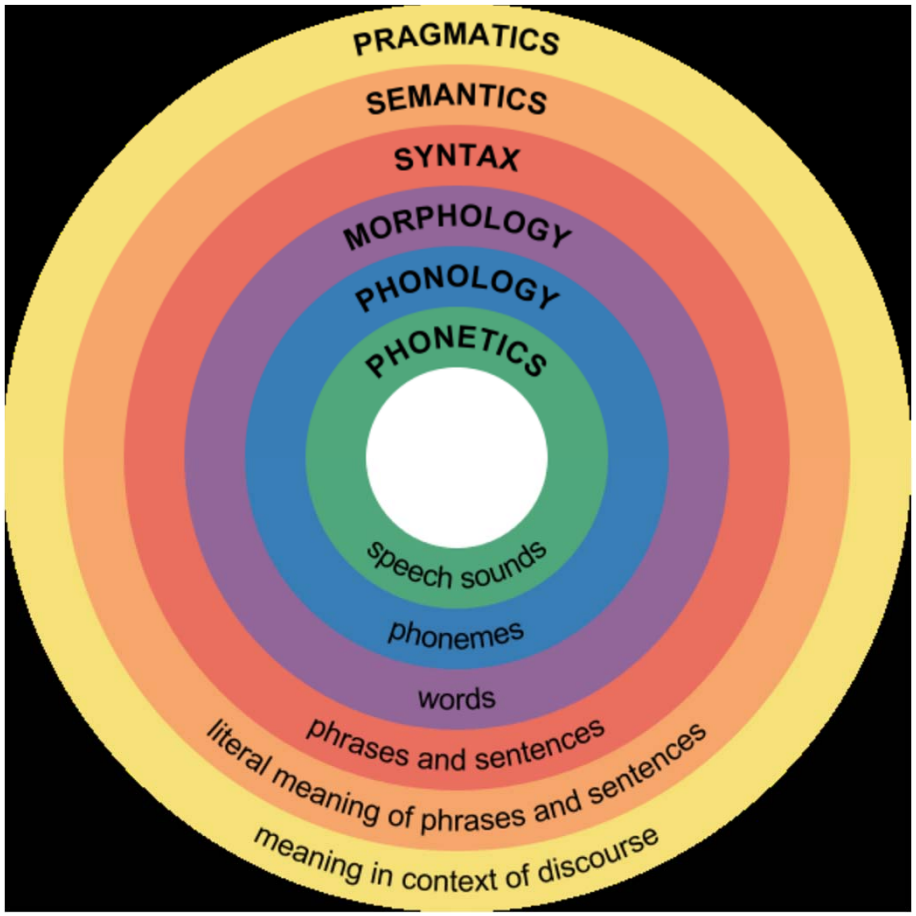
LAP 3: COMPUTATIONAL SYNTAX INTRODUCTION

Ruben Urizar

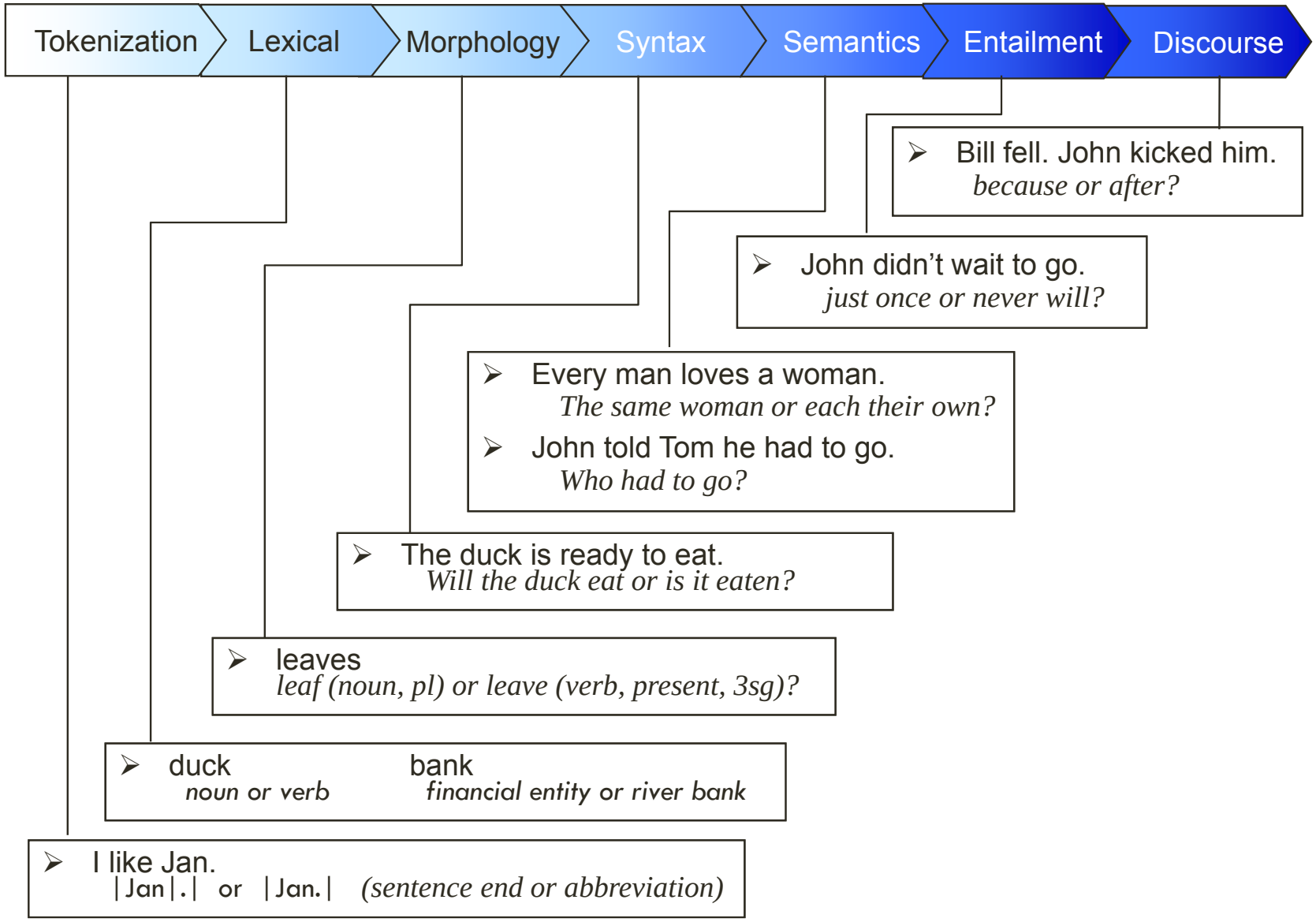
SYNTAX IN CONTEXT



WHO CARES?

- Grammar checkers
- Question answering
- Information extraction
- Machine Translation
- Generation
- …

AMBIGUITY


Tokenization
Lexical
Morphology
Syntax
Semantics
Entailment
Discourse
Bill fell. John kicked him.
because or after?
John didn't wait to go.
just once or never will?
Every man loves a woman.
The same woman or each their own?
John told Tom he had to go.
Who had to go?
The duck is ready to eat.
Will the duck eat or is it eaten?
leaves
leaf (noun, pl) or leave (verb, present, 3sg)?
duck
noun or verb
bank
financial entity or river bank
I like Jan.
|Jan|.| or |Jan.| (sentence end or abbreviation)

SYNTACTIC AMBIGUITY (i)

There are many types of syntactic ambiguity, just to mention some:

1. **PoS ambiguity** occurs when at least two words can belong to two or more parts of speech.
   - ✓ *They can fish.*
   - ✓ *I saw her duck.*

   This is **quite rare** despite the large numbers of words that can be both nouns and verbs in English.

SYNTACTIC AMBIGUITY (ii)

2. **PP** (prepositional phrase) **attachment.** PPs can modify VPs as well as NPs:
   - ✓ *She saw a man with a telescope*
     - [She] [saw] [a man] [*with a telescope*].
     - [She] [saw] [a man *with a telescope*].
   - ✓ *Peter waved Mary from the school with a flag (...)*
     - Peter waved [from the school with a flag] => (school with a flag)
     - Peter waved [from the school] [with a flag] => (Peter waved with a flag)

SYNTACTIC AMBIGUITY (iii)

3. **Coreference:**
   - ✓ *John told Tom he had to go*
     - he = John => *John had to go*
     - he = Tom => *Tom had to go.*

AMBIGUITY RESOLUTION (i)

- Sometimes ambiguity cannot be resolved; we cannot know what the producer of the sentence meant:

→ *I saw her duck:*

1. `I saw [NP her duck].`
2. `I saw [NP her] [VP duck].`

→ *They can fish:*

1. `They [VP can fish].`
2. `They [VP can] [NP fish].`

AMBIGUITY RESOLUTION (ii)

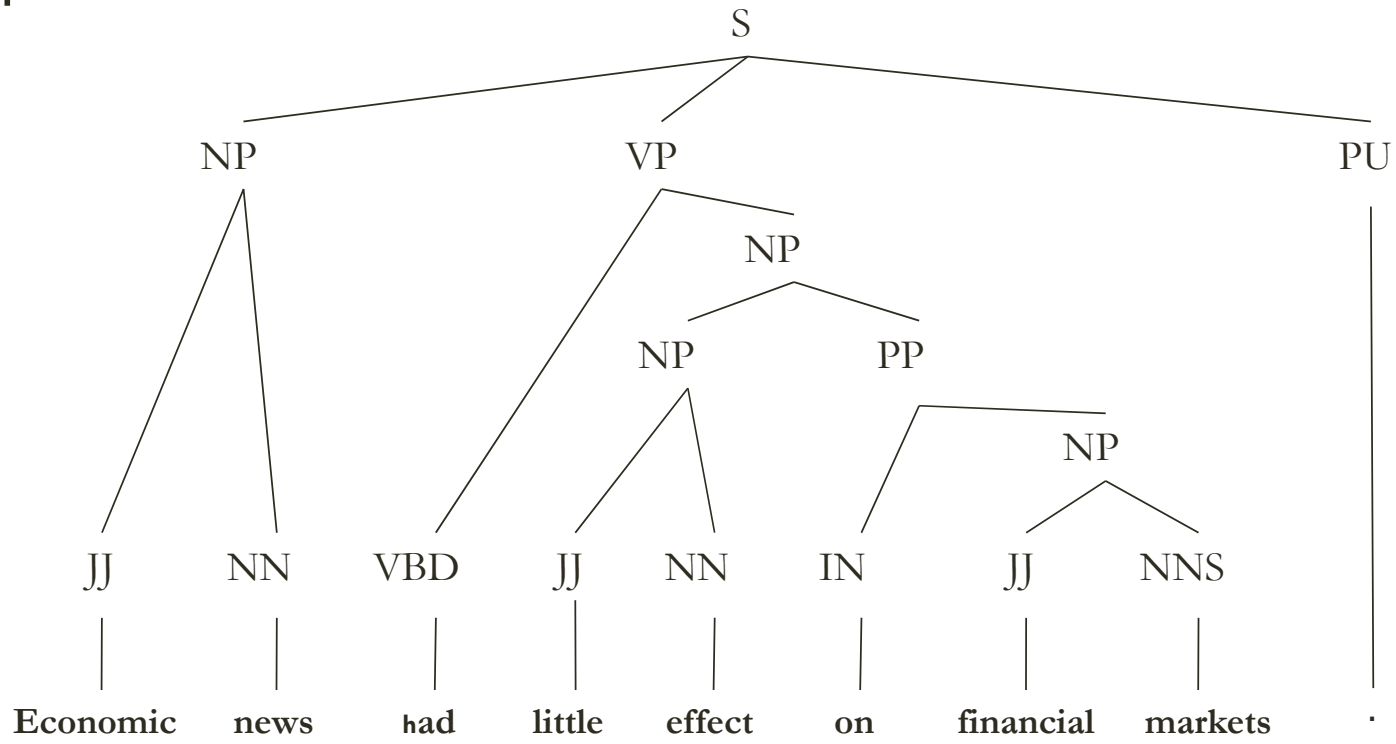
- Often, the **context** helps to identify the correct interpretation:
  → [*Peter waved from the school with a flag.] He kept waving the flag all the time to Mary's house*. ⇨ 'man with a flag' interpretation

- Sometimes, **world knowledge** (or 'commonsense knowledge') is needed:
  → *Peter waved from the school with a balcony...* ⇨ 'school with a balcony' interpretation (NOT 'man with a balcony')

- Many ambiguity cases can be resolved using **linguistic context**:
  → *I saw a duck in the pond* ⇨ 'duck' noun
  → *What should you do when she leaves you?* ⇨ leaves verb

TWO VIEWS OF LINGUISTIC STRUCTURE

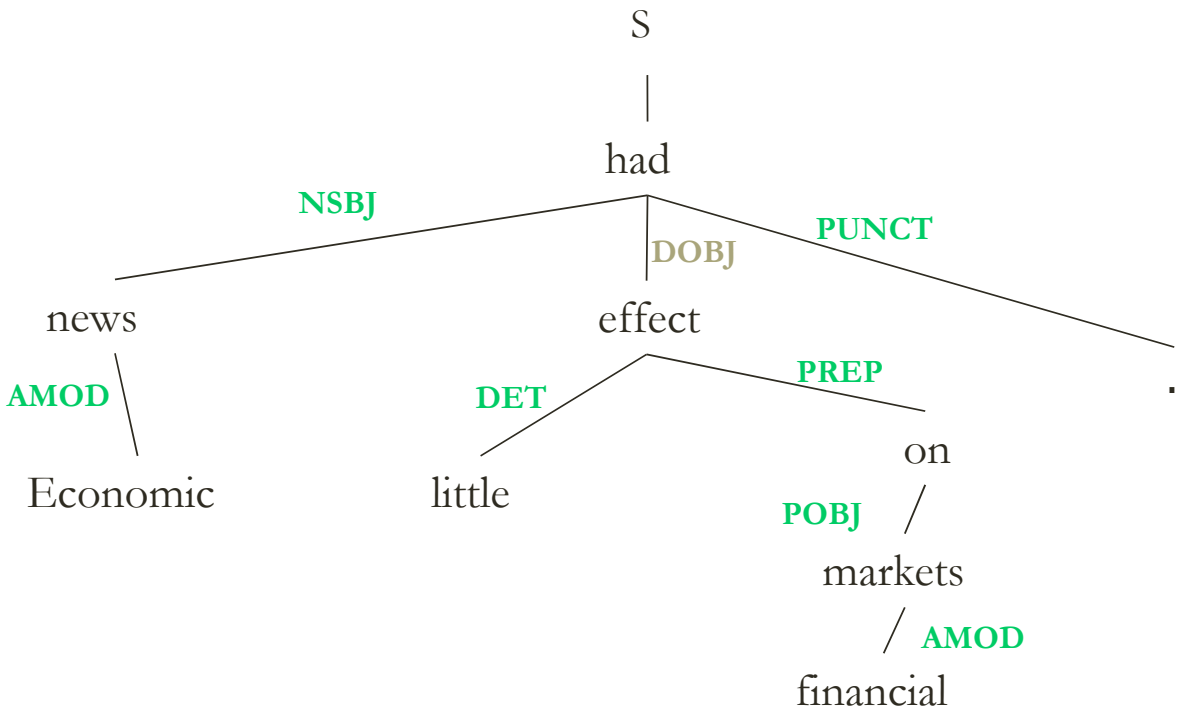
1. **Phrase structure** organizes words into nested constituents or phrase structures.



TWO VIEWS OF LINGUISTIC STRUCTURE

2. **Dependency structure** shows which words depend on (modify or are arguments of) which other words.

Economic news had little effect on financial markets



TWO VIEWS OF LINGUISTIC STRUCTURE

Dependency structure (2)

Economic news had little effect on financial markets

- root (ROOT-0 , had-3)
- nsubj (had-3 , news-2)
- amod (news-2 , Economic-1)
- dobj (had-3 , effect-5)
- amod (effect-5 , little-4)
- prep (effect-5 , on-6)
- amod (markets-8 , financial-7)

APPROACHES TO SYNTAX (i)

Both tasks (giving all the possible interpretations of a sentence and choosing one out of them) can be tackled with different approaches:

- The knowledge-based way
  - *Using linguistic knowledge to eliminate nonsense analyses and/or choose coherent interpretations.*
- The automatic way
  - *Using statistical methods based on empirical evidence (corpora).*
- A combination of both

APPROACHES TO SYNTAX (ii)

1. Based on linguistic knowledge
   - Context-free grammars (CFG)
   - Unification-based grammars
     - LFG (Lexical Functional Grammar)
     - HPSG (Head-driven Phrase Structure Grammar)
     - PATR-II
     - …
   - Finite-state mechanisms
     - CG (Constraint Grammar)
     - XFST (Xerox Finite State Tool)
2. Probabilistic
3. Combined

Context Free Grammars
CFG

CONTEXT-FREE GRAMMARS

- Formulated by Chomsky (1956) and Backus (1959)
- Capture constituents and ordering
  - Need something else for grammatical relations and dependency relations
- Consists of
  - A set of rules
  - A lexicon

CONTEXT-FREE GRAMMARS

Consist of:

- **A lexicon**
- **A set of rules** (productions) – expressing the way symbols of the language can be grouped together.

CONTEXT-FREE GRAMMAR: lexicon

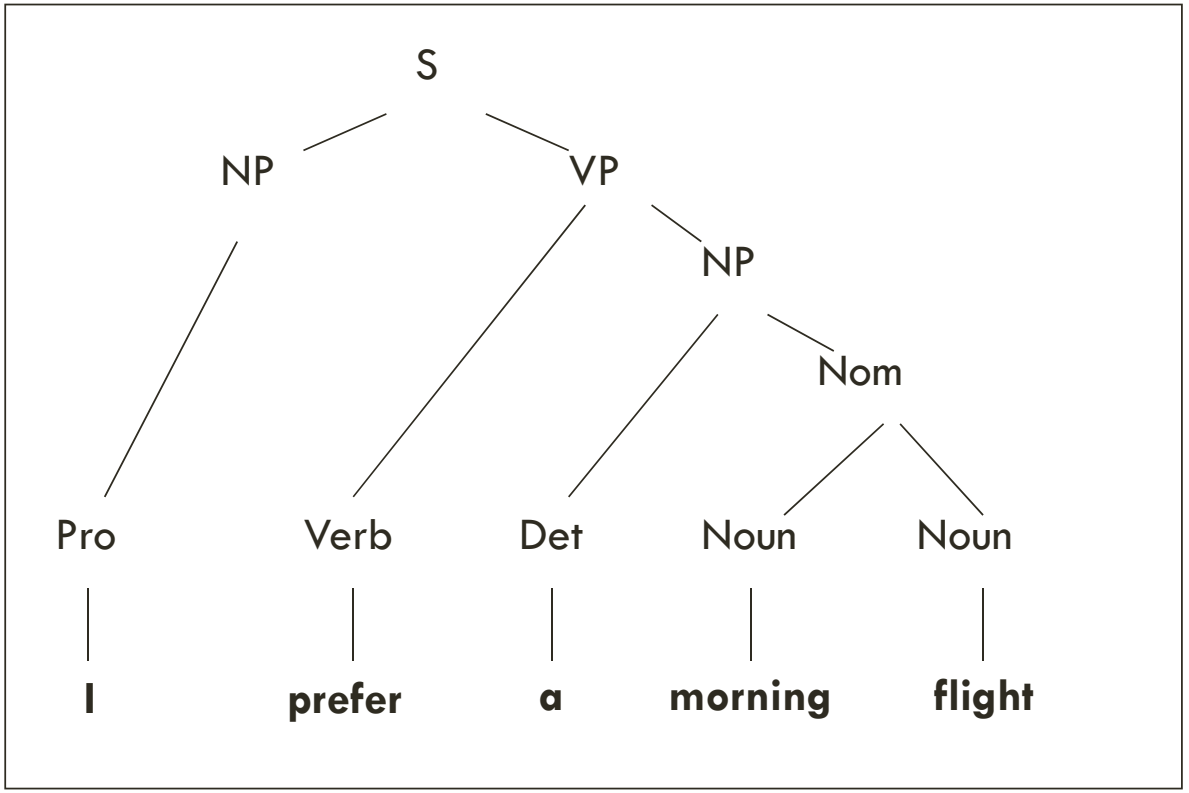
Noun	→	*flights* | *breeze* | *trip* | *morning* | ...
Verb	→	*is* | *prefer* | *like* | *need* | *want* | *fly*
Adjective	→	*cheapest* | *non-stop* | *first* | *latest*
		| *other* | *direct* | ...
Pronoun	→	*me* | *I* | *you* | *it* | ...
Proper-Noun	→	*Alaska* | *Baltimore* | *Los Angeles*
		| *Chicago* | *United* | *American* | ...
Determiner	→	*the* | *a* | *an* | *this* | *these* | *that* | ...
Preposition	→	*from* | *to* | *on* | *near* | ...
Conjunction	→	*and* | *or* | *but* | ...

CONTEXT-FREE GRAMMAR: set of rules

S	$\rightarrow$	*NP VP*	I + want a morning flight
NP	$\rightarrow$	*Pronoun*	I
	|	*Proper-Noun*	Los Angeles
	|	*Det Nominal*	a + flight
Nominal	$\rightarrow$	*Noun Nominal*	morning + flight
	|	*Noun*	flights
VP	$\rightarrow$	*Verb*	do
	|	*Verb NP*	want + a flight
	|	*Verb NP PP*	leave + Boston + in the morning
	|	*Verb PP*	leaving + on Thursday
PP	$\rightarrow$	*Preposition NP*	from + Los Angeles

CONTEXT-FREE GRAMMAR: derivations and trees

GRAMMAR rules			LEXICON entries		
S	→	NP VP	Pro	→	I
VP	→	V NP	V	→	prefer
NP	→	Pro	Det	→	a
NP	→	Det Nom	Noun	→	morning
Nom	→	Noun Noun	Noun	→	flight



Unification-based grammars

FEATURE STRUCTURES (i)

- Context-free grammars do not deal with issues like agreement
- Unification-based grammars use features such as ‘number’, ‘person’, ‘gender’…
- A feature structure allows us to state properties e.g. about a noun phrase

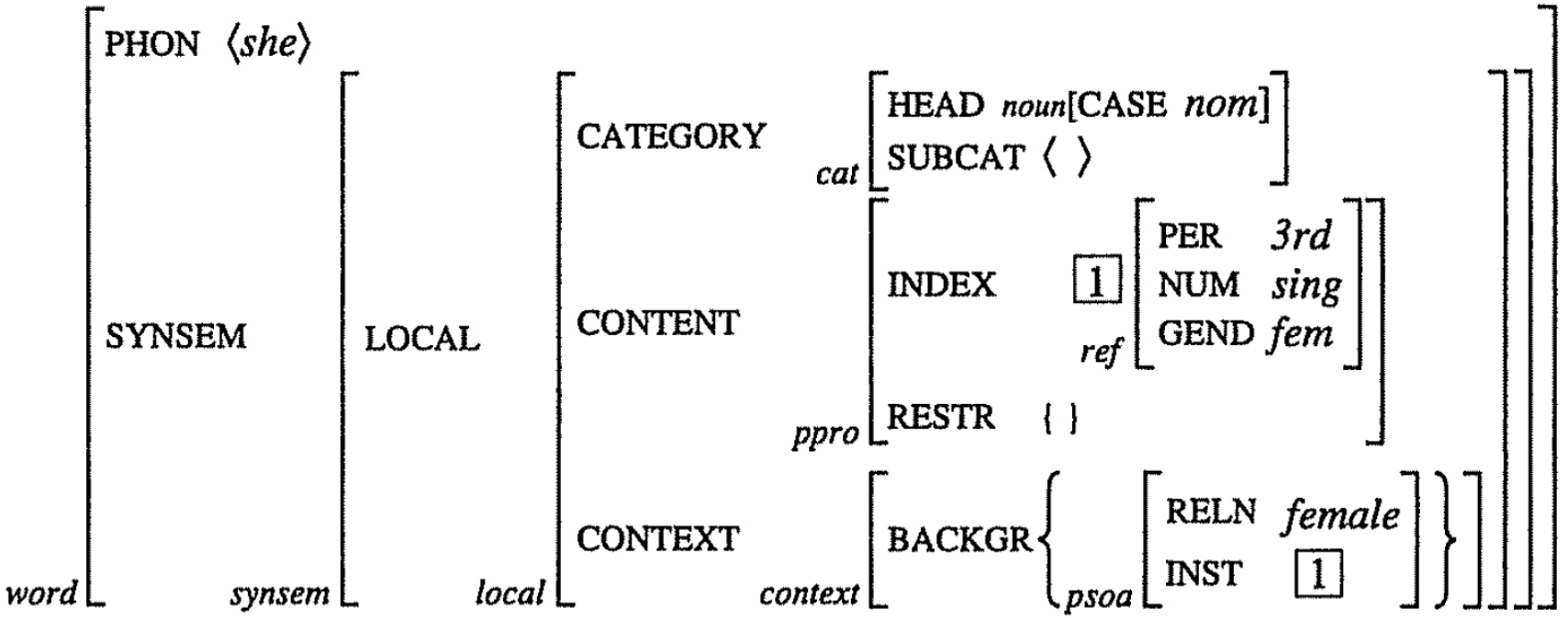
```
[she      NP
number    sg
person    3
gender    fem ]
```

or a verb phrase

```
[walks    VP
number    sg
person    3
tense     present ]
```

FEATURE STRUCTURES (ii)

- Each feature (e.g., 'number') is paired with a value (e.g., 'sg')
- A bundle of feature-value pairs can be put into an attribute-value matrix (AVM)

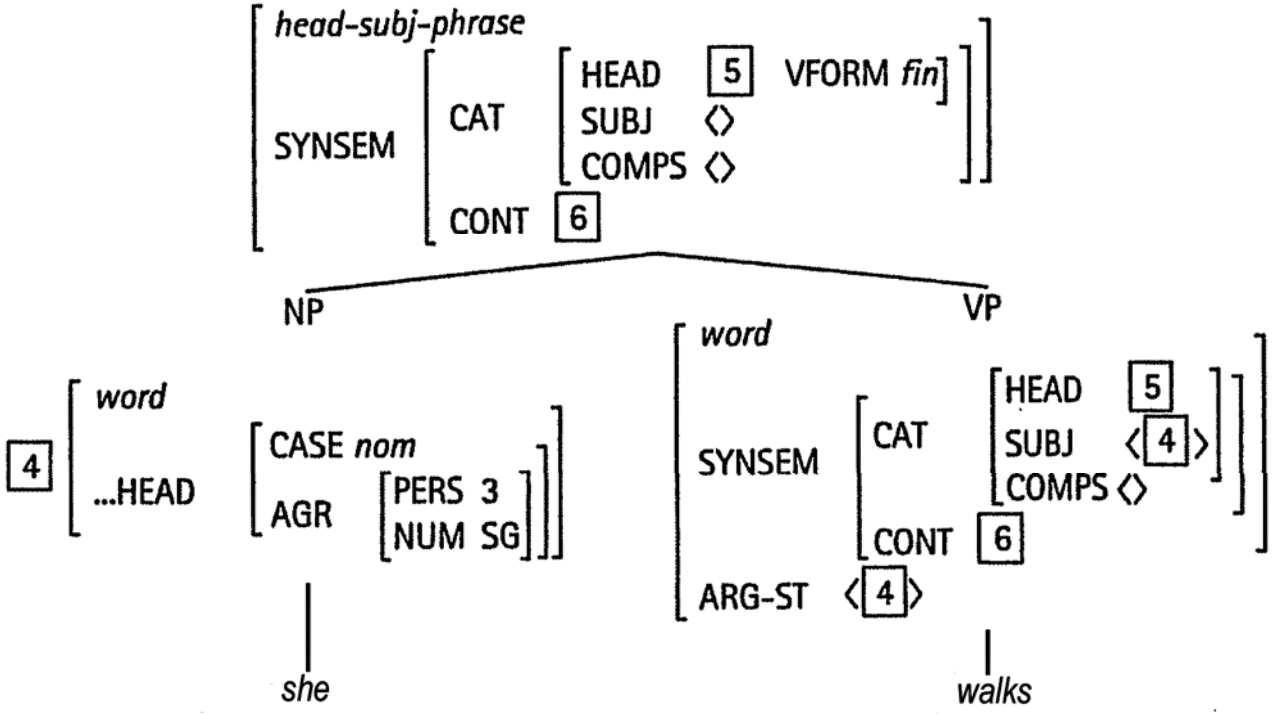


UNIFICATION (i)

- Unification (U) = a basic operation to merge two feature structures into a resultant feature structure (FS)
- The two-feature structures must be compatible, i.e., have no values that conflict
  - **Identical** FSs:
    - [number sg] U [number sg] = [number sg]
  - **Conflicting** FSs:
    - [number sg] U [number pl] = Fail
  - **Merging** with an unspecified FS:
    - [number sg] U [number []] = [number sg]

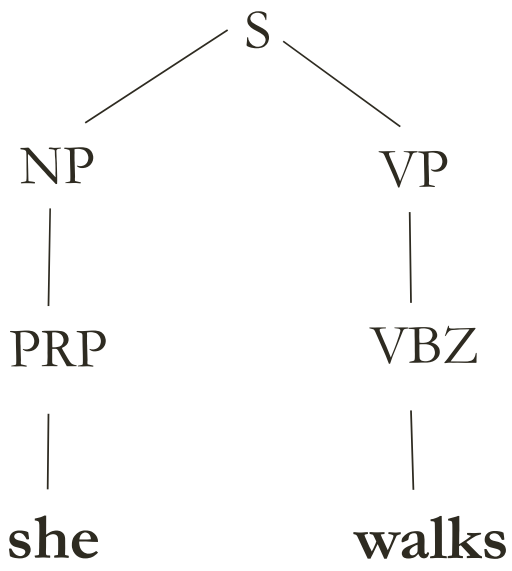
UNIFICATION (ii)

A feature structure matrix of type *phrase* in HPSG states that the VP must agree with its subject in number and person (3 sg). In this case, the feature structures of *she* and *walks* are compatible so unification takes place.



UNIFICATION (iii)

The resulting parse tree would be the following:



Probabilistic Parsing and Treebanks

STATISTICAL PARSING (i)

- The basic idea is:
  - Start with a treebank
    - treebank: a corpus with syntactic annotation, i.e., already-parsed sentences (e.g. the Penn Treebank)
  - Examine which parse trees occur frequently
  - Extract grammar rules corresponding to those parse trees, estimating the probability of the grammar rule based on its frequency
- That is, we'll have a context-free grammar (CFG) augmented with probabilities

STATISTICAL PARSING (ii)

Computing the probabilities of a particular parse tree

- We have the following rules and probabilities:
  - S → VP .05
  - VP → V NP .40
  - NP → Det N .20
  - V → book .30
  - Det → that .05
  - N → flight .25
- Being P(T) = the probability of a particular parse tree,

P(T) = P(S→VP)*P(VP→V NP)*…*P(N→flight)
= .05*.40*.20*.30*.05*.25 = .000015, or 1.5 x 10-5

USING PROBABILITIES

- So, the probability for that parse is 0.000015. What's the big deal?
  - Probabilities are useful for comparing with other probabilities
- Whereas we couldn't decide between two parses using a regular CFG, we now can.
- For example, "`TWA flights`" is ambiguous between being two separate NPs (as in I gave [NP John] [NP money]) or one NP:
  - A: `[book [TWA] [flights]]`
  - B: `[book [TWA flights]]`
- Comparing probabilities (previous slide) allows us to choose option B

TREEBANKS

The rise of annotated data: The Penn Treebank

```
( (S
    (NP-SBJ (DT The) (NN move))
    (VP (VBD followed)
      (NP
        (NP (DT a) (NN round))
        (PP (IN of)
          (NP
            (NP (JJ similar) (NNS increases))
            (PP (IN by)
              (NP (JJ other) (NNS lenders)))
            (PP (IN against)
              (NP (NNP Arizona) (JJ real) (NN estate) (NNS loans))))))
      (, ,)
      (S-ADV
        (NP-SBJ (-NONE- *))
        (VP (VBG reflecting)
          (NP
            (NP (DT a) (VBG continuing) (NN decline))
            (PP-LOC (IN in)
              (NP (DT that) (NN market)))))))
    (. .)))
```

TREEBANKS

- Building a treebank may seem a lot slower and less useful than building a grammar... but
- A treebank gives us many things
  - Reusability of the labor
  - Broad coverage
  - Frequencies and distributional information
  - A way to evaluate systems

Parts of Speech
PoS

PARTS OF SPEECH (PoS)

… or ‘word classes’

Here are some sentences extracted from different conversations in a café.

- *Our friends are sitting in the corner, look.*
- *I have an important conference at work tomorrow, so I am rather busy*
- *Would you like to come to our party on Saturday, Jessica?*
- *This coffee is really good.*
- *And it’s cheap here.*

1. **What different parts of speech can you distinguish?**
2. **Classify all the words in their corresponding part of speech**

PARTS OF SPEECH (PoS)

1. **Verb**
   - **Lexical:** *have, am, is, like, come, sitting, look*
   - **Auxiliary:** *would, are*
2. **Noun**
   - **Common:** *conference, work, coffee, party, Saturday, friends, corner*
   - **Proper:** *Jessica*
3. **Adjective:** *important, busy, good cheap*
4. **Adverb:** *really, tomorrow, rather, here*
5. **Preposition:** *at, to, on, in*
6. **Determiner:** *an, this, our, the*
7. **Pronoun:** *I, it, you*
8. **Conjunction:** *because, and*

PENN TREEBANK TAGSET (i)

VERB	**Lexical and Auxiliary (not modal)**	**VB**	Verb, base form
		VBD	Verb, past tense
		VBG	Verb, gerund or present participle
		VBN	Verb, past participle
		VBP	Verb, non-3rd person singular present
		VBZ	Verb, 3rd person singular present
	Modal	**MD**	Modal
NOUN	**Common**	**NN**	Noun, singular or mass
		NNS	Noun, plural
	Proper	**NNP**	Proper noun, singular
		NNPS	Proper noun, plural
ADJECTIVE		**JJ**	Adjective
		JJR	Adj. comparative
		JJS	Adjective, superlative

PENN TREEBANK TAGSET (ii)

ADVERB	**RB**	Adverb
	RBR	Adverb, comparative
	RBS	Adverb, superlative
	WRB	Wh-adverb
PREPOSITION	**IN**	Preposition (or subordinating conjunction)
DETERMINER	**DT**	Determiner
	WDT	Wh-determiner
	PDT	Predeterminer
PRONOUN	**PRP**	Personal pronoun
	PRP$	Possessive pronoun
	WP	Wh-pronoun
	WP$	Possessive wh-pronoun
CONJUNCTION	**CC**	Coordinating conjunction
	IN	Subordinating conjunction (or preposition)

PENN TREEBANK TAGSET (iii)

OTHER	**EX**	Existential there
	CD	Cardinal number
	FW	Foreign word
	LS	List item marker
	POS	Possessive ending
	RP	Particle
	SYM	Symbol
	TO	to
	UH	Interjection

VISLCG TAGSET

Part of Speech Tags	
ADJ	adjective
ADV	adverb
ART	article, cp. <art> (*)
DET	determiner pronoun (inflecting)
IN	interjection
INDP	independent pronoun (non-inflecting)
INFM	infinitive marker
KC	coordinating conjunction
KS	subordinating conjunction
N	noun
PROP	proper noun, name
PRP	preposition
PERS	personal pronoun
V	verb --> PR, IMPF, IMP, INF, PCP[12]

Morphological information Tags	
1P	1. person plural (PERS +, V ¤)
1S	1. person singular (PERS +, V ¤)
2P	2. person plural (PERS +, V ¤)
2S	2. person singular (PERS +, V ¤)
3P	3. person plural (PERS +, V ¤)
3S	3. person singular (PERS +, V ¤)
ACC	accusative case (N, PERS)
COM	comparative degree (ADJ, ADV)
CONJ	conjunction, cp. KS and KC
GEN	genitive case
IMP	imperative (V)
IMPF	past tense (V), also PAST (*)
INF	infinitive (V)
NOM	nominative case
NUM	numeral --> <card>, <ord>
P	plural
PCP	past participle, cp. PCP2, PED (*)
PCP1	present participle, cp. GER, PING (*)
PCP2	past participle, cp. PCP, PED (*)
PED	-ed participle, cp. PCP, PCP2 (*)
PR	present tense --> AKT, PAS
S	singular, cp. SG (*)
SUBJ	subjunctive, cp. CONJ (*)
SUP	superlative degree (ADJ, ADV)
(...)	

EXERCISES

Exercise 01 in eGela

DEPENDENCY TREE

Global emissions of C02 will rise for the first time in four years.

PARSERS

- Stanford: nlp.stanford.edu:8080/corenlp/
- Freeling: nlp.lsi.upc.edu/freeling/demo/demo.php

EXERCISES

Exercise 02 in eGela

Exercise 03 in eGela