Formal Grammars

Chapter 12 J&M'09

Grammars

Recall that finite-state methods assume Natural Language sentences only involve local dependencies.

This assumption is false, and creates trivial errors:

- Machine translation

  (1) This is the company that the bank bought.

- Speech syntesis

  (2) Which record broke?

- Search engines (and question answering)

  (3) Who would have been vice-president if Romney had won the election?

Grammars

[What **would** happen **if** Obama and **Romney** tied? - US **elections** ...]
www.independent.co.uk › News › World › US Elections
Nov 4, 2012 – On 21 January, Mitt **Romney** takes the presidential oath **of** office. ... On **election** eve, Obama **has** a slim lead in polls ... and the Senate, likely to remain controlled by Democrats, **would** select the **vice-president**. "**If** ... Just as past presidential **election** disputes **have been** resolved with an act **of** statesmanship, ...

[What went wrong with my prediction about Mitt **Romney** and the ...]
www.foxnews.com/.../what-went-wrong-with-my-prediction-about-...
Nov 13, 2012 – I predicted that Mitt **Romney would** be our next president. ... This **election** should **have been**, 100 out **of** 100 times, a repeat **of** Reagan's landslide ... The GOP **has** to reach out to women and Latinos. ... That argument **would** have **won** the presidency. ... He is a former Libertarian **vice presidential** nominee.

[Adviser: **Romney** "shellshocked" by loss - CBS News]
www.cbsnews.com/8301-250.../adviser-**romney**-shellshocked-by-loss...
Nov 8, 2012 – **Romney's** pre-**election** optimism was the result **of** a few key miscalculations. ... They just couldn't believe they **had been** so wrong. ... They didn't want to **have** to withdraw their concession, like Al Gore did in 2000, and they .... (I, also could relate to **Vice President**, Joe Biden's behavior, in debate with Ryan.

[Live Coverage and Results - **Election** 2012 - NYTimes.com]
elections.nytimes.com/2012/results/live-coverage
Nov 7, 2012 – From left, Michelle Obama, President Obama, **Vice President** Joseph R. Biden and ... The candidates **have** given their speeches and gone home, and we're heading .... President Obama narrowly **won** Virginia over Mitt **Romney**, giving him at .... CHICAGO — **When** NBC projected President Obama **had been** ...

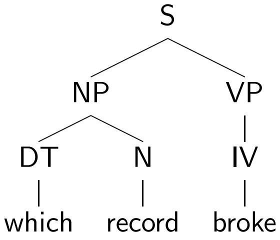
Grammars

Recall the **Chomsky-Schützenberger hierarchy**

Language	**Grammar**	**Automaton**
Regular	A $\rightarrow$ a, A $\rightarrow$ aB	Finite state machine
Context-free	A $\rightarrow \gamma$	Non-deterministic pushdown automaton
Context-sensitive	$\gamma \mathsf{A} \beta \rightarrow \alpha\gamma\beta$	Linear-bounded non-deterministic Turing machine
Recursively enumerable	$\alpha \rightarrow \beta$	Turing machine

Grammars

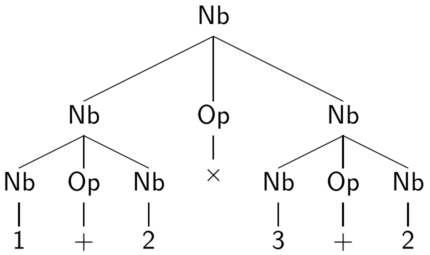
If we construct a more complete representation of the sentence, then the word tags will be more consistent with the overall structure:

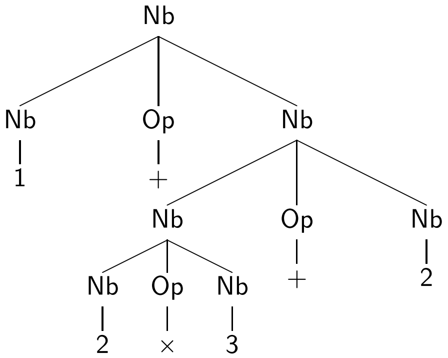


Grammars

ANALOGY:
imagine modeling the language of arithmetic with n-grams or FSA:

$(1{+}2) \times (3{+}2) = 15$ $\qquad\qquad$ $1 + ((2 \times 3) + 2) = 9$





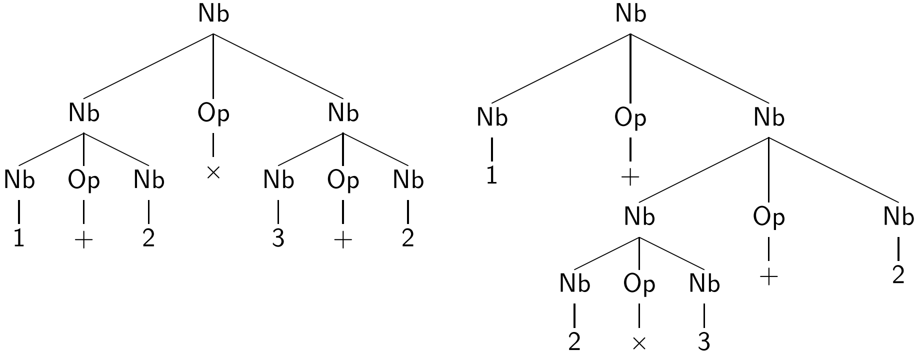
Grammars

Lexicon:

Nb → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Op → + | – | × | =

Syntax: Nb → Nb Op Nb



Grammars

NLs have a bigger lexicon, more rules, and no visible brackets.

(4) a. [French [history teacher]]
    b. [[French history] teacher]

(5) a. [The reporter [said [that the elephant died yesterday]]].
    b. [The reporter [said [that the elephant died] yesterday]].

(6) a. The room contained [[noisy children] and animals].
    b. The room contained [noisy [children and animals]].

(7) a. Put the block in the box.
    b. Put the block in the box on the table.
    c. Put the block in the box on the table in the hallway.
    d. Put the block in the box on the table in the hallway outside the bedroom.
    e. Put the block in the box on the table in the hallway outside the bedroom by the pool.

Grammars

In many languages, word order is relatively free.
Again, Russian exemplifies:

(8) a. Anna uvidela Natashu
‘Anna saw Natasha’
b. Anna Natashu uvidela.
c. Natashu Anna uvidela.
d. Natashu uvidela Anna.
e. Uvidela Natashu Anna.
f. Uvidela Anna Natashu.

(9) a. Natasha uvidela Annu
‘Natasha saw Anna’
b. Annu Natasha uvidela.
c. Natasha Annu uvidela.
d. Natasha uvidela Annu.
e. Uvidela Natasha Annu.
f. Uvidela Annu Natasha.

Subject, Verb and Object order in a sample of 1228 languages:

- 497 are SOV (e.g. like Japanese – Japan)
- 435 are SVO (e.g. like Mandarin – China)
- 85 are VSO (e.g. like Irish – Ireland)
- 26 are VOS (e.g. like Nias – Indonesia)
- 9 are OVS (e.g. like Hixkaryana – Brazil)
- 4 are OSV (e.g. like Jaupés-Japurá – Brazil)
- 172 are mixed (e.g. Syrian Arabic; both SVO/VSO)

Haspelmath et al. 2005 The World Atlas of Language Structures, Oxford Press

Word order map of the world's languages

Grammars

There are many grammar-based systems (some with online demos):

Lisp:

- ERG
- KRG
- Heart of Gold
- Delph-In

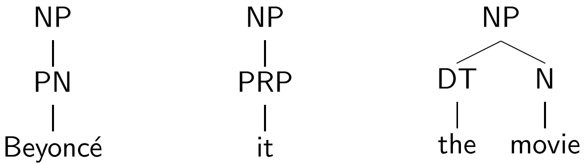
Java:

- Stanford parser
- CCG

Easy access to lots of grammars and parsers in Java via:

- StanfordNLP's parser
- OpenNLP's parser
- OpenCCG's parser
- NLTK's parsers

Three examples of basic types of NP:



Minigrammar

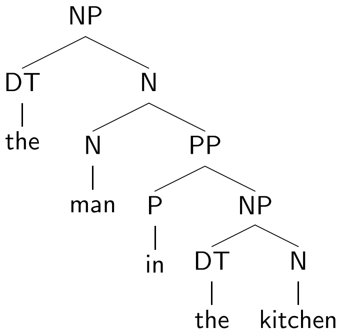
PRP → *we* | *she* | *he* | *her* | ...
DT → *the* | *each* | *a* | ...
N → *movie* | *person* | *cat* | ...
PN → *Tom* | *Beyoncé* | ...

NP → PRP
NP → PN
NP → DT N

Grammars

More rules

N → N PP
PP → P NP
P → *with* | *on* | *in* | ...
N → *man* | *cat* | *rock* | ...



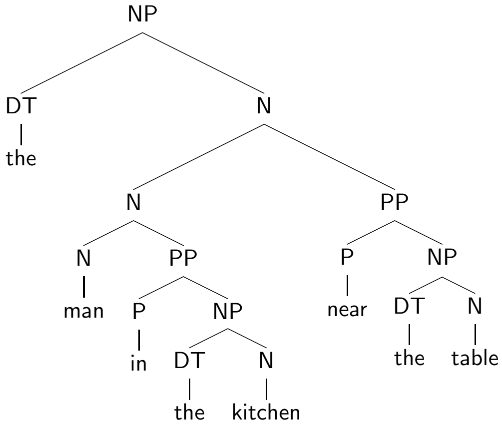
(10) a. I know the man in the kitchen.

b. I know the man in the kitchen, near the table.

c. I know the man in the kitchen, near the table, in the blue shirt.

d. I know the man in the kitchen, near the table, in the blue shirt, with the mustache.

This 'N → N PP' rule can apply to its own output:



In this case, each PP is imposing constraints on the same noun:
... 'the man who is in the kitchen, who is near the table'

Grammars

But the added PPs need not be about the same N:

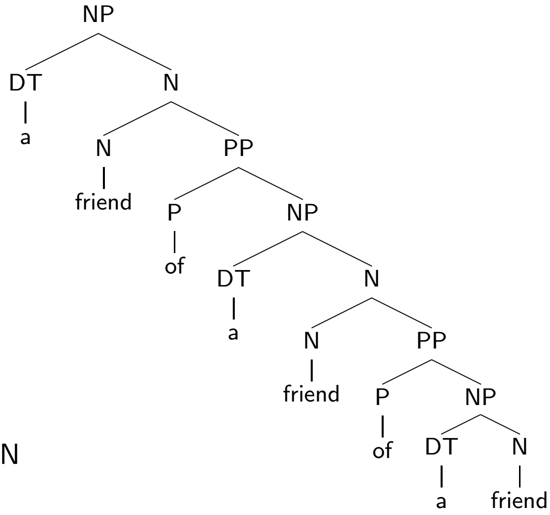
(11) a. The winner [of the race]
b. The winner [of the race [on TV]]
c. The winner [of the race [on TV [in the restaurant]]]
d. The winner [of the race [on TV [in the bar [near my house]]]]
e. ....

(12) I met a friend [of a friend [of a friend [of a friend [of Sue]]]].

This is already predicted by our rules

NP $\rightarrow$ DT N
N $\rightarrow$ N PP
PP $\rightarrow$ P NP

Grammars



NP → DT N
N → N PP
PP → P NP

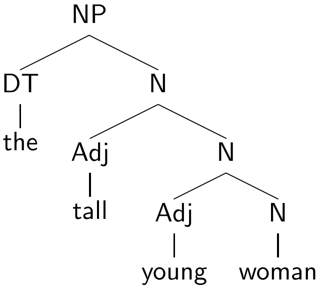
NPs optionally contain adjectives

(13) a. The tall woman
b. The tall young woman
c. The tall young obnoxious woman
d. The tall young obnoxious Italian woman
e. ...

Two new rules:

N → Adj N

Adj → *tall* | *short* | *green* | ...



Grammars

Verb classes

- **Intransitive verbs** (no complements)
  VP → IV
  IV → *died* | *smiled* | *fainted* | *blushed* | ...

  (14) a. Tom fainted.
  b.*Tom fainted it

- **Transitive verbs** (one direct object complement)
  VP → TV NP
  TV: → *devour* | *print* | *hit* | *drive* | ...

  (15) a. Tom devoured the burger.
  b.*Tom devoured.

- **Ditransitive verbs** (two complements)
  VP → DTV NP NP
  DTV → *gave* | *sent* | *offer* | *supply* | ...

  (16) a. Tom gave Sue the burger.
  b.*Tom gave Sue.

Verbs combine with whatever complements they need in order to form **verb phrases** (VPs):

$$\text{VP} = \left\{ \begin{array}{l} \text{laughed} \\ \text{opened the door.} \\ \text{tossed him a book.} \\ \ldots \end{array} \right\}$$

VPs are mobile constituents:

(17) a. I wanted to sleep, and I did [sleep].
b. I wanted to sleep, and [sleep] I did.
c. I wanted to read a book, and I WILL [read a book].
d. I wanted to read a book, and [read a book] I WILL.
e.*I wanted to read a book, and [read] I WILL a book.

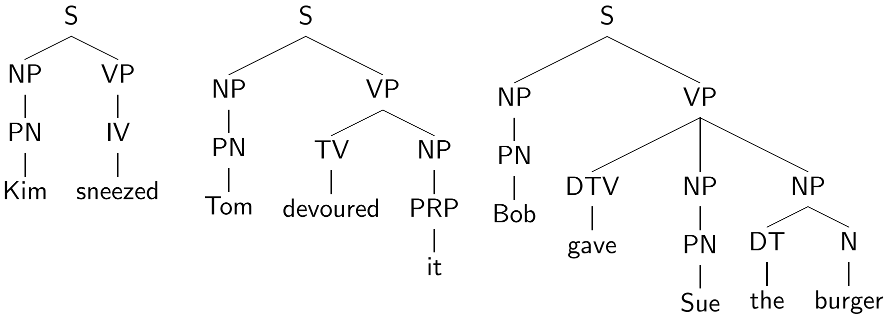
VPs can conjoin with other VPs to create another VP:

(18) You can't [[drive a car] and [drink whiskey]].

Adding a NP to the left of a VP creates a sentence:

S → NP VP

For example:

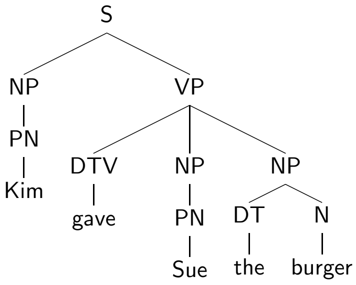


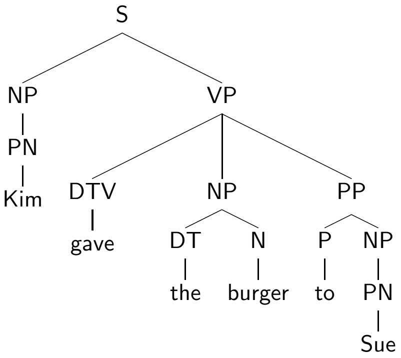
Ditransitive verbs can be used as Prepositional Dative verbs:

(19) a. Kim gave Sue the burger.
     b. Kim gave the burger to Sue.

We can capture this fact with an extra rule:

VP → DTV NP PP

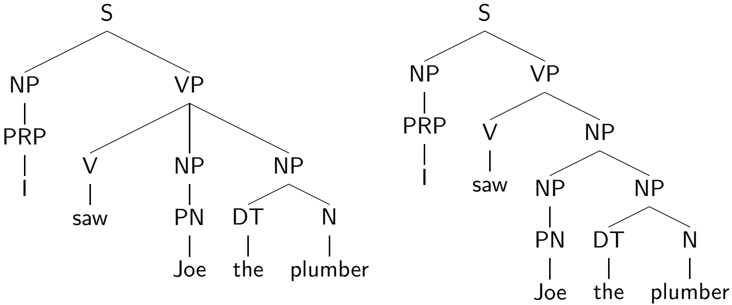




Fine-grained distinctions like TV / DTV are useful

- they reduce incorrect parses;

(20) I saw Joe the plumber.



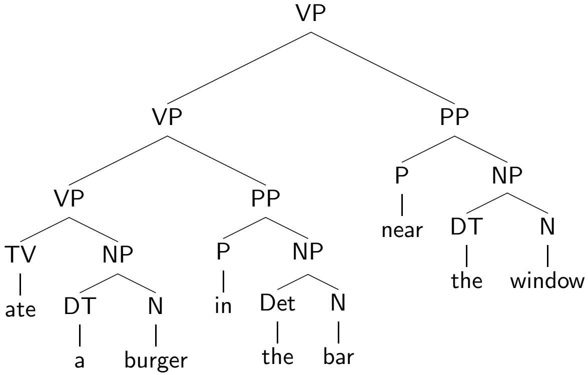
- they are crucial for sentence generation:

(21) a.*Saw I Joe the plumber.
b.*Her saw the plumber Joe.

VPs can be 'modified' by PPs

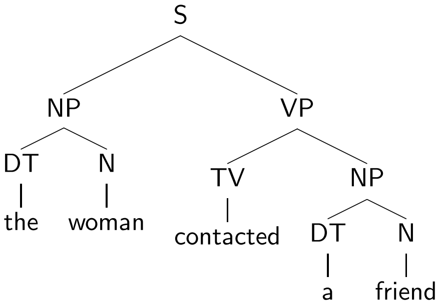
(22) a. Tom ate a burger.
b. Tom ate a burger in the bar.
c. Tom ate a burger in the bar, near the window.
d. Tom ate a burger in the bar, near the window, at noon.
e. ...

New rule:
VP → VP PP



Our phrasal grammar rules so far:

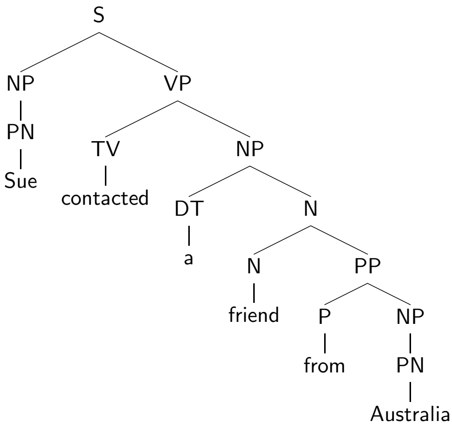
1. NP → DT N
2. N → N PP
3. N → Adj N
4. NP → PN
5. NP → PRP
6. PP → P NP
7. S → NP VP
8. VP → IV
9. VP → TV NP
10. VP → DTV NP NP
11. VP → DTV NP PP
12. VP → VP PP



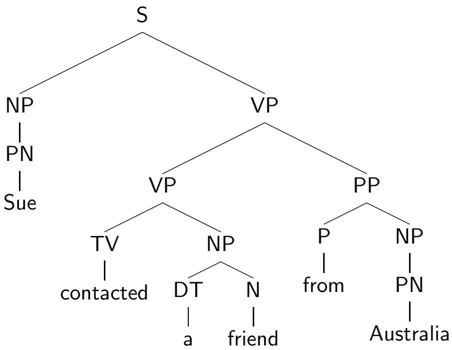
Where can '*from Australia*' go?

Grammars

$N \rightarrow N\ PP$



$VP \rightarrow VP\ PP$



(23) Sue contacted a friend from Australia.
Interpretation 1 (I1): involves a friend from Australia
Interpretation 2 (I2): involves Sue being in Australia

This ambiguity vanishes in the **subject position**:

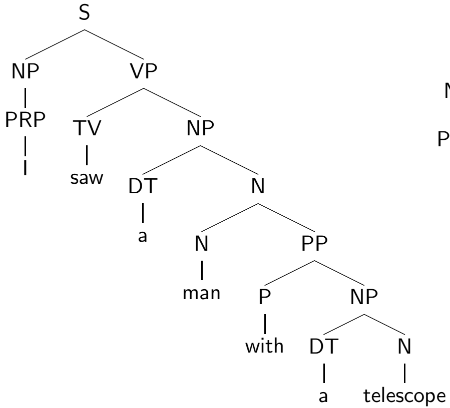
(24) A friend from Australia contacted Sue. (I1)

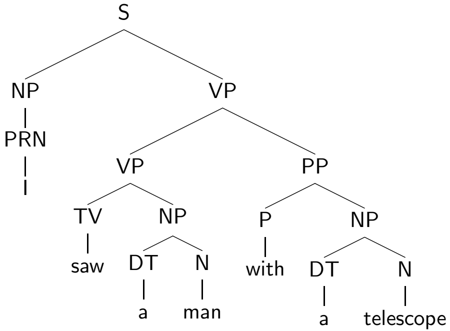
And in **it-cleft** sentences:

(25) a. It was a friend from Australia that Sue contacted. (I1)
b. It was from Australia that Sue contacted a friend. (I2)
c. It was a friend that Sue contacted from Australia. (I2)

The same set of rules explains the ambiguity of (26).

(26) I saw a man with a telescope.



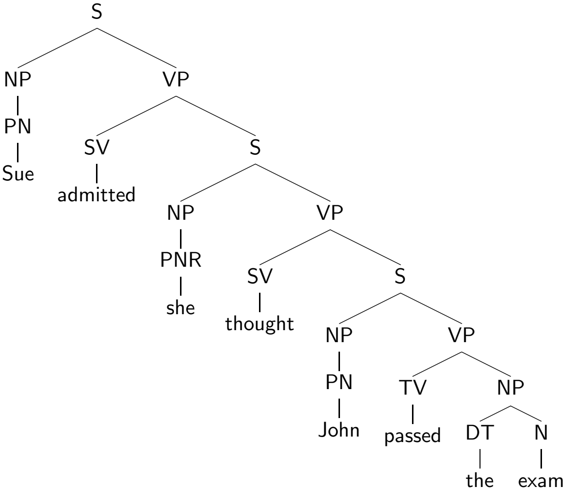


Indirect discourse structures

VP → SV S

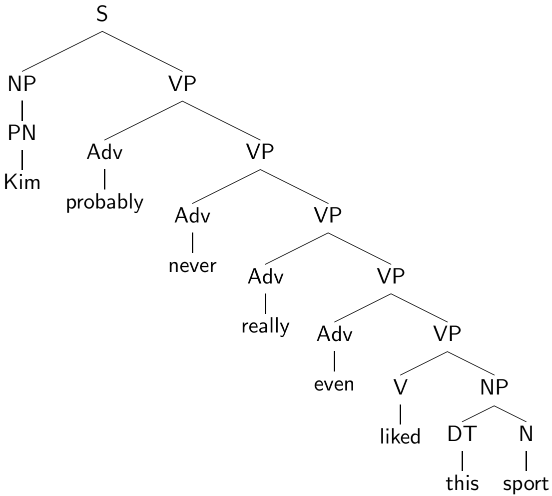
SV → *said* | *thought* | *feared* | *denied* | *shouted* | *wished* | ...

(27) I think [Sue admitted [she thought [John passed the exam]]].



Adverbial adjunction

VP $\rightarrow$ ADV VP



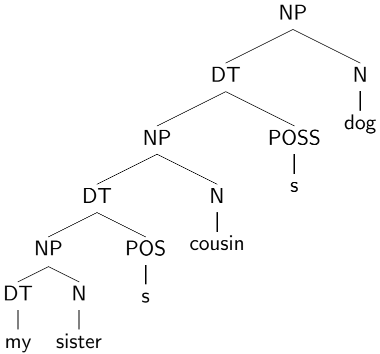
Grammars

Genitive structures

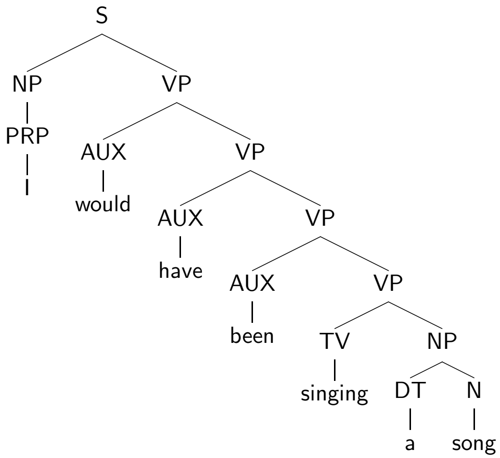
DT → NP POS

POS → 's

(28) a. [[My father]'s tortoise] died.
b. [[[[My father]'s father]'s father]'s tortoise] died.
c. [[[[[My teacher]'s sister]'s best friend]'s cousin]'s dog] died.



Other sources of recursion:



]

VP → AUX VP

More sources of recursion:

$X \rightarrow X_1$ .... X_{n-1} CRD X_n
CRD $\rightarrow$ and | or | but

(29) a. [[John] and [the neighbor]] caught the cat.
b. [[The car and the truck]] collided.

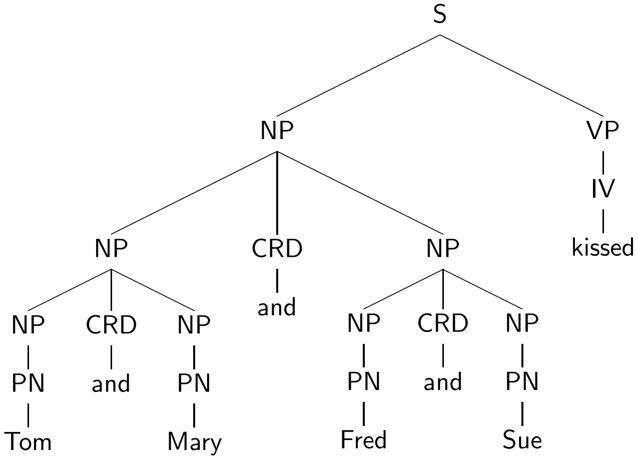
(30) a. The trips [[to France] and [to Iceland]] were great.
b. Fred relied either [[on Sue] or [on Kim]].

(31) a. I bought the [[big] and [yellow]] book.
b. Robin is both [[tall] and [slim]].

(32) a. Kim [[washed the car] and [cleaned the garage]].
b. You can't [[drink booze] and [drive a car]].

(33) a. [[Tom went to NY] and [Mary stayed in California]].
b. He said [[Kim was friendly] and [Bob was annoying]].

Embedding coordination:



(34) John met with either [[John and Mary] or [Tom and Sue]].

Grammars

How big is the English Grammar?

Nobody knows. Still an open problem.
Active formal research started in the 50s.

Modern large-scale computational grammars:

- Hand-written feature-based grammar
  ERG: total of 266 rules (morphology + syntax)
- Corpora-induced grammars: 17,500 rules (Penn Treebank)
  Over 17k (J&M'09:§12.4.2)

Do we really need type 2 (context-free) grammars?

Yes: some structures have the form $a^n b^n$

(35) a. The movie was *Inception*.
b. The movie everyone loved was *Inception*.
c. The movie everyone I interviewed loved was *Inception*.

Do we need type 1 (context-sensitive) grammars?

Yes: some structures have the form $a^n b^m c^n d^m$

- Swiss German (Shieber 1985)
- Dutch (Bresnan & al. 1982)
- Bambara (Culy 1985)
- ... etc. (Walther 1999, Beesley & Karttunen 2000).

Dutch example: $...noun_1^n noun_2^m ... verb_1^n verb_2^m ...$

(36) ... dat ik$_1$ Henk$_2$ haar$_3$ de nijlpaarden$_3$ zag$_1$ helpen$_2$ voeren$_3$
... that I Henk her the hippos saw help feed
'... that I saw Henk help her feed the hippos'

Some formalisms 'push' the grammar into the lexicon

A categorial grammar consists of:

1. Categories
   1. Basic Categories: S, NP, PP, ...
   2. Complex Categories:
      If X and Y are a categories, then so are X/Y and X\Y.
2. Two functional rules

   (X/Y) Y ⇒ X (Forward composition (>))
   Y (X\Y) ⇒ X (Backward composition (<))

Example:

(37) Tom sneezed.

$$\dfrac{\dfrac{Tom}{\mathrm{NP}} \quad \dfrac{sneezed}{(\mathrm{S}\backslash\mathrm{NP})}}{\mathrm{S}}<$$

(38) Kim saw Sandy.

$$\cfrac{\cfrac{Kim}{\mathrm{NP}} \quad \cfrac{\cfrac{saw}{((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})} \quad \cfrac{Sandy}{\mathrm{NP}}}{(\mathrm{S}\backslash\mathrm{NP})}{>}}{\mathrm{S}}{<}$$

(39) Kim sneezed yesterday.

$$\cfrac{\cfrac{Kim}{\mathrm{NP}} \quad \cfrac{\cfrac{sneezed}{(\mathrm{S}\backslash\mathrm{NP})} \quad \cfrac{yesterday}{(\mathrm{S}\backslash\mathrm{NP})\backslash(\mathrm{S}\backslash\mathrm{NP})}}{(\mathrm{S}\backslash\mathrm{NP})}{<}}{\mathrm{S}}{<}$$

(40) Tom saw Sue.

$$\dfrac{\dfrac{Tom}{\mathrm{NP}} \qquad \dfrac{\dfrac{saw}{((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})} \qquad \dfrac{Sue}{\mathrm{NP}}}{(\mathrm{S}\backslash\mathrm{NP})}\,>}{\mathrm{S}}\,<$$

(41) Tom gave Sue a yellow cat.

$$\dfrac{\dfrac{Tom}{\mathrm{NP}} \qquad \dfrac{\dfrac{\dfrac{gave}{(((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})/\mathrm{NP})} \qquad \dfrac{Sue}{\mathrm{NP}}}{((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})}\,> \qquad \dfrac{\dfrac{a}{\mathrm{DT}} \qquad \dfrac{\dfrac{yellow}{(\mathrm{NP}\backslash\mathrm{DT})/(\mathrm{NP}\backslash\mathrm{D})} \qquad \dfrac{cat}{(\mathrm{NP}\backslash\mathrm{DT})}}{(\mathrm{NP}\backslash\mathrm{DT})}\,>}{\mathrm{NP}}\,<}{(\mathrm{S}\backslash\mathrm{NP})}\,>}{\mathrm{S}}\,<$$

(42) The owner of the bar called you.

$$\dfrac{\dfrac{\dfrac{\textit{the}}{\mathrm{DT}} \quad \dfrac{\dfrac{\textit{owner}}{(\mathrm{NP}\backslash\mathrm{DT})/\mathrm{PP}} \quad \dfrac{\dfrac{\textit{of}}{\mathrm{PP}/\mathrm{NP}} \quad \dfrac{\dfrac{\textit{the}}{\mathrm{DT}} \quad \dfrac{\textit{bar}}{(\mathrm{NP}\backslash\mathrm{DT})}}{\mathrm{NP}}{<}}{\mathrm{PP}}{>}}{\mathrm{NP}\backslash\mathrm{DT}}{>}}{\mathrm{NP}}{<} \quad \dfrac{\dfrac{\textit{called}}{(\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP}} \quad \dfrac{\textit{you}}{\mathrm{NP}}}{\mathrm{S}\backslash\mathrm{NP}}{>}}{\mathrm{S}}{>}$$

Coordination

and : $(X\backslash X)/X$

(43) Sue and Tom

$$\dfrac{\dfrac{Sue}{\mathrm{NP}} \quad \dfrac{\dfrac{and}{(\mathrm{NP}\backslash\mathrm{NP})/\mathrm{NP}} \quad \dfrac{Tom}{\mathrm{NP}}}{\mathrm{NP}\backslash\mathrm{NP}}>}{\mathrm{NP}}<$$

(44) Tom challenged and defeated Robin.

$$\dfrac{\dfrac{Tom}{\mathrm{NP}} \quad \dfrac{\dfrac{\dfrac{played}{(\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP}} \quad \dfrac{\dfrac{and}{(((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})\backslash((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP}))/((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})} \quad \dfrac{defeated}{(\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP}}}{((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})\backslash((\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP})}>}{(\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP}}< \quad \dfrac{Robin}{\mathrm{NP}}}{\mathrm{S}\backslash\mathrm{NP}}>}{s}<$$

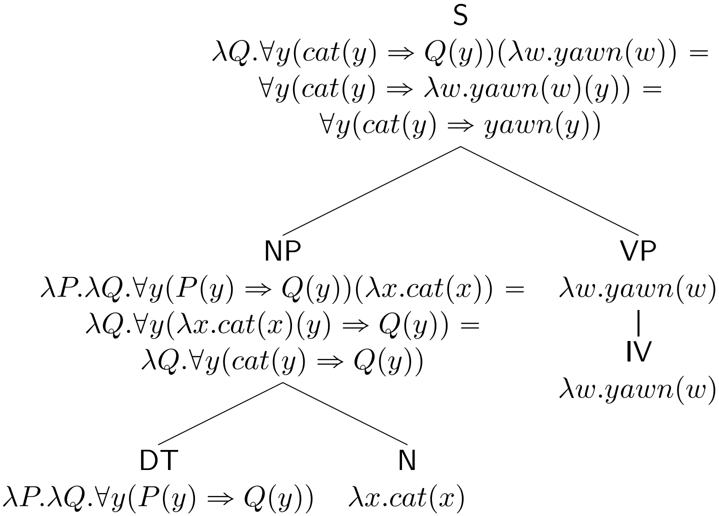
Grammars

One of the great advantages of having access to syntactic structure is that this gives us some access to semantic structure.

(45) a. *cat* : N ; $\lambda x.cat(x)$
b. *every* : DT ; $\lambda P.\lambda Q.\forall y(P(y) \Rightarrow Q(y))$
c. *a* : DT ; $\lambda P.\lambda Q.\exists y(P(y) \wedge Q(y))$
d. *yawn* : IV ; $\lambda w.yawn(w)$

- $\forall x(\phi)$ translates as 'for all values of x, ϕ is true'
- $\exists x(\phi)$ translates as 'for some value of x, ϕ is true'
- $P(x)$ translates as the value of x has property P
- $\phi \wedge \psi$ translates as 'ϕ and ψ are both true'
- $\phi \Rightarrow \psi$ translates as 'if ϕ is true then ψ is true too'
- $\lambda x.\phi$ translates as 'replace x's in ϕ with a given argument'

(46) a. NP;$\phi(\psi)$ $\rightarrow$ DT;ϕ N;ψ
b. VP;ϕ $\rightarrow$ IV;ϕ
c. S;$\phi(\psi)$ $\rightarrow$ NP;ϕ VP;ψ


S
$\lambda Q.\forall y(cat(y) \Rightarrow Q(y))(\lambda w.yawn(w)) =$
$\forall y(cat(y) \Rightarrow \lambda w.yawn(w)(y)) =$
$\forall y(cat(y) \Rightarrow yawn(y))$
NP
$\lambda P.\lambda Q.\forall y(P(y) \Rightarrow Q(y))(\lambda x.cat(x)) =$
$\lambda Q.\forall y(\lambda x.cat(x)(y) \Rightarrow Q(y)) =$
$\lambda Q.\forall y(cat(y) \Rightarrow Q(y))$
VP
$\lambda w.yawn(w)$
IV
$\lambda w.yawn(w)$
DT
$\lambda P.\lambda Q.\forall y(P(y) \Rightarrow Q(y))$
N
$\lambda x.cat(x)$

Grammars

More words and rules:

1. *Robin* : PN ; $\lambda P.P(robin)$
2. *chased* : TV ; $\lambda R.\lambda z.R(\lambda w.chase(z,w))$

1. NP;ϕ $\rightarrow$ PN;ϕ
2. VP;$\phi(\psi)$ $\rightarrow$ TV;ϕ NP;ψ

S

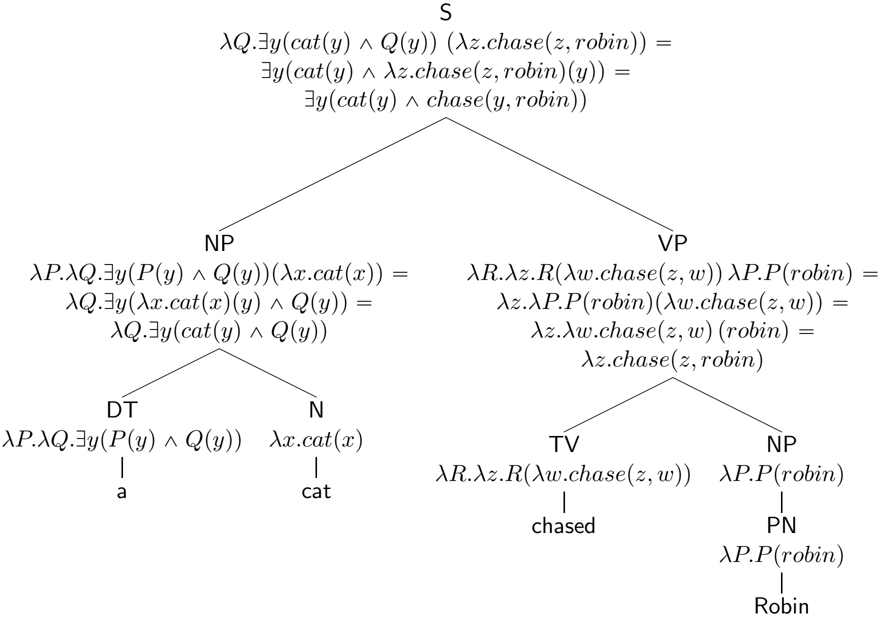
$$\lambda P.P(robin)(\lambda w.yawn(w)) =$$
$$\lambda w.yawn(w)(robin) =$$
$$yawn(robin)$$

NP $\lambda P.P(robin)$ — VP $\lambda w.yawn(w)$

PN $\lambda P.P(robin)$ — IV $\lambda w.yawn(w)$

Robin — yawned

- S
  $$\lambda P.P(robin)\ (\lambda z.\forall y(cat(y) \Rightarrow chase(z,y))) =$$
  $$\lambda z.\forall y(cat(y) \Rightarrow chase(z,y))\ (robin) =$$
  $$\forall y(cat(y) \Rightarrow chase(robin,y))$$
  - NP
    $\lambda P.P(robin)$
    - PN
      $\lambda P.P(robin)$
      - Robin
  - VP
    $$\lambda R.\lambda z.R(\lambda w.chase(z,w))\,(\lambda Q.\forall y(cat(y) \Rightarrow Q(y))) =$$
    $$\lambda z.\lambda Q.\forall y(cat(y) \Rightarrow Q(y))\ (\lambda w.chase(z,w)) =$$
    $$\lambda z.\forall y(cat(y) \Rightarrow \lambda w.chase(z,w)(y)) =$$
    $$\lambda z.\forall y(cat(y) \Rightarrow chase(z,y))$$
    - TV
      $\lambda R.\lambda z.R(\lambda w.chase(z,w))$
      - chased
    - NP
      $$\lambda P.\lambda Q.\forall y(P(y) \Rightarrow Q(y))(\lambda x.cat(x)) =$$
      $$\lambda Q.\forall y(\lambda x.cat(x)(y) \Rightarrow Q(y)) =$$
      $$\lambda Q.\forall y(cat(y) \Rightarrow Q(y))$$
      - DT
        $\lambda P.\lambda Q.\forall y(P(y) \Rightarrow Q(y))$
        - every
      - N
        $\lambda x.cat(x)$
        - cat


S
$\lambda Q.\exists y(cat(y) \wedge Q(y))\ (\lambda z.chase(z, robin)) =$
$\exists y(cat(y) \wedge \lambda z.chase(z, robin)(y)) =$
$\exists y(cat(y) \wedge chase(y, robin))$
NP
$\lambda P.\lambda Q.\exists y(P(y) \wedge Q(y))(\lambda x.cat(x)) =$
$\lambda Q.\exists y(\lambda x.cat(x)(y) \wedge Q(y)) =$
$\lambda Q.\exists y(cat(y) \wedge Q(y))$
VP
$\lambda R.\lambda z.R(\lambda w.chase(z, w))\, \lambda P.P(robin) =$
$\lambda z.\lambda P.P(robin)(\lambda w.chase(z, w)) =$
$\lambda z.\lambda w.chase(z, w)\,(robin) =$
$\lambda z.chase(z, robin)$
DT
$\lambda P.\lambda Q.\exists y(P(y) \wedge Q(y))$
a
N
$\lambda x.cat(x)$
cat
TV
$\lambda R.\lambda z.R(\lambda w.chase(z, w))$
chased
NP
$\lambda P.P(robin)$
PN
$\lambda P.P(robin)$
Robin

S

$$\lambda Q.\forall y(dog(y) \Rightarrow Q(y))\ (\lambda z.\exists k(cat(k) \wedge chase(z,k))) =$$
$$\forall y(dog(y) \Rightarrow \lambda z.\exists k(cat(k) \wedge chase(z,k))(y)) =$$
$$\forall y(dog(y) \Rightarrow \exists k(cat(k) \wedge chase(y,k)))$$

NP

$$\lambda P.\lambda Q.\forall y(P(y) \Rightarrow Q(y))(\lambda x.dog(x)) =$$
$$\lambda Q.\forall y(\lambda x.dog(x)(y) \Rightarrow Q(y)) =$$
$$\lambda Q.\forall y(dog(y) \Rightarrow Q(y))$$

DT

$\lambda P.\lambda Q.\forall y(P(y) \Rightarrow Q(y))$

every

N

$\lambda x.dog(x)$

dog

VP

$$\lambda R.\lambda z.R(\lambda w.chase(z,w))\ \lambda Q'.\exists k(cat(k) \wedge Q'(k)) =$$
$$\lambda z.\lambda Q'.\exists k(cat(k) \wedge Q'(k))(\lambda w.chase(z,w)) =$$
$$\lambda z.\exists k(cat(k) \wedge \lambda w.chase(z,w)(k)) =$$
$$\lambda z.\exists k(cat(k) \wedge chase(z,k))$$

TV

$\lambda R.\lambda z.R(\lambda w.chase(z,w))$

chased

NP

$$\lambda P'.\lambda Q'.\exists k(P(k) \wedge Q'(k))(\lambda y.cat(y)) =$$
$$\lambda Q'.\exists k(\lambda y.cat(y)(k) \wedge Q'(k)) =$$
$$\lambda Q'.\exists k(cat(k) \wedge Q'(k))$$

DT

$\lambda P'.\lambda Q'.\exists k(P'(k) \wedge Q'(k))$

a

N

$\lambda y.cat(y)$

cat