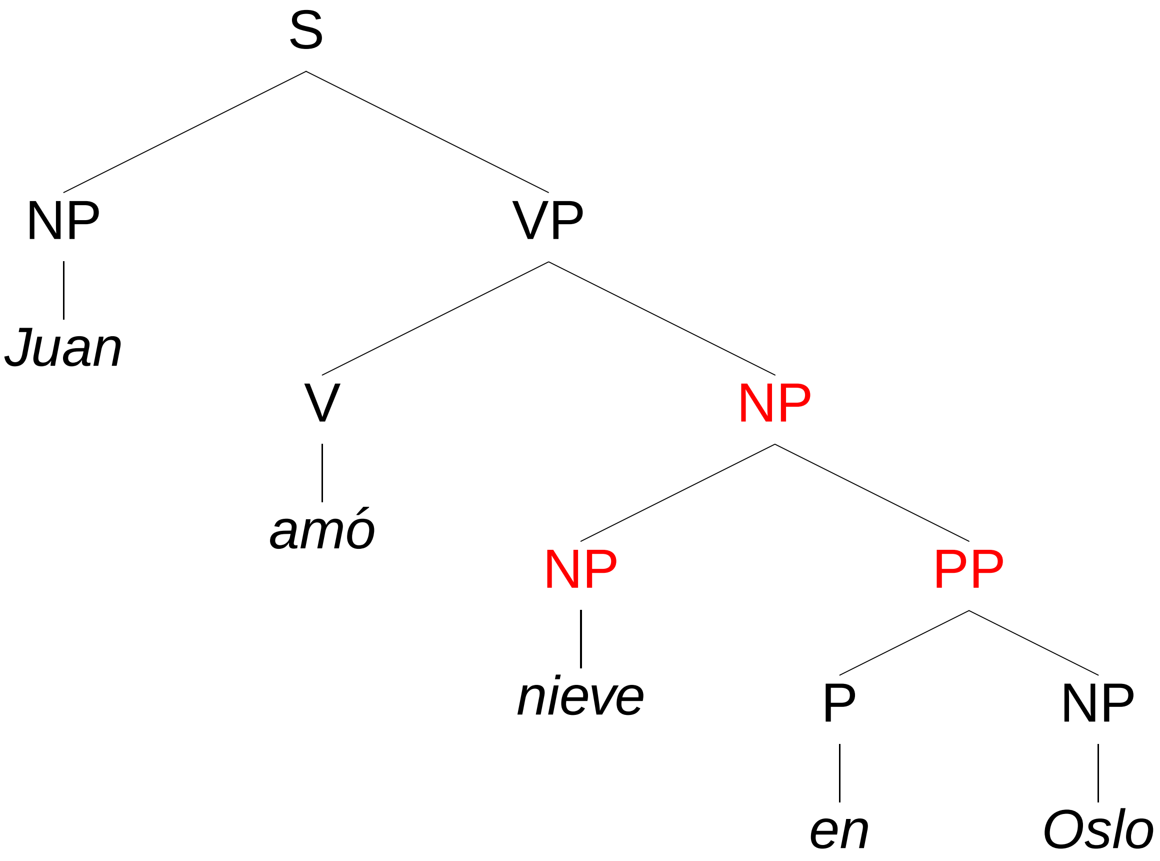
nieve

en

Oslo

VP $\rightarrow$ V NP { V (NP) }

Another Interpretation — Structural Ambiguity



NP → NP PP { PP (NP) }

An Outlook — Context-Free Grammars

- Formally, a *context-free grammar* (CFG) is a quadruple: $\langle C, \Sigma, P, S \rangle$
- C is the set of categories (aka *non-terminals*), e.g. $\{$S, NP, VP, V$\}$;
- Σ is the vocabulary (aka *terminals*), e.g. $\{$Juan, nieve, amó, en$\}$;
- P is a set of category rewrite rules (aka *productions*), e.g.

S $\rightarrow$ NP VP

VP $\rightarrow$ V NP

NP $\rightarrow$ Juan

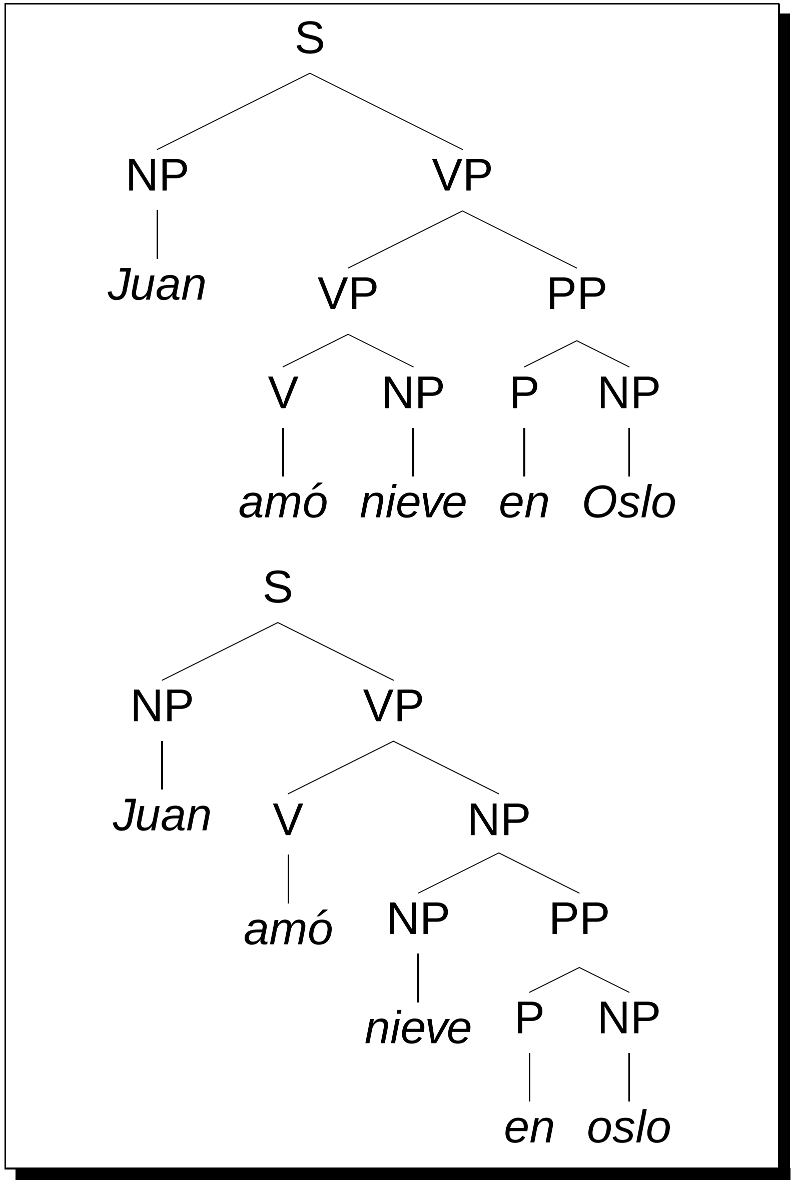
NP $\rightarrow$ nieve

V $\rightarrow$ amó

- $S \in C$ is the *start symbol*, a filter on complete (‘sentential’) results;
- for each rule ‘$\alpha \rightarrow \beta_1, \beta_2, ..., \beta_n$’ $\in P$: $\alpha \in C$ and $\beta_i \in C \cup \Sigma$; $1 \leq i \leq n$.

Parsing: Recognizing the Language of a Grammar

S → NP VP
VP → V NP
VP → VP PP
NP → NP PP
PP → P NP
NP → Juan | nieve | Oslo
V → amó
P → en



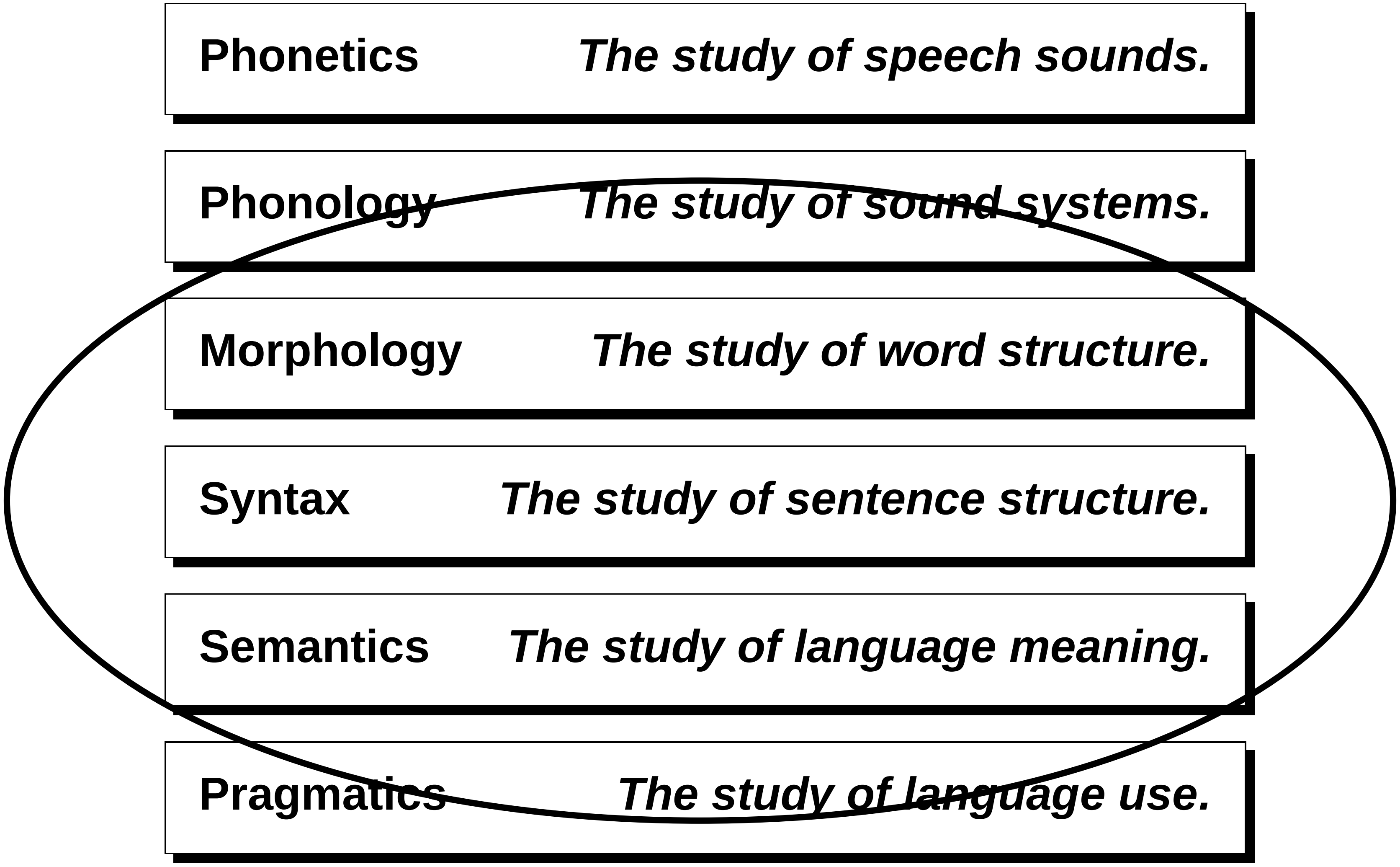
All Complete Derivations

- are rooted in the start symbol S;
- label internal nodes with categories $\in C$, leafs with words $\in \Sigma$;
- instantiate a grammar rule $\in P$ at each local subtree of depth one.

Some Areas of Descriptive Grammar

Phonetics	***The study of speech sounds.***
Phonology	***The study of sound systems.***
Morphology	***The study of word structure.***
Syntax	***The study of sentence structure.***
Semantics	***The study of language meaning.***
Pragmatics	***The study of language use.***

Some Areas of Descriptive Grammar



Phonetics	***The study of speech sounds.***
Phonology	***The study of sound systems.***
Morphology	***The study of word structure.***
Syntax	***The study of sentence structure.***
Semantics	***The study of language meaning.***
Pragmatics	***The study of language use.***

More, and More, and More Ambiguity

Speech Recognition

its		*hard*	*to*	*wreck*	*a*	*nice*	*beach*
it	*’s*	*hard*	*to*	*recognize*			*speech*

Morphology

- *fisker* *fisk* $_{\text{N}}$ $+$ plural vs. *fiske* $_{\text{V}}$ $+$ present vs. *fisker* $_{\text{N}}$ $+$ singular;
- *brus-automat* vs. *bru-sau-tomat*; *vinduene* vs. *vin-duene*; et al.

Semantics

- *All Norwegians speak two languages.* $\exists l_1, l_2 \forall n \ldots$ vs. $\forall n \exists l_1, l_2 \ldots$

Limitations of Context-Free Grammar

Agreement and Valency (For Example)

That dog barks.

*That dogs barks.

*Those dogs barks.

The dog chased a cat.

*The dog barked a cat.

*The dog chased.

*The dog chased a cat my neighbours.

The cat was chased by a dog.

*The cat was chased of a dog.

...

Third Generation of Natural Language Grammars

(Constraint- or) Unification-Based Grammars

- Structured categories: (nested) feature structures and unification;
- typing: multi-dimensional, hierarchical encoding of knowledge (OO);
- declarativity and reversibility: support both parsing and generation;
- focus on engineering methodologies and processing efficiency;
- cross-discipline, cross-language fertilization: linguistics meets CS.

Some Acronyms

- LFG, GPSG, HPSG now widely accepted grammatical frameworks;
- broad-coverage resource grammars (20^+ person years) available:

`http://lingo.stanford.edu/erg`

(Unification-Based) HPSG Parsing — Then and ‘Now’

Version	Platform	Test Set	filter %	etasks ϕ	pedges ϕ	tcpu ϕ (s)	space ϕ (kb)
October 1996	PAGE	*‘tsnlp’*	49·9	656	44	4·77	19,016
		‘aged’	51·3	1763	**97**	**36·69**	79,093
August 2000	PET	*‘tsnlp’*	93·9	170	55	0·03	333
		‘aged’	95·1	753	**292**	**0·14**	1,435
		‘fuse’	95·5	3084	1140	0·65	10,589

(generated by [incr tsdb()] at 5-nov-2000 (21:23 h)

Cumulative Break-Through in Parsing Efficiency

- Oldest comparable profiles: net speed-up of around 260 (excluding gc);
- grammar evolution: problem size (in edges) increased by factor of three;
- additional factors (hardware, packing): above four orders of magnitude;

→ Unification-based parsing nowadays applied at ‘Web scale’ (PowerSet).

The Rationalist vs. Empiricist Stand-Off

> *Every time I fire a linguist,*
> *system performance goes up.*
>
> [Fred Jelinek, 1980s]

The Rationalist vs. Empiricist Stand-Off

Every time I fire a linguist,
system performance goes up.

[Fred Jelinek, 1980s]

Competition of Paradigms

- Rationalist: formally encode linguistic and extra-linguistic knowledge;
- empiricist: statistical models trained on distributional data (corpora);
- Jelinek eventually turned off the lights — grammar research stable;

→ hybrids: combination of approaches required for long-term success.

Competing Approaches (1 of 2)

Can you send me copies of all checks in December?

Statistical Part-of-Speech Tagging (96.7 % Accuracy)

1.0 MD	1.0 PRP	0.98 VB	1.0 PRP	1.0 NNS	1.0 IN	1.0 DT	0.93 NNS	1.0 IN	1.0 NNP	1.0 .
Can	you	send	me	copies	of	all	checks	in	December	?

Text Classification (∼ 85 % Accuracy)

CheckCopyRequest 0.6934, *CheckBookRequest* 0.0247,
StatementCopyRequest 0.0066, ...

Competing Approaches (2 of 2)

$$\begin{array}{l}
\langle\, h_1, \\
\quad \{\, h_1\text{:int\_m}(h_2),\ h_3\text{:\_can\_v\_modal}(e_4,\ h_5),\ h_7\text{:\_send\_v}(e_8,\ x_9,\ x_{10},\ x_{11}), \\
\qquad h_{12}\text{:pronoun\_q}(x_9,\ h_{13},\ h_{14}),\ h_{15}\text{:pron}(x_9\,\{\,\text{2nd}\,\}), \\
\qquad h_{16}\text{:pronoun\_q}(x_{10},\ h_{17},\ h_{18}),\ h_{19}\text{:pron}(x_{10}\,\{\,\text{1sg}\,\}), \\
\qquad h_{20}\text{:bare\_q}(x_{11},\ h_{21},\ h_{22}),\ h_{23}\text{:\_copy\_n\_of}(x_{11}\,\{\,\text{pl}\,\},\ x_{24}), \\
\qquad h_{25}\text{:\_all\_q}(x_{24},\ h_{26},\ h_{27}),\ h_{28}\text{:\_check\_n}(x_{24}\,\{\,\text{pl}\,\}), \\
\qquad h_{28}\text{:temp\_loc}(\_,\ x_{24},\ x_{29}),\ h_{30}\text{:proper\_q}(x_{29},\ h_{31},\ h_{32}),\ h_{33}\text{:mofy}(x_{29},\ \text{``DEC''})\,\}, \\
\quad \{\, h_2 =_q h_3,\ h_5 =_q h_7,\ h_{13} =_q h_{15},\ h_{17} =_q h_{19},\ h_{21} =_q h_{23},\ h_{26} =_q h_{28},\ h_{31} =_q h_{33}\,\}\,\rangle
\end{array}$$

(Truth-Conditional or) Logical-Form Semantics

+ high-level abstraction; grounded in entities and relations → inference;

– very difficult to construct (correctly, with broad-coverage) and process.

Putting Things Together: Language and the World

Discourse and Pragmatics

- h_{15}:pron(x_9 { 2nd }) → email recipient; h_{19}:pron(x_{10} { 1sg }) → email sender;
- h_{28}:temp_loc(_, x_{24}, x_{29}), h_{33}:mofy(x_{29}, “DEC”) → 2005 (but maybe 2006);
- h_1:int_m(h_2), h_3:_can_v_modal(e_4, h_5), h_7:_send_v(e_8, x_9, x_{10}, x_{11}) → request.

World Knowledge (Plus Back-End Databases)

- ‘all checks in December 2005’
  $\rightarrow \{\, x \mid x \; isa$ check $\wedge$ 20051201 $\leq$ date(x) $\leq$ 20051231 $\}$
- request h_7:_send_v(e_8, x_9, x_{10}, x_{11}), h_{23}:_copy_n_of(x_{11}, x_{24}), h_{28}:_check_n(x_{24})
  → <CheckCopyRequest from="26046712345" ...> ... </>

Summary — Computational Linguistics Today

Some Lessons Learned

- Surprisingly hard problem: many unknowns in human language capacity;
- statistical NLP can deliver robust, practical systems → limited scalability;
- knowledge-based systems demand long-term development → re-usability;
- limited-domain applications possible (e.g. BUSSTUC); too few end-to-end;

→ empiricist vs. rationalist stand-off now largely reconciled: cross-fertilization.

Background Reading

- general: `http://www.coli.uni-saarland.de/~hansu/what_is_cl.html`;
- Jurafsky, Daniel and Martin, James H.: *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Upper Saddle River, NJ (2000).

INF2880 — What We Are About to Do (and Why)

Course Outline

- Extend understanding of (natural) language as a system of rules;
- learn how to *formalize* grammars through typed feature structures;
- design and implement common algorithms and probabilistic models;
- solve regular exercises: immediate gratification (risk of late hours).

Three Interacting Components

- **grammar engineering** formalize linguistic theories with complex interactions of multiple phenomena; implementation and debugging;
- **processing** understand common parsing algorithms; unification of feature structures; implement an efficient unification-based parser;
- **probabilistic models** capture relative frequency of (competing) phenomena; approximate graded grammaticality or soft constraints.

Grammar Engineering from a CS Perspective

Implementation Goals

- Translate linguistic constraints into specific formalism → formal model;
- computational grammar provides mapping between form and meaning;
- assign correct analyses to grammatical, reject ungrammatical inputs;
- parsing and generation algorithms: apply mapping in either direction.

Analogy to (Object-Oriented) Programming

- Computational system with observable behavior: immediately testable;
- typed feature structures as a specialized (OO) programming language;
- make sure that all the pieces fit together; revise – test – revise – test ...

The Linguistic Knowledge Builder (LKB)

General & History

- Specialized grammar engineering environment for TFS grammars;
- main developers: Copestake (original), Carroll, Malouf, and Oepen;
- open-source and binary distributions (Linux, Windows, and Solaris).

Grammar Engineering Functionality

- Compiler for typed feature structure grammars → wellformedness;
- parser and generator: map from strings to meaning and vice versa;
- visualization: inspect trees, feature structures, intermediate results;
- debugging and tracing: interactive unification, ‘stepping’, et al.

Why Common-Lisp for Implementation Exercises?

- Arguably most widely used language for ‘symbolic’ computation;
- easy to learn: extremely simple syntax; straightforward semantics;
- a rich language: multitude of built-in data types and operations;
- full standardization; Common-Lisp has been stable for a decade;
- LKB (experimentation environment) implemented in Common-Lisp;

→ for our purposes, (at least) as good a choice as any other language.

$$n! \quad \equiv \quad \begin{cases} 1 & \text{for } n = 0 \\ n \times (n-1)! & \text{for } n > 0 \end{cases}$$

```
(defun ! (n)
  (if (= n 0)
    1
    (* n (! (- n 1)))))
```

Course Organization

Comments on Background Literature

Formal Grammar and General NLP

- Sag, Ivan A. Tom Wasow, and Emily M. Bender: *Syntactic Theory. A Formal Introduction (2nd Edition).* Stanford, CA: CSLI Publications (2003);
- Jurafsky, Daniel and Martin, James H.: *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (2nd Edition).* Upper Saddle River, NJ: Prentice Hall (2008).

The Linguistic Knowledge Builder

- Copestake, Ann: *Implementing Typed Feature Structure Grammars.* Stanford, CA: CSLI Publications (2001).