Natural Language Processing

2 sessions in the course INF348
at the Ecole Nationale Superieure des Télécommunications,
in Paris/France, in Summer 2011

by Fabian M. Suchanek

Organisation

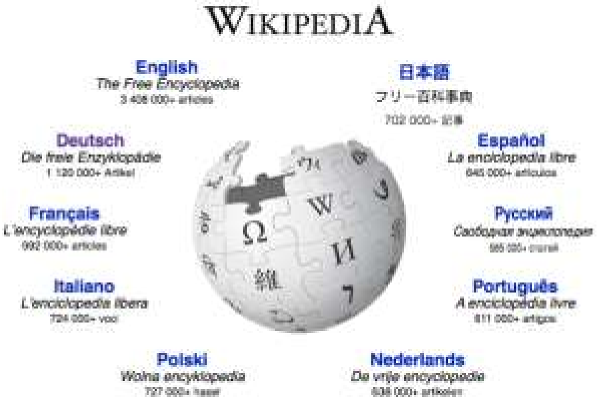
- 2 sessions on Natural Language Processing
  each consisting of 1.5h class + 1.5h practical exercise

- The class will give an overview of Linguistics with special deep divings for natural language processing

- Web-site: http://suchanek.name → Teaching

Natural Language

... appears everywhere on the Internet



WIKIPEDIA

English
The Free Encyclopedia
3 408 000+ articles

日本語
フリー百科事典
702 000+ 記事

Deutsch
Die freie Enzyklopädie
1 120 000+ Artikel

Español
La enciclopedia libre
645 000+ artículos

Français
L'encyclopédie libre
992 000+ articles

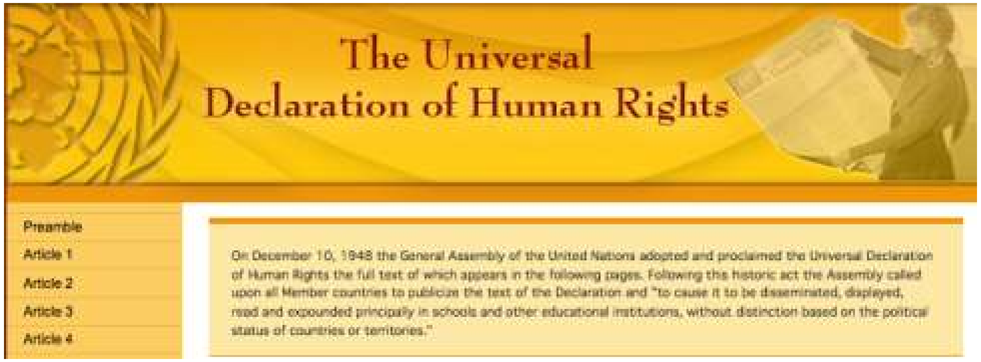
Русский
Свободная энциклопедия
585 000+ статей

Italiano
L'enciclopedia libera
724 000+ voci

Português
A enciclopédia livre
611 000+ artigos

Polski
Wolna encyklopedia
727 000+ haseł

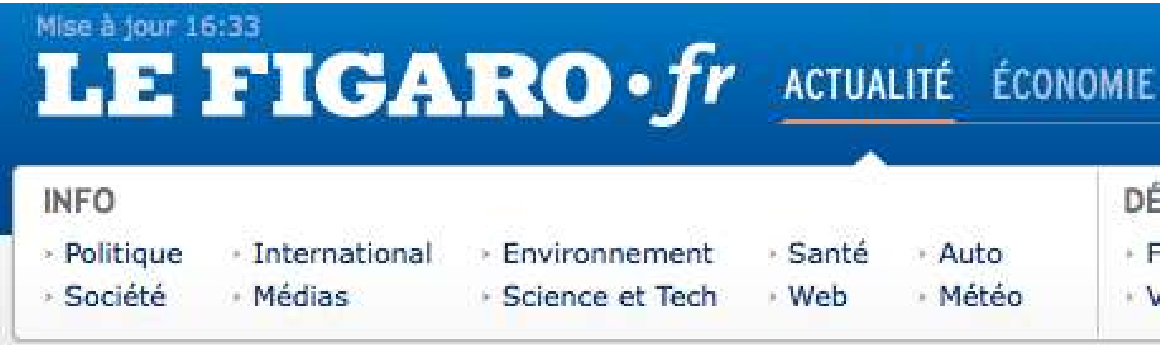
Nederlands
De vrije encyclopedie
638 000+ artikelen



The Universal Declaration of Human Rights

- Preamble
- Article 1
- Article 2
- Article 3
- Article 4

On December 10, 1948 the General Assembly of the United Nations adopted and proclaimed the Universal Declaration of Human Rights the full text of which appears in the following pages. Following this historic act the Assembly called upon all Member countries to publicize the text of the Declaration and "to cause it to be disseminated, displayed, read and expounded principally in schools and other educational institutions, without distinction based on the political status of countries or territories."



Mise à jour 16:33

LE FIGARO • fr

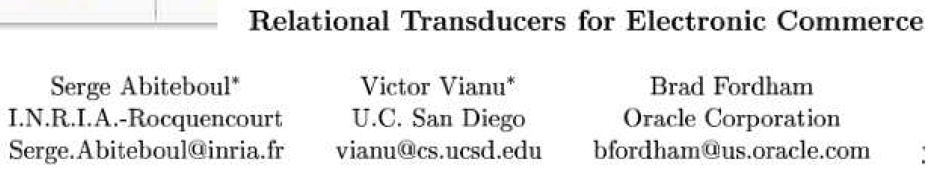
ACTUALITÉ ÉCONOMIE

INFO

- Politique
- Société
- International
- Médias
- Environnement
- Science et Tech
- Santé
- Web
- Auto
- Météo

DÉ

- F
- V



Relational Transducers for Electronic Commerce

Serge Abiteboul*
I.N.R.I.A.-Rocquencourt
Serge.Abiteboul@inria.fr

Victor Vianu*
U.C. San Diego
vianu@cs.ucsd.edu

Brad Fordham
Oracle Corporation
bfordham@us.oracle.com

(1 trillion Web sites;
1 trillion = 10^12
≈ number of cells in the human body)

Natural Language

Inbox: 3726 unread messages

(250 billion mails/day;
≈ number of stars in our galaxy;
80% spam)

Blog.com
What's your story?

(100 million blogs)

me: Would you like to have dinner with me tonight?
Cindy: no.

(1 billion chat msg/day on Facebook;
1 billion = 10^9 = distance Chicago-Tokio in cm)

Natural Language: Tasks

- Automatic text summarization

  Let me first say how proud I am to be the president of this country. Yet, in the past years, our country... [1 hour speech follows]

  → Summary: Taxes will increase by 20%.

- Machine translation

  librairie → book store

- Information Extraction

  Elvis Presley lives on the moon.
  → lives(ElvisPresley, moon)

Natural Language: Tasks

- Natural language understanding

  Close the file! Clean up the kitchen!

- Natural language generation

  Dear user, I have cleaned up the kitchen for you.

- Text Correction

  My hardly loved mother-in law
  → My heartily loved mother-in-law

- Question answering

  Where is Elvis?
  → On the moon

Views on Natural Language

Elvis will be on concert tomorrow in Paris!

For humans:

For a machine: 45 6C 76 69 73 20 77 69 6C…

Linguistics

Linguistics is the study of language.

Linguistics studies language just like biology studies life

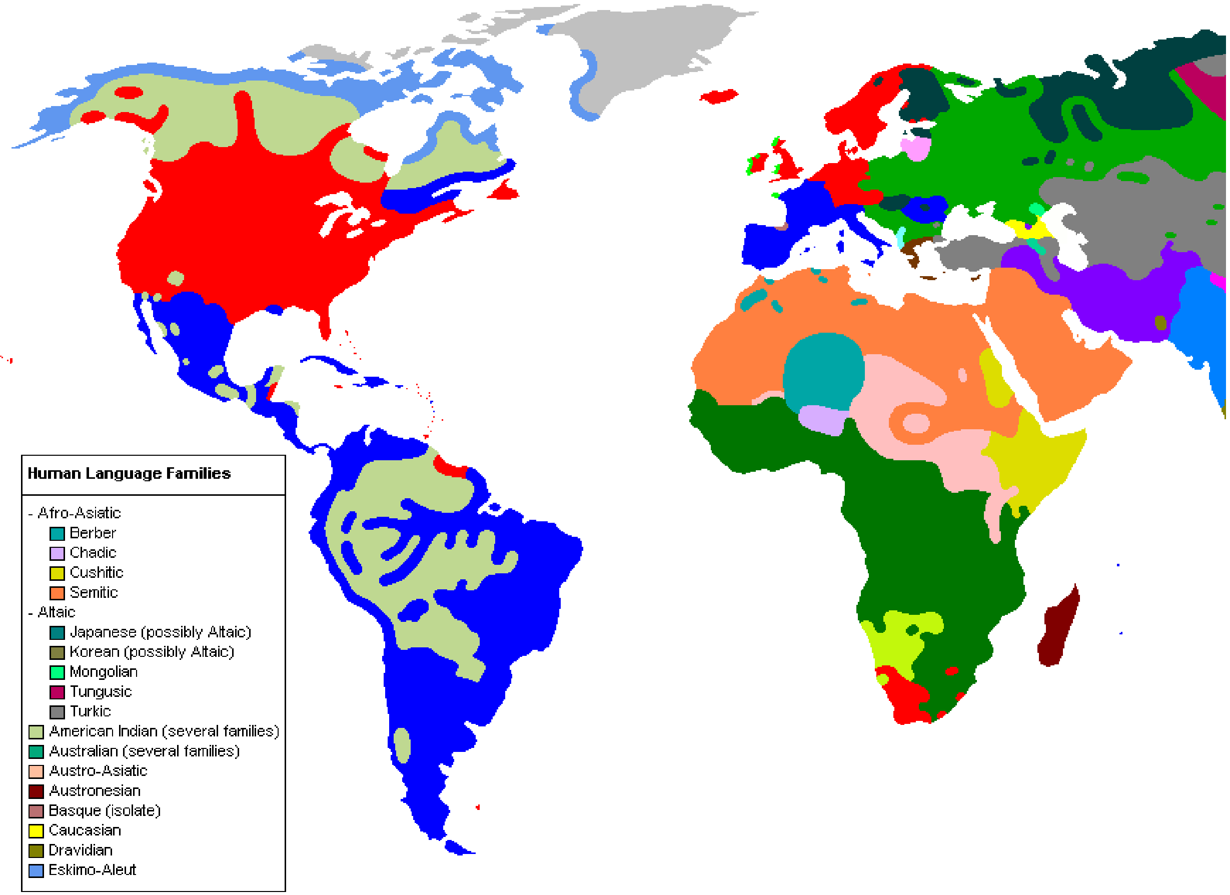
blah

blah blah

BLAH blà

b-b-blah

Languages



Mandarin (850m)
Spanish (330m)
English (330m)
Hindi (240m)
Arabic (200m)
Bengali (180m)
Portuguese (180m)
Russian (140m)
French (120m)
Japanese (120m)
Punjabi (100m)
German (100m)
Javanese (90m)
Shanghainese (90m)

- around 6000 languages
- around 20 language families
- European languages are mostly Indo-European

Counts depend a lot on the definition and may vary.

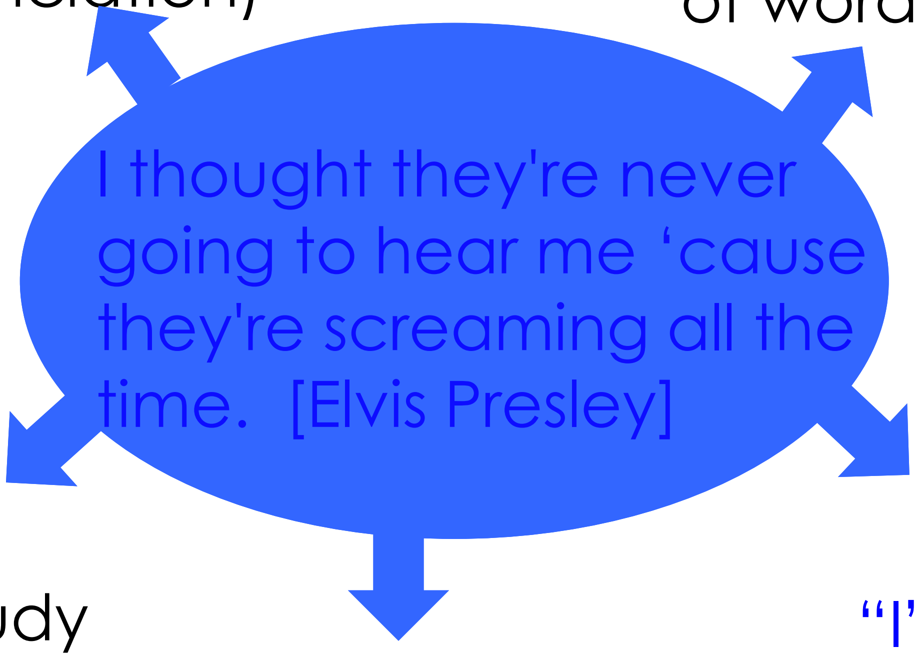
Fields of Linguistics

/ai θot…/

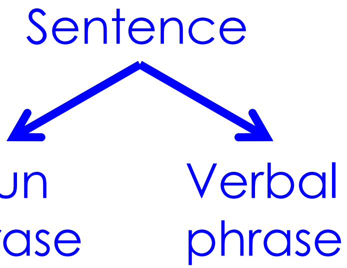
(Phonology, the study of pronunciation)

go/going

(Morphology, the study of word constituents)



I thought they're never going to hear me ‘cause they're screaming all the time. [Elvis Presley]



Sentence

Noun phrase

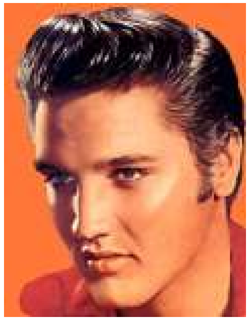
Verbal phrase

(Syntax, the study of grammar)

It doesn’t matter what I sing.

(Pragmatics, the study of language use)

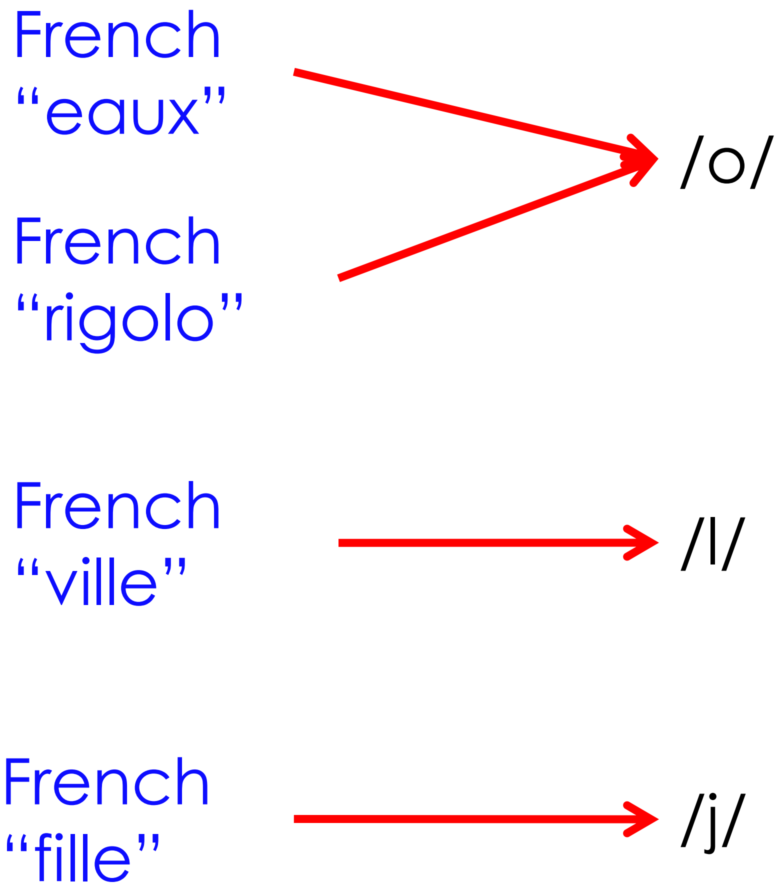
“I” =



(Semantics, the study of meaning)

Sounds of Language

Spelling and sounds do not always coincide



French "eaux"

French "rigolo"

/o/

Different letters are pronounced the same

French "ville"

/l/

French "fille"

/j/

The same letters are pronounced differently

...ough: ought, plough, cough, tough, though, through

Different Languages

Some languages have sounds that some other languages do not know

French: Nasal sounds

English: th

German: lax and tense vowels

Arab: Guttural sounds

Chinese: tones

Spanish: double rolled R

Phonology

Phonology is the study of the sounds of language.

Words of the language

eaux

rigolo

the, that

…

Sounds of the language

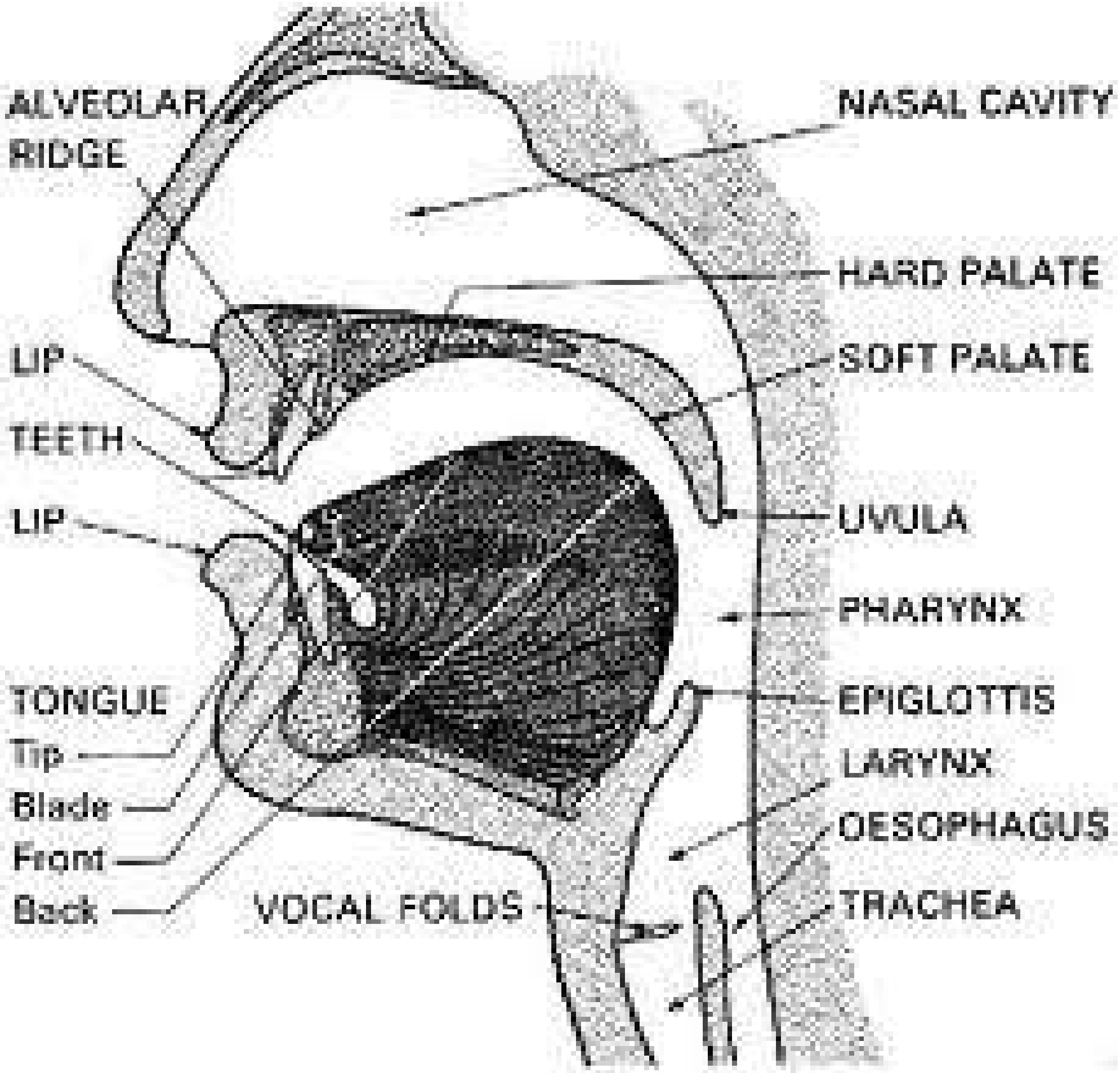
/o/

/ ə/

/j/

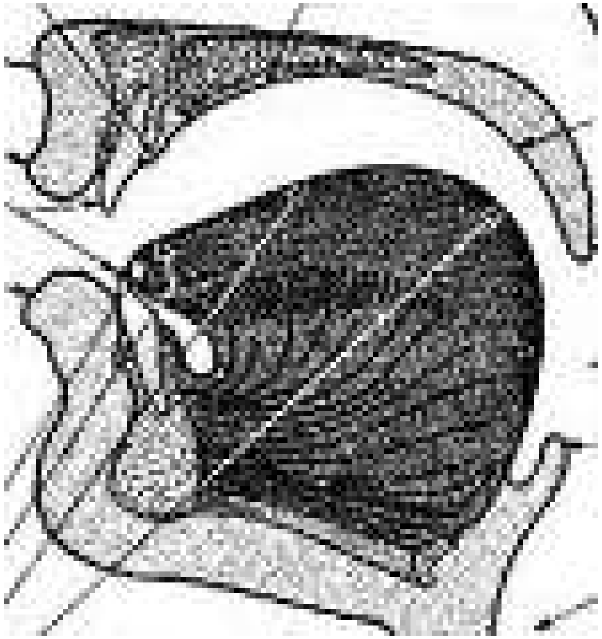
/l/

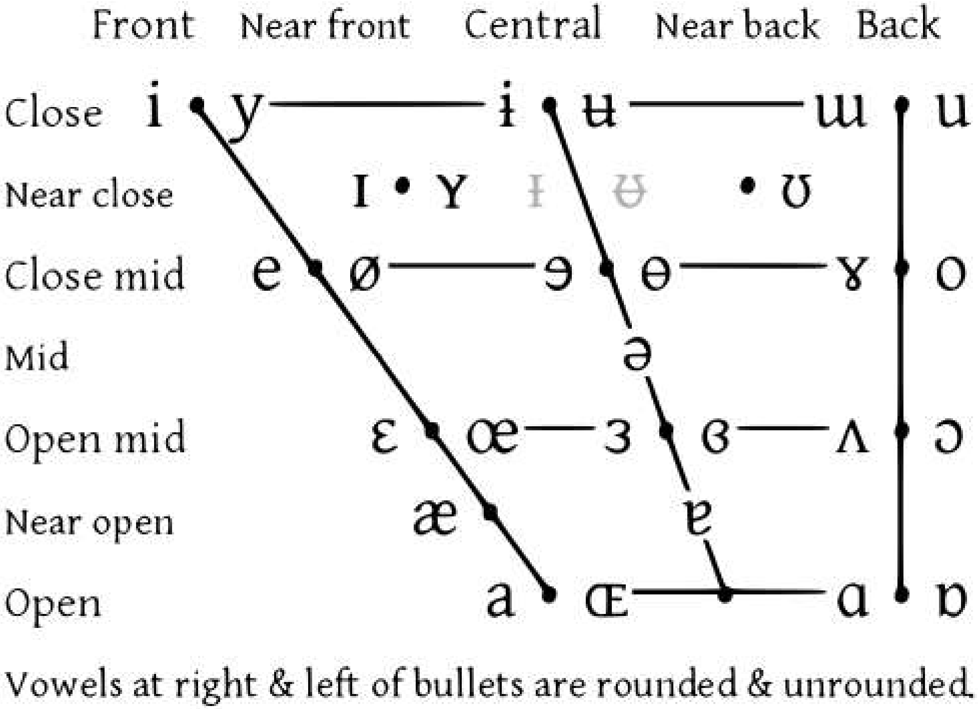
Speech Organs



IPA

The **International Phonetic Alphabet (IPA)** maps exact mouth positions (=sounds) to phonetic symbols.





Front Near front Central Near back Back

Close i • y ɨ • ʉ ɯ • u

Near close ɪ • ʏ ɪ̈ ʊ̈ • ʊ

Close mid e • ø ɘ • ɵ ɤ • o

Mid ə

Open mid ɛ • œ ɜ • ɞ ʌ • ɔ

Near open æ ɐ

Open a • ɶ ɑ • ɒ

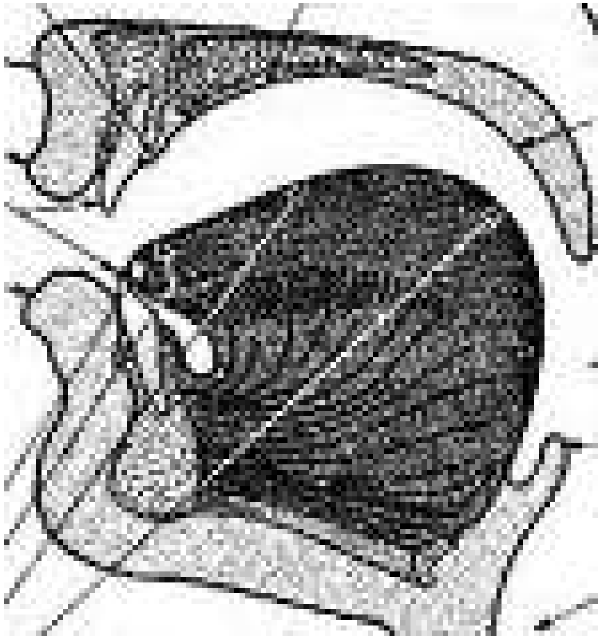
Vowels at right & left of bullets are rounded & unrounded.

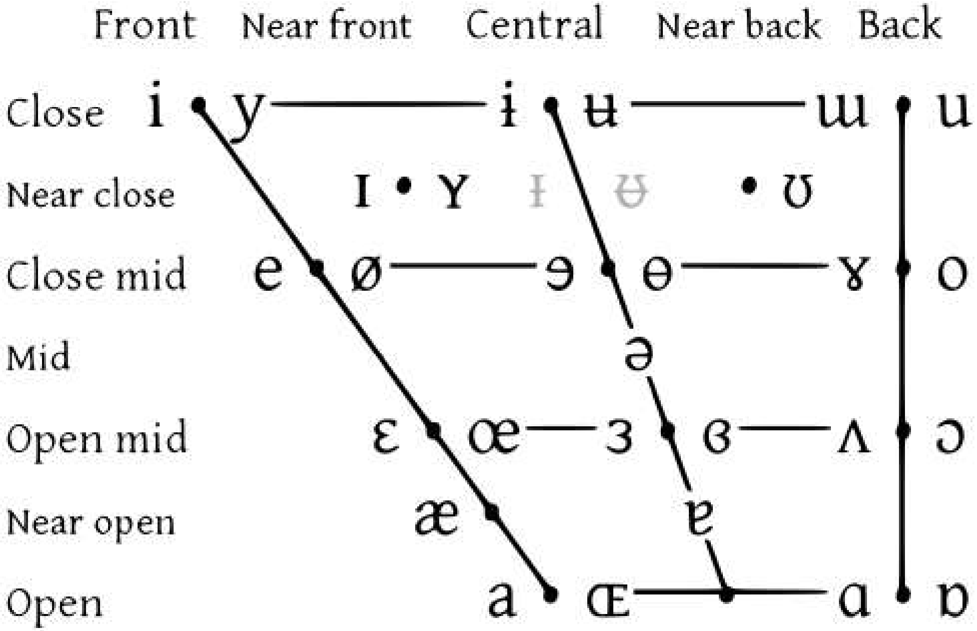
The phonetic symbols loosely correspond to latin letters

Vowels

The vowels are described by:

- the position of the tongue in the mouth (try /ø/ vs. /o/)
- the opening of the mouth (try /i/ vs. /ɑ/)
- the lip rounding (try /i/ vs. /y/)



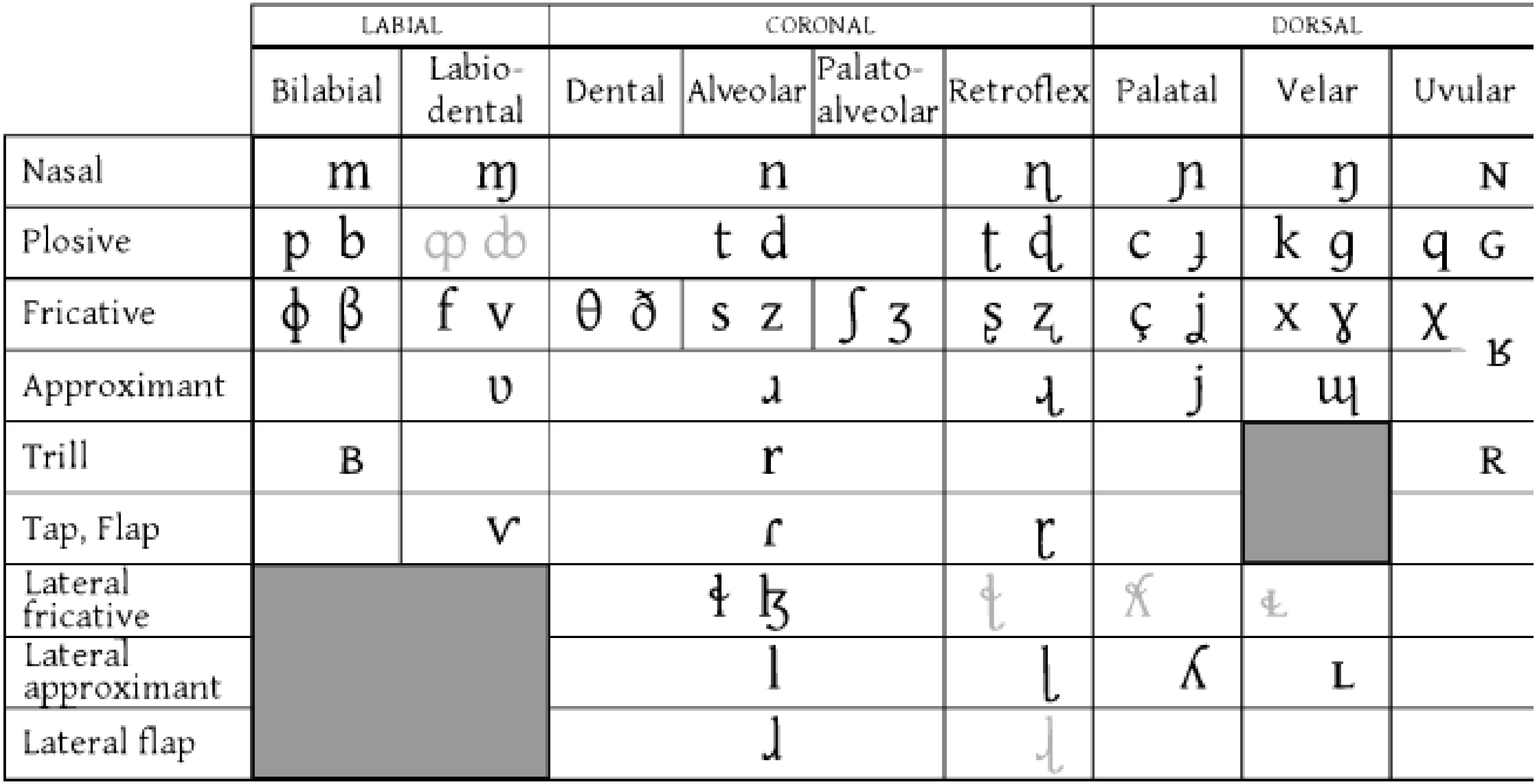


Vowels at right & left of bullets are rounded & unrounded.

Consonants

The consonants are described by:

- the place in the mouth (try /f/ vs. /s/)
- the action (try /t/ vs. /s/)



	LABIAL		CORONAL				DORSAL		
	Bilabial	Labio-dental	Dental	Alveolar	Palato-alveolar	Retroflex	Palatal	Velar	Uvular
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ
Plosive	p b	ȹ ȸ		t d		ʈ ɖ	c ɟ	k g	q ɢ
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ
Approximant		ʋ		ɹ		ɻ	j	ɰ	
Trill	ʙ			r				▨	ʀ
Tap, Flap		ⱱ		ɾ		ɽ		▨	
Lateral fricative	▨	▨		ɬ ɮ		ꞎ	𝼆	𝼄	
Lateral approximant	▨	▨		l		ɭ	ʎ	ʟ	
Lateral flap	▨	▨		ɺ		𝼈			

IPA applied

The IPA allows us to describe the pronunciation of a word precisely.

French “eau” → /o/

French “fille” → /fij/

English “mailed” → / mɛ̝͟ᶦ ɫ:d /

Heteronyms

The same spelling can be pronounced in different ways. Such words are called **heteronyms**.

I read a book every day.   / ... ri:d .../

I read a book yesterday.   / ... rɛ:d .../

Homophones

The same pronunciation can be spelled in different ways (such words are called **homophones**)

site / sight / cite

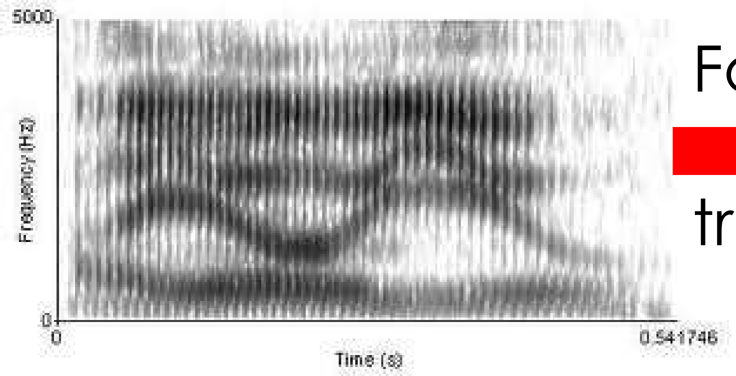
Find homophones in French!

Therefore: It is hard to wreck a nice beach

(= It is hard to recognize speech)

Speech Recognition

Speech Recognition is the process of transforming a sequence of sounds into written text.

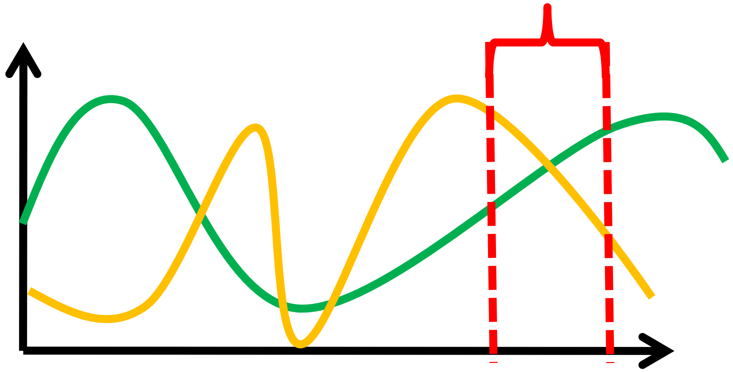


Spectrogram

Fourier

tranformation

window



Spectrogram components

Guess the sound of the window, based on

- what sound such a window was during training
- what sound is likely to follow the previous one

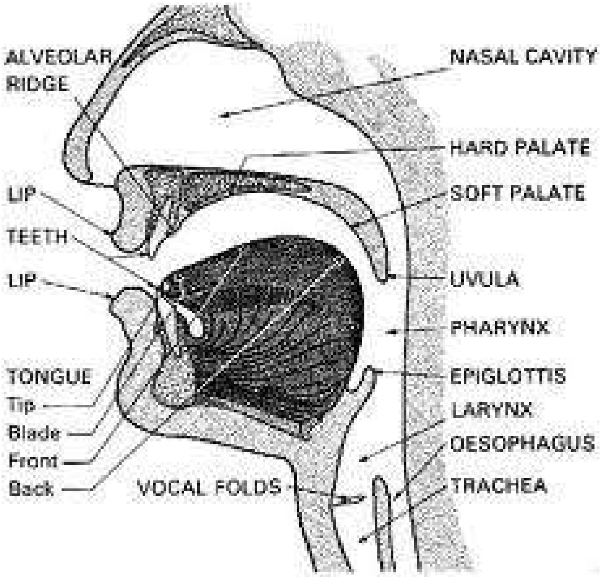
/o/

Phonology Summary

Phonology is the study of the sounds of language

Letters in words and their sounds do not always correspond.

The International Phonetic Alphabet can be used to describe the speech sounds



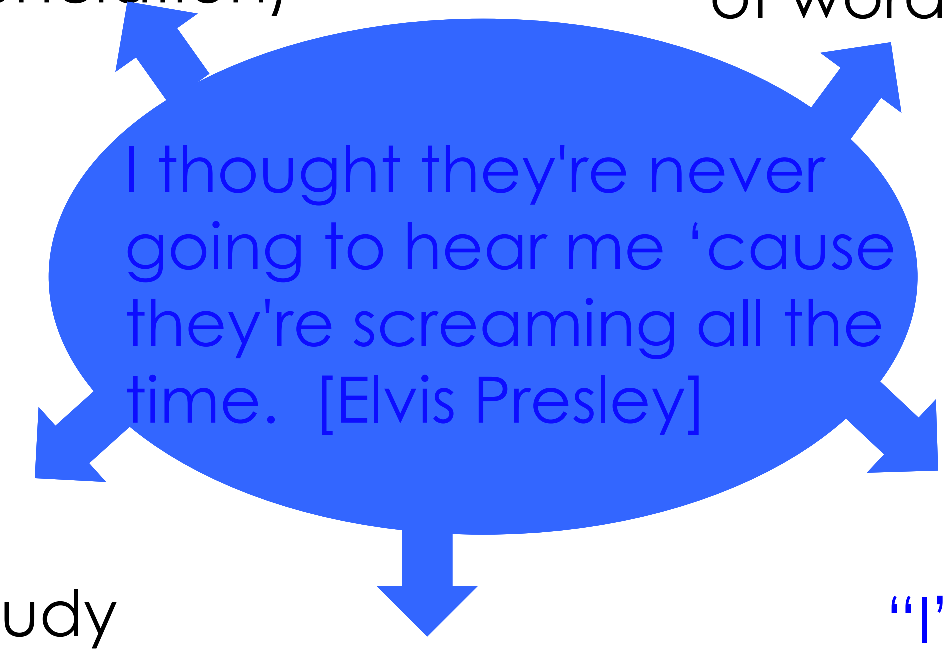
Fields of Linguistics

/ai θot…/ ✓

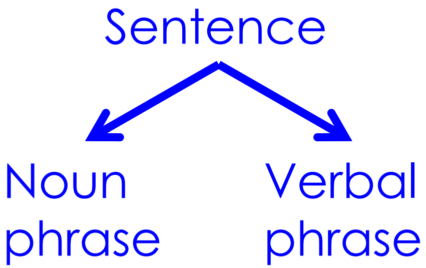
(Phonology, the study of pronunciation)

go/going

(Morphology, the study of word constituents)



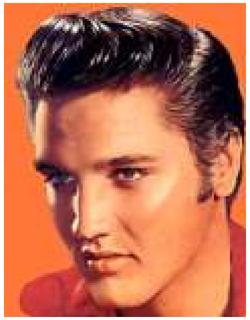
I thought they're never going to hear me ‘cause they're screaming all the time. [Elvis Presley]



(Syntax, the study of grammar)

It doesn’t matter what I sing.

(Pragmatics, the study of language use)

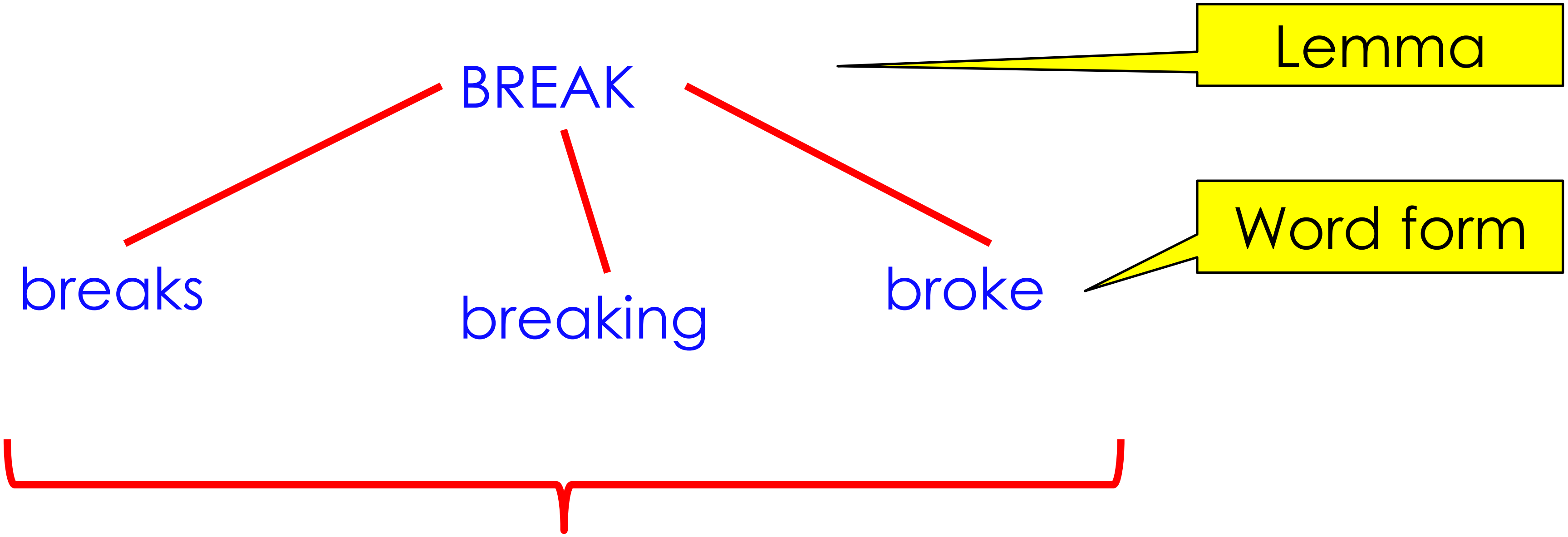
“I” = 

(Semantics, the study of meaning)

Lexemes

A **lexeme/lemma** is the base form of a word.

He is breaking the window.
He broke the window before.



BREAK

breaks

breaking

broke

Lemma

Word form

Inflection
(the phenomenon that one lemma has different word forms)

Inflectional Categories (Nouns)

The following properties influence the inflection of nouns:

- gender: masculine, feminine, neuter, ...

only vaguely related to natural gender

le garçon, la fille   das Auto

- number: singular, plural, dual, trial, ...

child, children

in Arabic

in Tolomako

- case: nominative, accusative, dative, ablative...

das Auto, des Autos,…

Only some of the 8 indo-european cases survived

- class: animate, dangerous, edible, …

In Dyirbal

the man’s face / ~~the face of the man~~

Inflectional Categories

The following properties influence the inflection of verbs:

- person: 1st, 2nd, honorifics...

  I, you, he, vous, san, chan,…

  Japanese honorifics conjugate the verb

- number: singular, plural, ...

  I/we, she/they, …

- tense: past, future, ...

  go, went, will go

  Others: “later today”, “past, but not earlier than yesterday”

- aspect, aktionsart: state, process, perfect, ...

  Peter is running / ~~Peter is knowing Latin~~

- modus: indicative, imperative, conjunctive, ...

  Peter runs / Run, Peter!

Morphemes

A **morpheme** is a word constituent that carried meaning.

prefix suffix affixes

un-break-able

- "un" is a morpheme that indicates negation
- "break" is the root morpheme
- "able" is a morpheme that indicates being capable of something

unbreakability, unbreakably, The Unbreakables…

Morpheme "s" indicates plural

Morphology is not trivial

- Morphemes do not always simply add up

  happy + ness ≠ happyness (but: happiness)
  dish + s ≠ dishs (but: dishes)
  un + correct ✗ in +correct = incorrect

- Example: plural and singular

  boy + s -> boys (easy)

  city + s -> cities

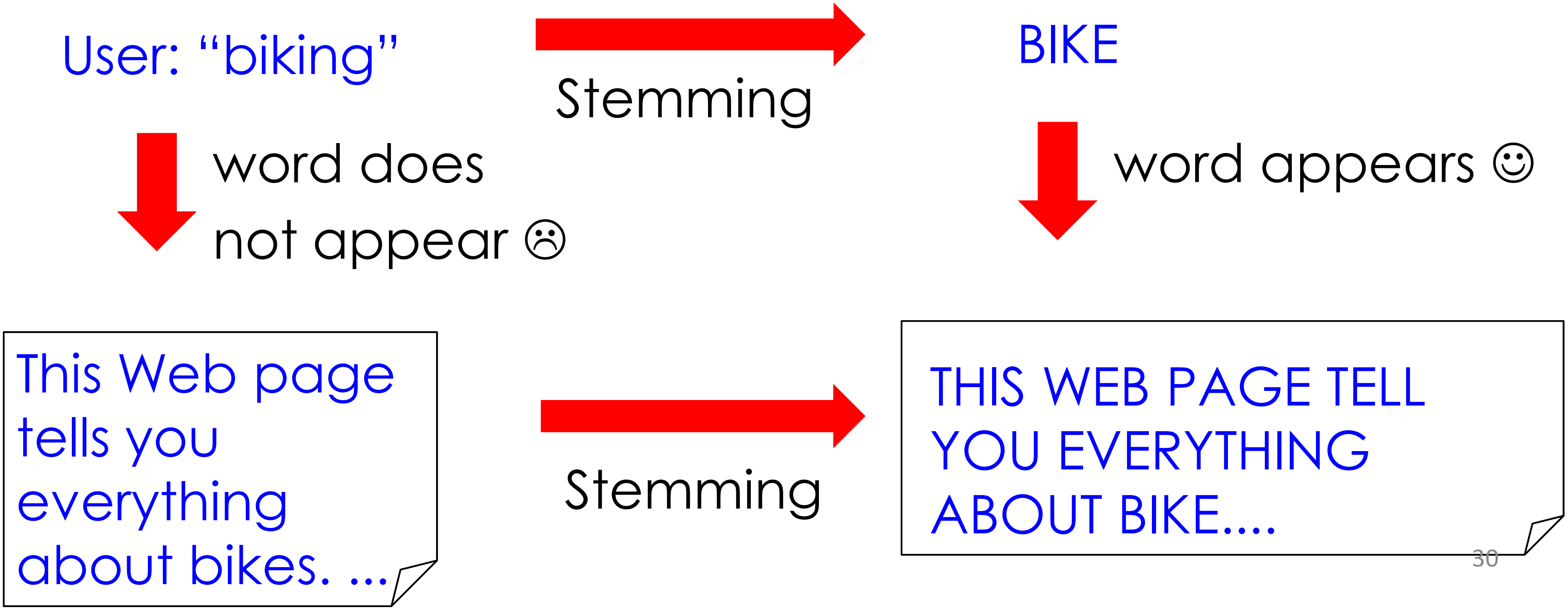
  atlas -> atlas, bacterium -> bacteria,
  automaton -> automata, mouse -> mice,
  person -> people,
  physics -> (no pl)

Stemming

Stemming is the process of mapping different related words onto one word form.

bike, biking, bikes, racebike → BIKE

Stemming is allows search engines to find related words:



User: “biking”

Stemming

BIKE

word does not appear ☹

word appears ☺

This Web page tells you everything about bikes. …

Stemming

THIS WEB PAGE TELL YOU EVERYTHING ABOUT BIKE….

Stemming to Singular

Stemming be done at different levels of aggressiveness:

- Just mapping plural forms to singular

words → word

Stemming is the process of mapping different related words onto one word form.

⬇

Stemming is the process of mapping different related word onto one word form.

Still not trivial:

universities → university
emus → emu, but genus → genus
mechanics → mechanic (guy) or mechanics (the science)

Stemming to the Lemma

- Reduction to the lemma, i.e., the non-inflected form

  mapping → map, stemming → stem, is → be,
  related → relate

  Stemming is the process of mapping
  different related words onto one word form.

  ⬇

  Stem be the process of map
  different relate word onto one word form.

Still not trivial:

interrupted, interrupts, interrupt → interrupt
ran → run

Stemming to the Stem

- Reduction to the stem, i.e., the common core of all related words

different → differ (because of "to differ")

Stemming is the process of mapping
different related words onto one word form.

⬇

Stem be the process of map
differ relate word onto one word form.

May be too strong:

interrupt, rupture, disrupt → rupt

Brute Force Stemming

The **brute force / dictionary-based** stemming method uses a list of all word forms with their lexemes.

break, broke, breaks, breakable → BREAK

computer, computable, computers → COMPUTE

My computer broke down.

⬇

MY COMPUTE BREAK DOWN.

Rule-based Stemming

Rule-based stemming uses IF-THEN rules to stem a word.

- IF the word ends with “s”, THEN cut “s”

breaks → break

- IF the word ends with “lves”, THEN replace “ves” by “f”

loves → love
calves → calf

- IF the word ends with “ing” and has a vowel in the stem, THEN cut the “ing”

thinking → think
thing → thing

(e.g., the Porter Stemmer for reduction to the stem)

Stochastic Stemming

Stochastic Stemming learns how to find the lemma from examples.

Examples:
computer, computers → COMPUTER
hit, hits → HIT
box, boxes → BOX

Learned rules:

- Cut off the "s".
- If the word ends in "x", also cut off the "e"

foxes → ~~foxe~~ / fox

Morphology Summary

Words can consist of constituents that carry meaning (morphemes)

In English, morphemes combine in very productive and non-trivial ways.

Stemming is the process of removing supplemantary morphemes

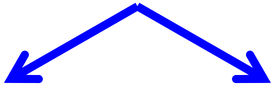
Fields of Linguistics

/ai θot.../ ✓
(Phonology, the study of pronunciation)

go/going ✓
(Morphology, the study of word constituents)

I thought they're never going to hear me ‘cause they're screaming all the time. [Elvis Presley]

Sentence



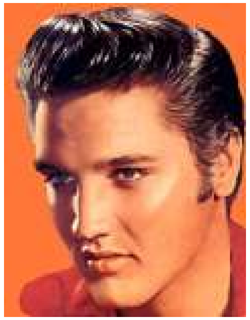
Noun phrase

Verbal phrase

(Syntax, the study of grammar)

It doesn’t matter what I sing.
(Pragmatics, the study of language use)

“I” =



(Semantics, the study of meaning)

Information Extraction

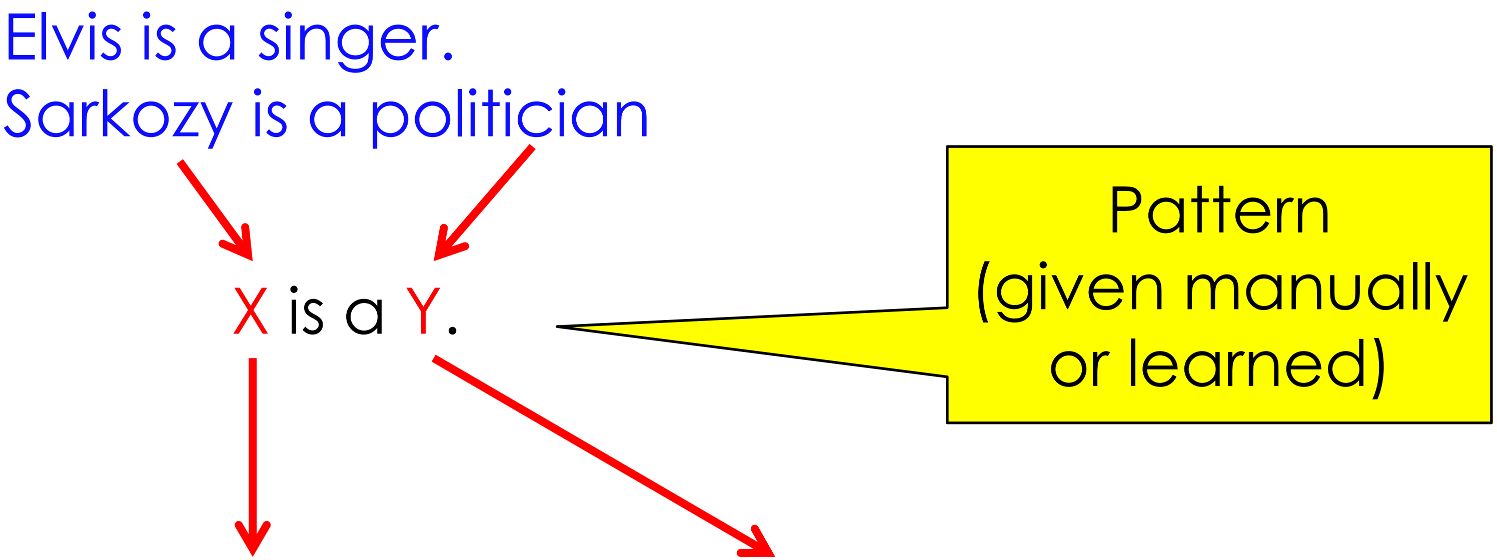
Information Extraction is the process of extracting structured information (a table) from natural language text.

Elvis is a singer.
Sarkozy is a politician

Person	Profession
Elvis	singer
Sarkozy	politician

Pattern Matching

Information Extraction can work by **pattern matching**.



Elvis is a singer.
Sarkozy is a politician

X is a Y.

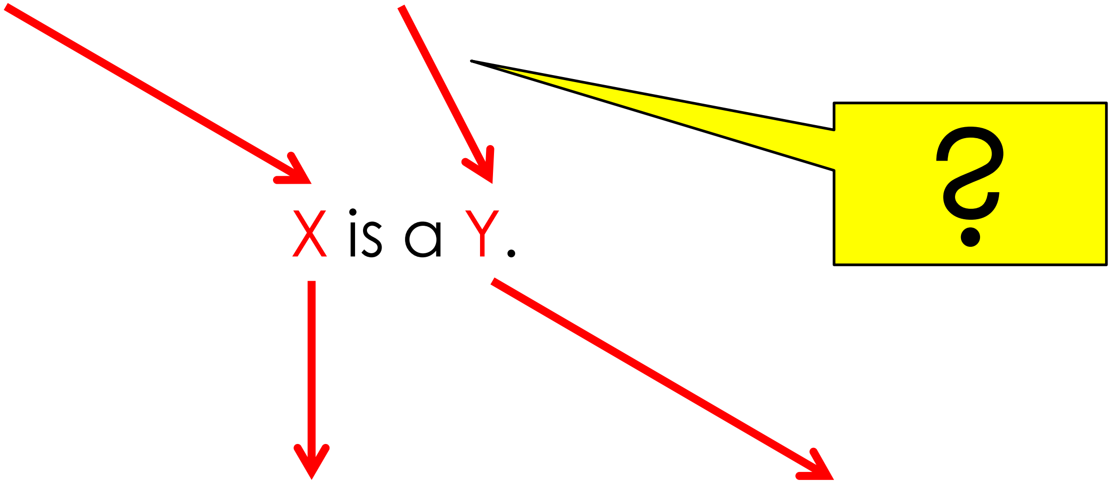
Pattern
(given manually
or learned)

Person	Profession
Elvis	singer
Sarkozy	politician

Pattern Matching Problems

Information Extraction can work by **pattern matching**.

Elvis is a wonderful rock singer and always there for me.

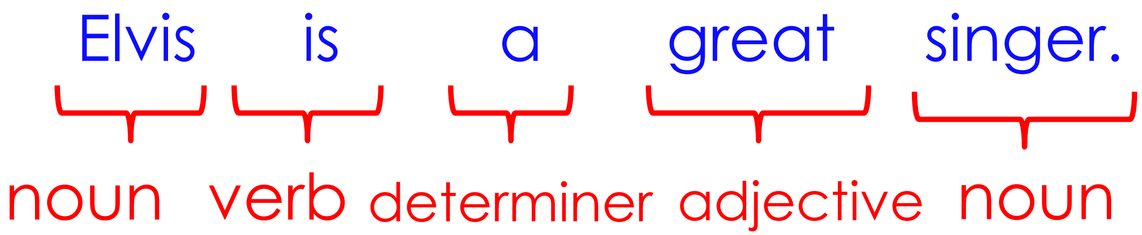


X is a Y.

Person	Profession
Elvis	wonderful
Elvis	rock
Elvis	singer and

Part of Speech

The **Part-of-Speech** (POS) of a word in a sentence is the grammatical role that this word takes.



Elvis is a great singer.

noun verb determiner adjective noun

Open POS Classes

The **Part-of-Speech** (POS) of a word in a sentence is the grammatical role that this word takes.

Open POS classes:

- Proper nouns: Alice, Fabian, Elvis, ...
- Nouns: computer, weekend, ...
- Adjectives: fantastic, self-reloading, ...
- Verbs: adore, download, ...

Elvis loves Priscilla.
Priscilla loves her fantastic self-reloading fridge.
The mouse chases the cat.

Closed POS Classes

Closed POS classes:

- Pronouns: he, she, it, this, ...

  (≈ what can replace a noun)
- Determiners: the, a, these, your, my, ...

  (≈ what goes before a noun)
- Prepositions: in, with, on, ...

  (≈ what goes before determiner + noun)
- Subordinators: who, whose, that, which, because, ...

  (≈ what introduces a sub-ordinate sentence)

This is his car.
DSK spends time in New York.
Elvis, who is thought to be dead, lives on the moon.

Exercise

POS classes:

- Proper nouns: Alice, Fabian, Elvis, ...
- Nouns: computer, weekend, ...
- Adjectives: fantastic, self-reloading, ...
- Verbs: adore, download, ...
- Pronouns: he, she, it, this, ... (≈ what can replace a noun)
- Determiners: the, a, these, your, my, ... (≈ what goes before a noun)
- Prepositions: in, with, on, ... (≈ what goes before determiner + noun)
- Subordinators: who, whose, that, which, because, ...

  (≈ what introduces a sub-ordinate sentence)

Determine the POS classes of the words in these sentences:

- *Carla Bruni works as a chamber maid in New York.*
- *Sarkozy loves Elvis, because his lyrics are simple.*
- *Elvis, whose guitar was sold, hides in Tibet.*

POS Tagging

POS tagging is the process of, given a sentence, determining the part of speech of each word.

Elvis is a great rock star who is adored by everybody.

↓

Elvis/ProperNoun is/Verb a/Det great/Adj rock/Noun star/Noun who/Sub is/Verb adored/Verb …

POS Tagging Difficulties

POS Tagging is not simple, because

- Some words belong to two word classes

  He is on the run/Noun.

  They run/Verb home.

- Some word forms are ambiguous:

  Sound sounds sound sound.

How can we POS tag a sentence efficiently?

Hidden Markov Model

A **Hidden Markov Model (HMM)** is a tuple of

- a set of states S

S = { Noun, Verb, … }

- transition probabilities

trans: S x S → [0,1]

$\sum_x$ trans(S,x) = 1

trans(Noun, Verb)= 0.7

trans(Noun, Det) =0.1

…

- a set of observations O

O = {run, the, on, Elvis, …}

- emission probabilities

em: S x O → [0,1]

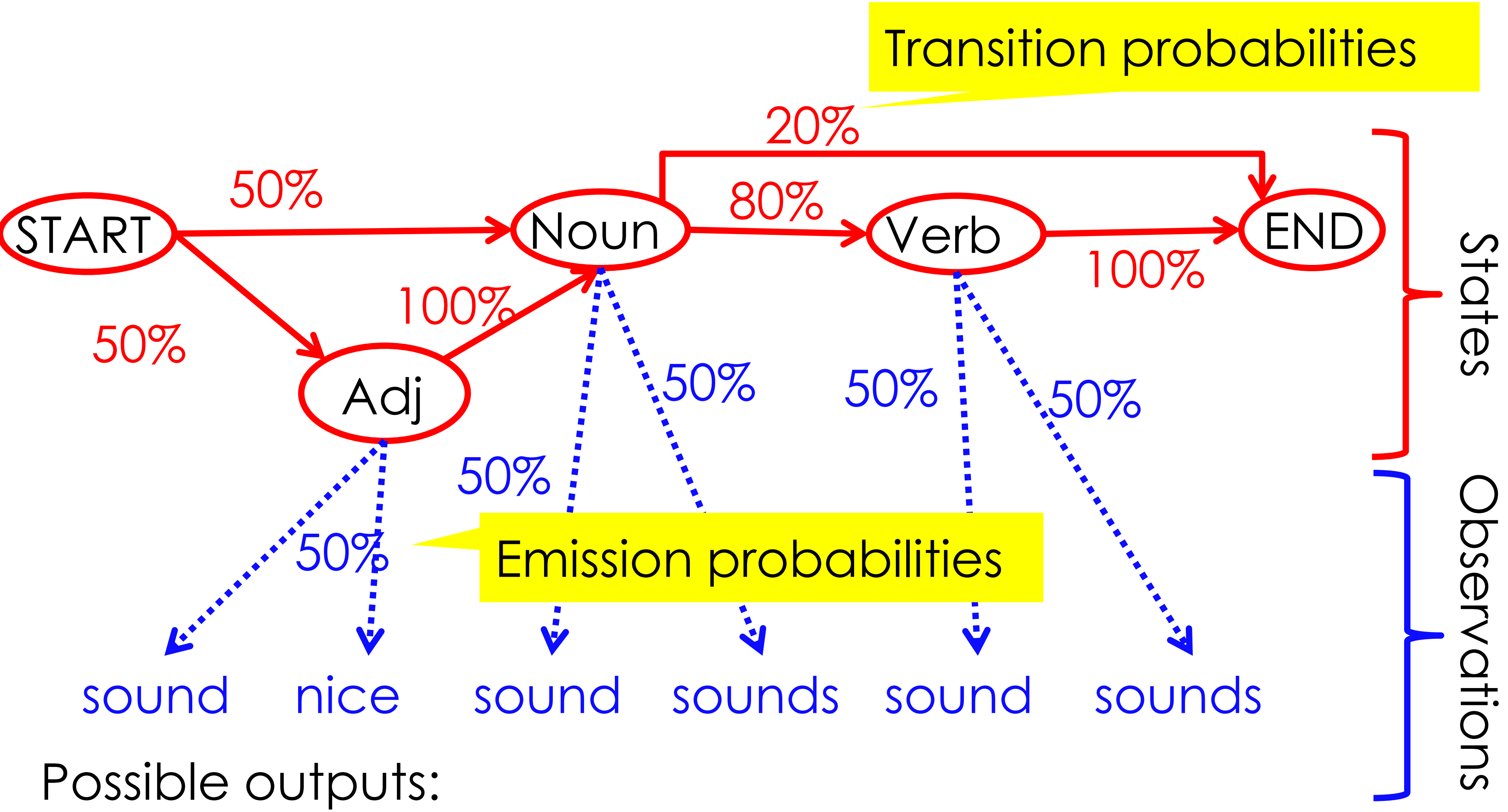
$\sum_x$ em(S,x) = 1

em(Noun,run) = 0.000001

em(Noun,house) = 0.00054

…

HMM Example 1



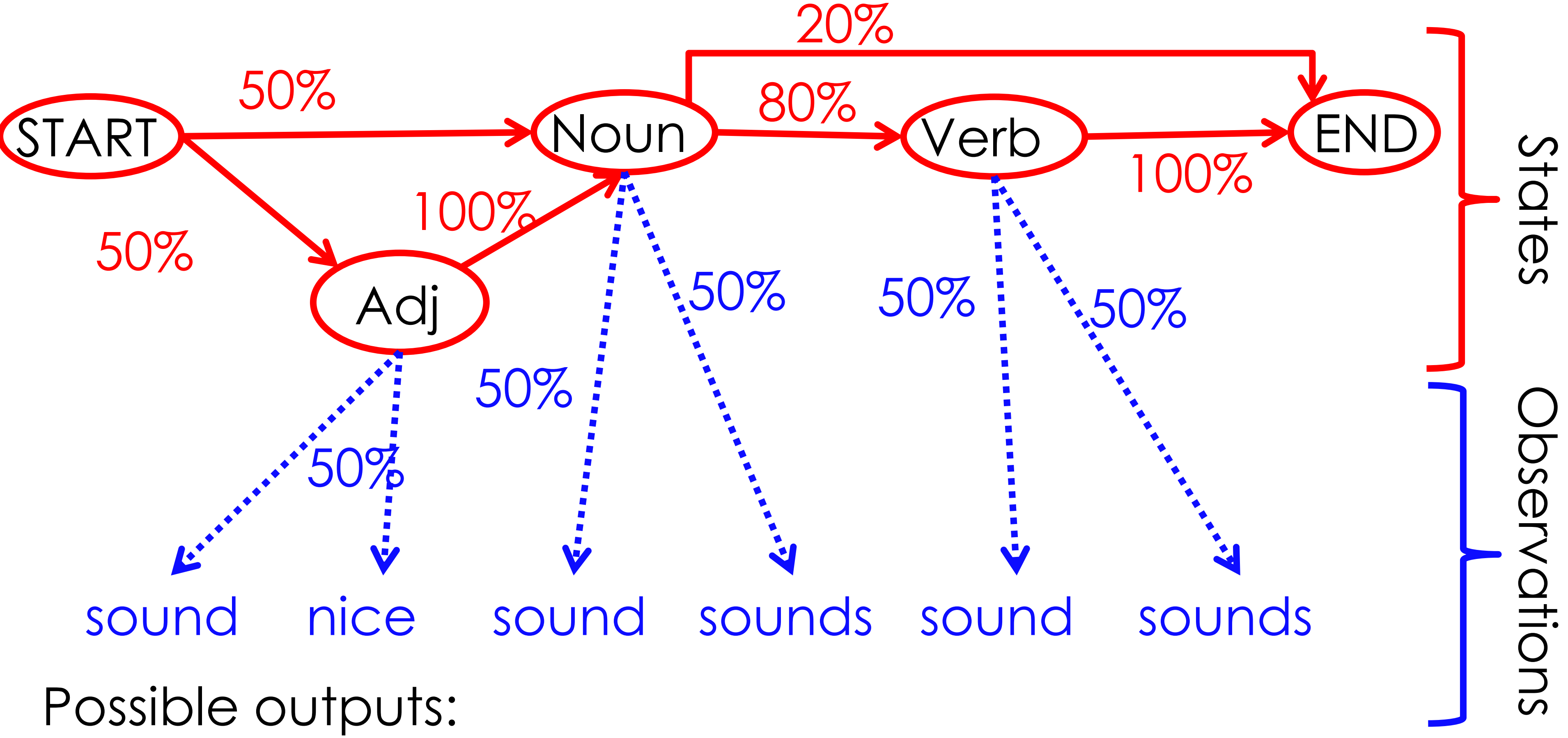
Possible outputs:

Sentence: “nice sounds!”

Sequence: Adj+Noun

Probability: 50%*50%*100%*50%*20% = 2.5%

HMM Example 2



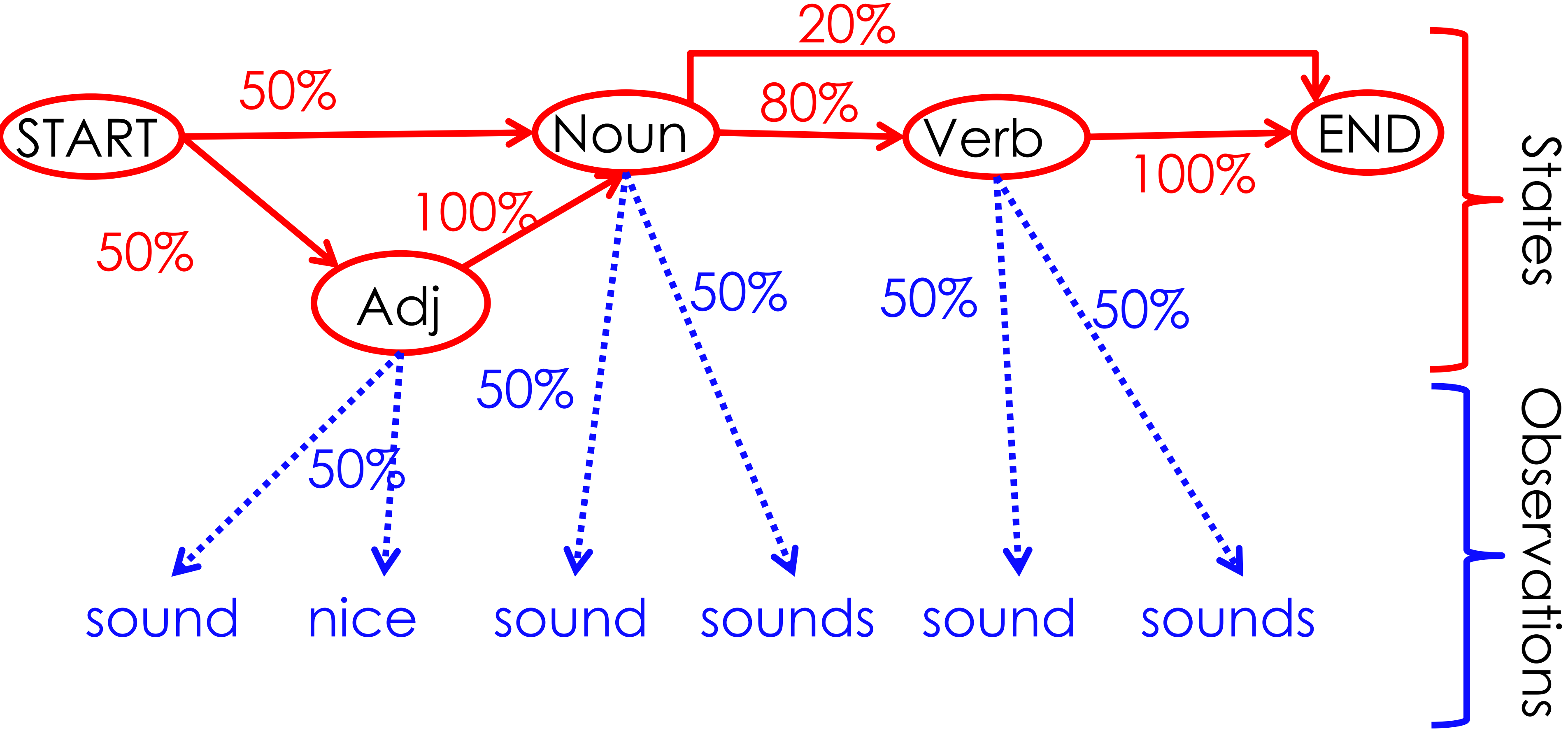
Possible outputs:

Sentence: "sound sounds sound"

Sequence: Adj+Noun+Verb

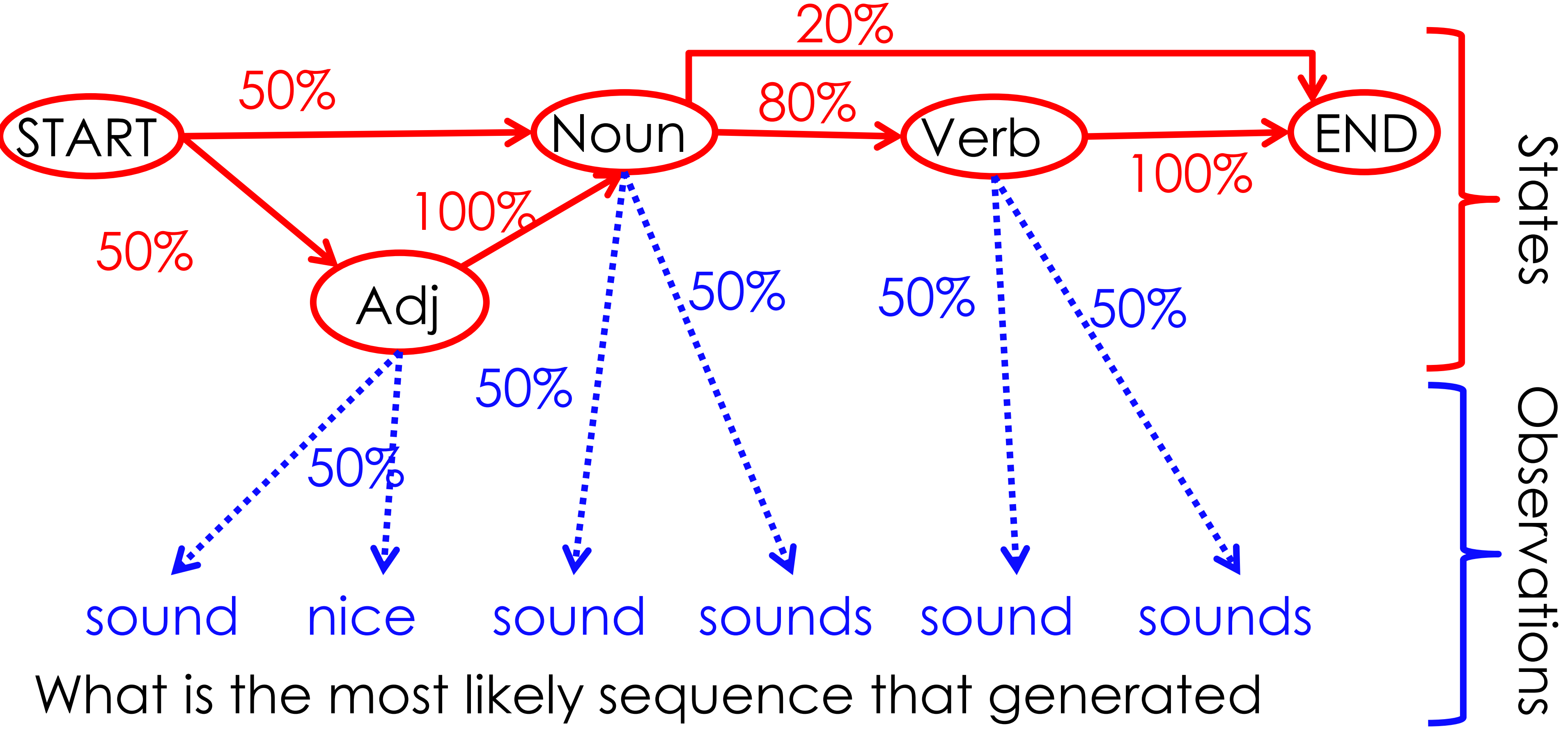
Probability: 50%*50%*100%*50%*80% *50% = 5%

HMM Exercise



Generate one output with its probability!

HMM Question



What is the most likely sequence that generated "Sound sounds"?

Adj + Noun (50%*50%*100%*50%*20% =2.5%)

Noun + Verb (50%*50%*80%*50% =10%)

POS Tagging = HMM

What is the most likely sequence that generated "Sound sounds"?

Adj + Noun (50%*50%*100%*50%*20% =2.5%)
Noun + Verb (50%*50%*80%*50% =10%)

Finding the most likely sequence of tags that generated a sentence **is** POS tagging (hooray!).

The task is thus to try out all possible paths in the HMM and compute the probability that they generate the sentence we want to tag.

Viterbi-Algorithm: Init

The **Viterbi Algorithm** is an efficient algorithm that, given an HMM and a sequence of observations, computes the most likely sequence of states.

	START	Adj	Noun	Verb	END
.	1	0	0	0	0
Sound	0				
sounds	0				

States

What is the probability that "." was generated by START?

Sentence (read top down)

Initial hard-coded values

Viterbi-Algorithm: Step

The **Viterbi Algorithm** is an efficient algorithm that, given an HMM and a sequence of observations, computes the most likely sequence of states.

	START	Adj	Noun	Verb	END
.	100%	0	0	0	0
Sound	0	○			
sounds	0				

What is the probability that "sound" is an adjective?

This depends on 3 things:

- The emission probability em(Adj,sound)
- The transition probability trans(*previousTag*,Adj)
- The probability that we guessed the *previousTag* right cell(*previousTag*, *previousWord*)

Viterbi-Algorithm: Step

The **Viterbi Algorithm** is an efficient algorithm that, given an HMM and a sequence of observations, computes the most likely sequence of states.

	START	Adj	Noun	Verb	END
.	100%	0	0	0	0
Sound	0	○			
sounds	0				

What is the probability that "sound" is an adjective?

Find *previousTag* that maximizes

em(Adj,sound)
* trans(*previousTag*,Adj)
* cell(*previousTag, previousWord*)

…then write this value into the cell, + a link to *previousTag*

Viterbi-Algorithm: Step

The **Viterbi Algorithm** is an efficient algorithm that, given an HMM and a sequence of observations, computes the most likely sequence of states.

	START	Adj	Noun	Verb	END
.	100%	0	0	0	0
Sound	0	○			
sounds	0				

What is the probability that "sound" is an adjective?

previousTag = START

50%	em(Adj,sound)
50%	* trans(*previousTag*,Adj)
100%	* cell(*previousTag, previousWord*)

…then write this value into the cell, + a link to *previousTag*

Viterbi-Algorithm: Iterate

The **Viterbi Algorithm** is an efficient algorithm that, given an HMM and a sequence of observations, computes the most likely sequence of states.

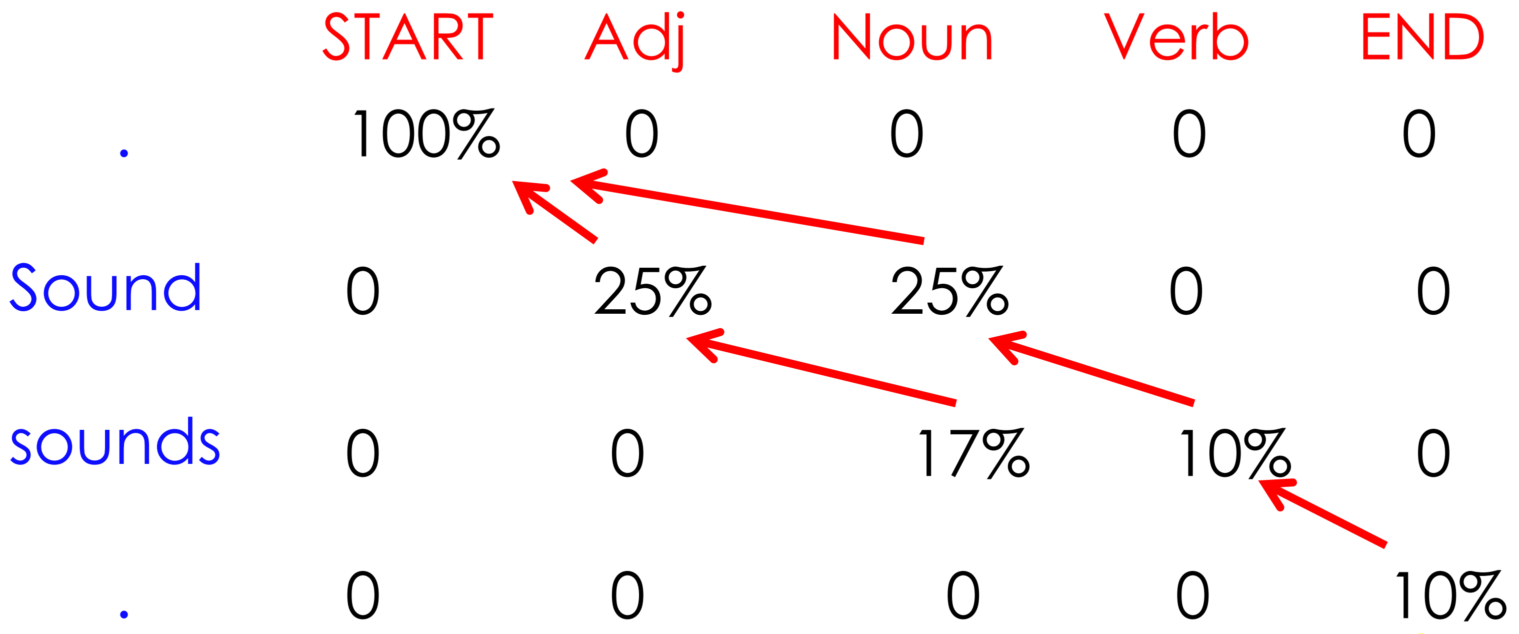
	START	Adj	Noun	Verb	END
.	100%	0	0	0	0
Sound	0	25%			
sounds	0				

This is the probability that "sound" is an adjective, with link to previous tag

Continue filling the cells in this way until the table is full

Viterbi-Algorithm: Result

The **Viterbi Algorithm** is an efficient algorithm that, given an HMM and a sequence of observations, computes the most likely sequence of states.



	START	Adj	Noun	Verb	END
.	100%	0	0	0	0
Sound	0	25%	25%	0	0
sounds	0	0	17%	10%	0
.	0	0	0	0	10%

Most likely sequence and probability can be read out backwards from here.

HMM from Corpus

The HMM can be derived from a hand-tagged corpus:

Blah blah Sarokzy/ProperNoun laughs/Verb blah.
Blub blub Elvis/ProperNoun ./STOP
Blub blub Elvis/ProperNoun loves/Verb blah.

=> em(ProperNoun,Sarkozy) = 1/3
em(ProperNoun,Elvis) = 2/3

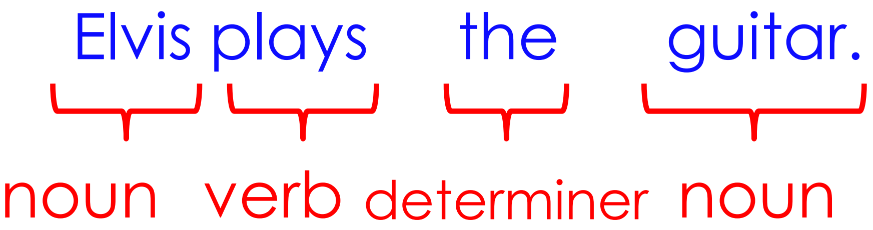
=> trans(ProperNoun,Verb) = 2/3
trans(ProperNoun,STOP) = 1/3

S = all POS tags that appear

O = all words that appear

POS Tagging Summary

The **Part-of-Speech** (POS) of a word in a sentence is the grammatical role that this word takes.



Elvis plays the guitar.

noun verb determiner noun

POS tagging can be seen as a **Hidden Markov Model.**

The **Viterbi Algorithm** is an efficient algorithm to compute the most likely sequence of states in an HMM.

The HMM can be extracted from a corpus that has been POS-tagged manually.

Stop words

Words of the closed word classes are often perceived as contributing less to the meaning of a sentence.

Words closed word classes often perceived
contributing less meaning sentence.

Stop words

Therefore, the words of closed POS-classes (and some others) are often ignored in Web search. Such words are called **stop words**.

a, the, in, those, could, can, not, ...

Ignoring stop words may not always be reasonable

"Vacation outside Europe"

↓

"Vacation Europe" →

Practical Exercise

… on Part-Of-Speech Tagging.

http://suchanek.name/work/teaching/nlp2011a_lab.html

- You have 2 sessions with 1.5 hours each. It is suggested to do exercises 1 and 2 in the first session and 3 in the second session

- The results of each exercise have to be explained in person to the instructor during the session. In addition, the results have to be handed in by e-mail to the instructor.

- This presentation will yield a PASS/NO-PASS grade for each exercise and each student

Correct Sentences

Bob stole the cat. ✔

Cat the Bob stole. ✗

Bob, who likes Alice, stole the cat.

Bob, who likes Alice, who hates Carl, stole the cat.

Bob, who likes Alice, who hates Carl, who owns the cat, stole the cat.

⇒ There are infinitively many correct sentences,

...yet not all sentences are correct.

Grammars

Bob stole the cat. ✔

Cat the Bob stole. ✗

Grammar: A formalism that decides whether a sentence is syntactically correct.

Example: Bob eats

Sentence -> Noun Verb

Noun -> Bob

Verb -> eats

Phrase Structure Grammars

Non-terminal symbols: abstract phrase constituent names,
such as “sentence”, “noun”, “verb” (in blue)

Terminal symbols: words of the language,
such as “Bob”, “eats”, “drinks”

Given two disjoint sets of symbols, N and T,
a (context-free) **grammar** is a relation between
N and strings over $N \cup T$: $G \subset N \times (N \cup T)^*$

N = {Sentence, Noun, Verb}
T = {Bob, eats}

Sentence -> Noun Verb
Noun -> Bob
Verb -> eats

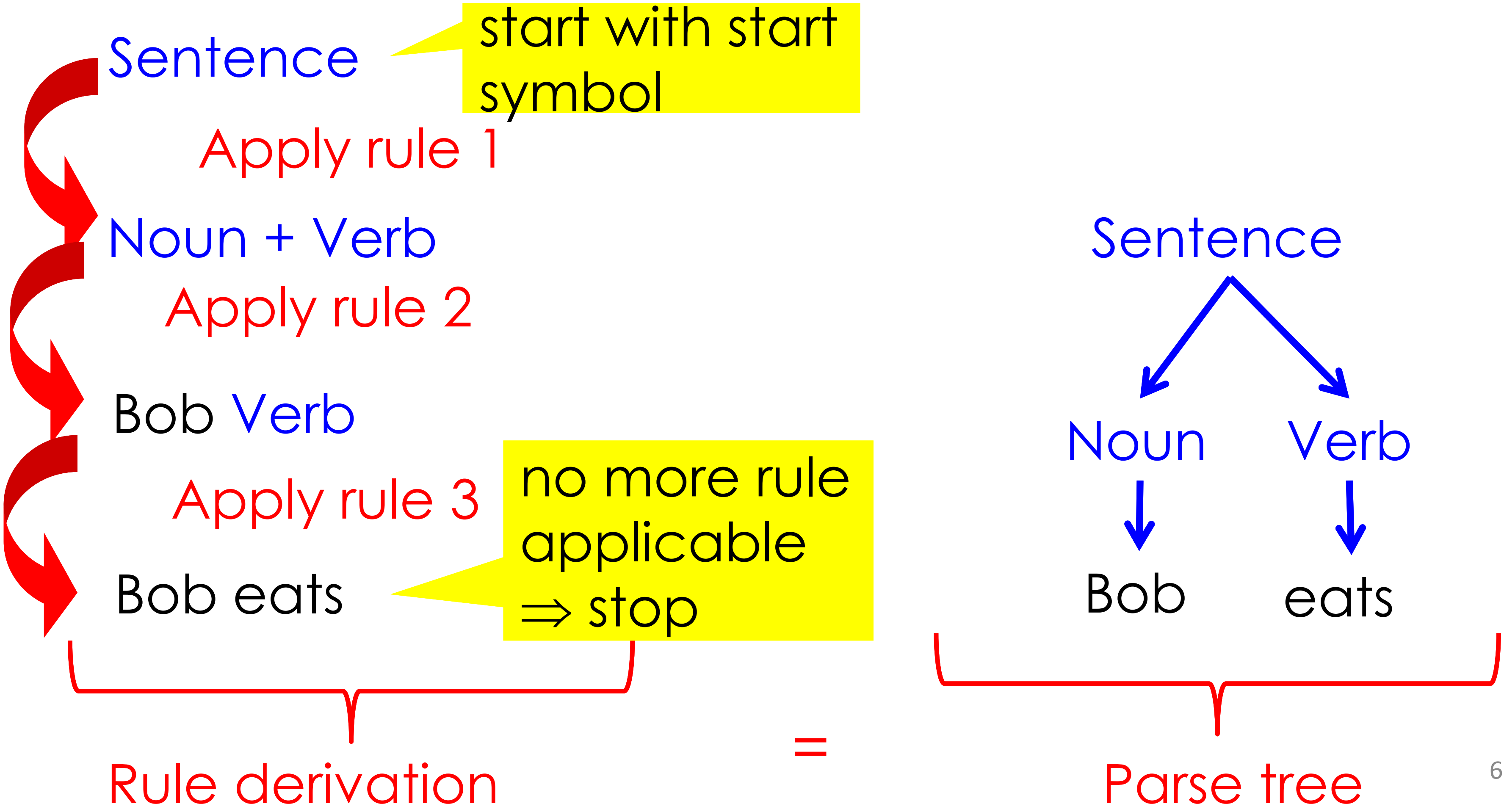
Production rules

Using Grammars

1. Sentence -> Noun Verb
2. Noun -> Bob
3. Verb -> eats

N = {Sentence, Noun, Verb}

T = {Bob, eats}



Sentence

start with start symbol

Apply rule 1

Noun + Verb

Apply rule 2

Bob Verb

Apply rule 3

Bob eats

no more rule applicable ⇒ stop

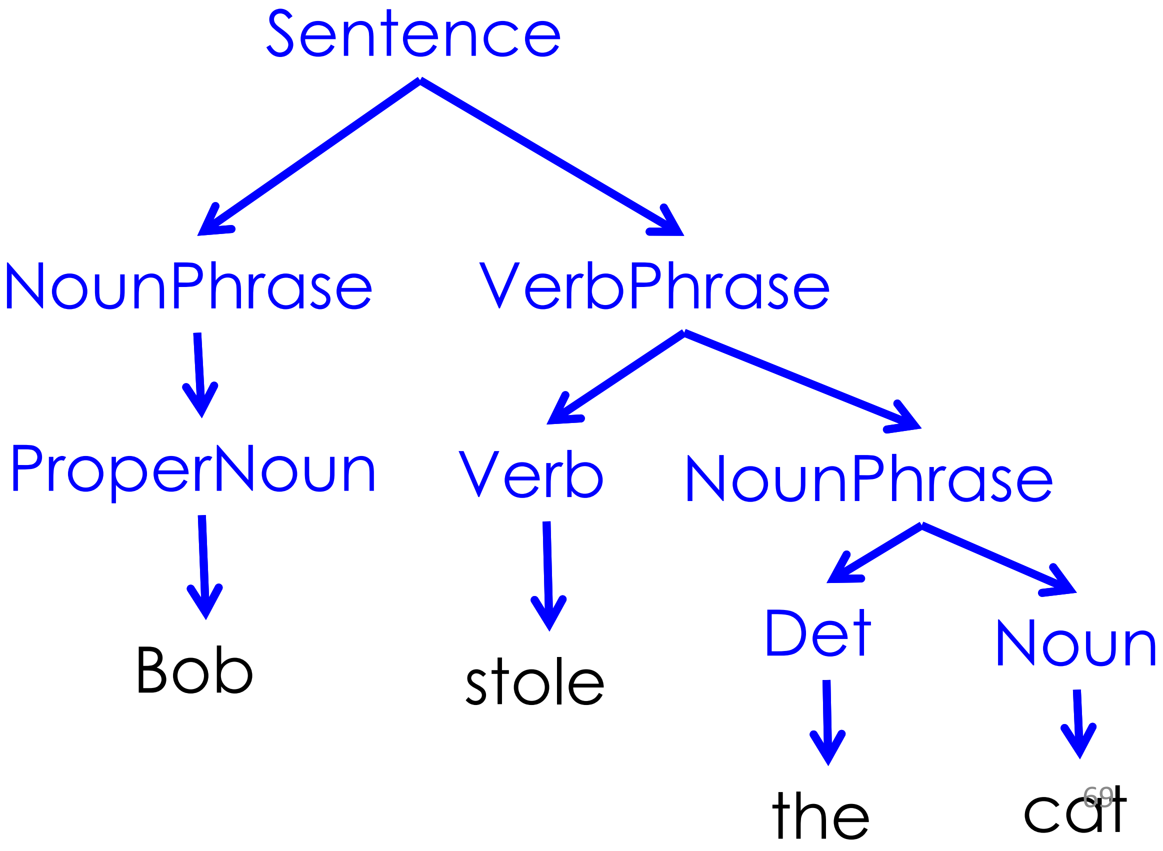
Rule derivation

=

Parse tree

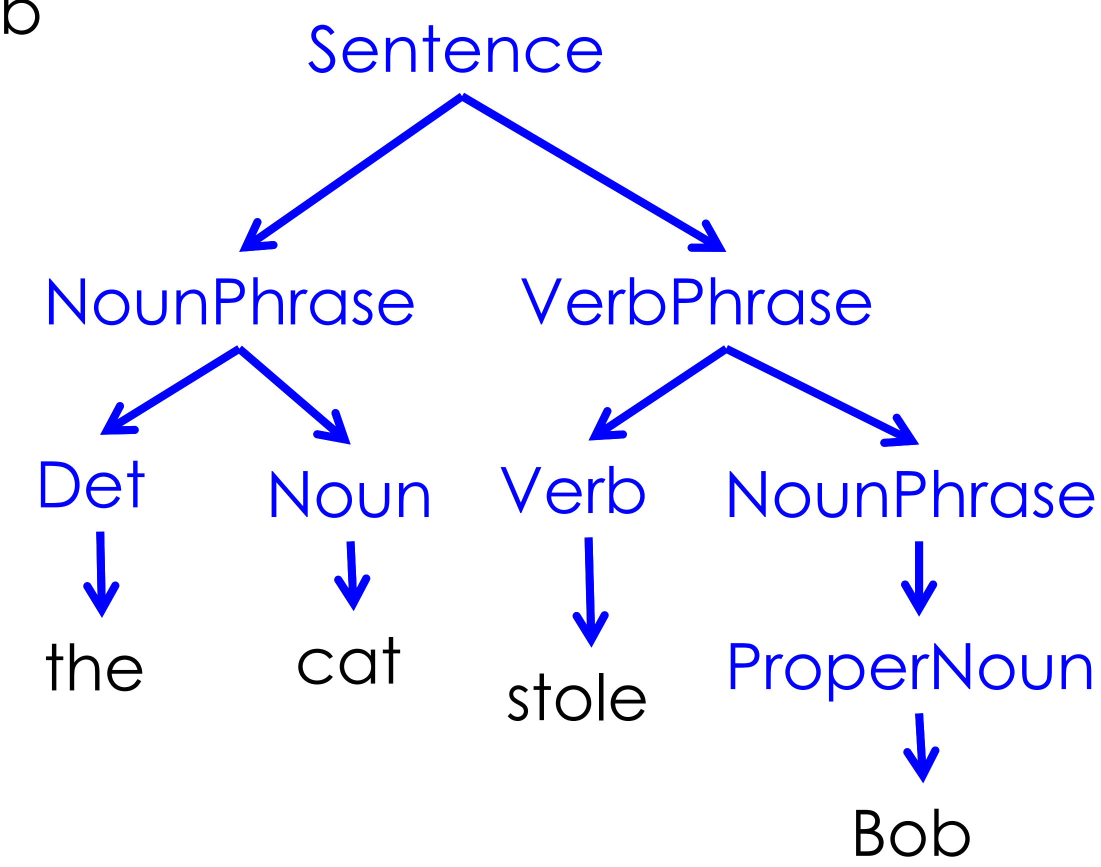
A More Complex Example

1. Sentence -> NounPhrase VerbPhrase
2. NounPhrase -> ProperNoun
3. VerbPhrase -> Verb NounPhrase
4. NounPhrase -> Det Noun
5. ProperNoun -> Bob
6. Verb -> stole
7. Noun -> cat
8. Det -> the



A More Complex Example

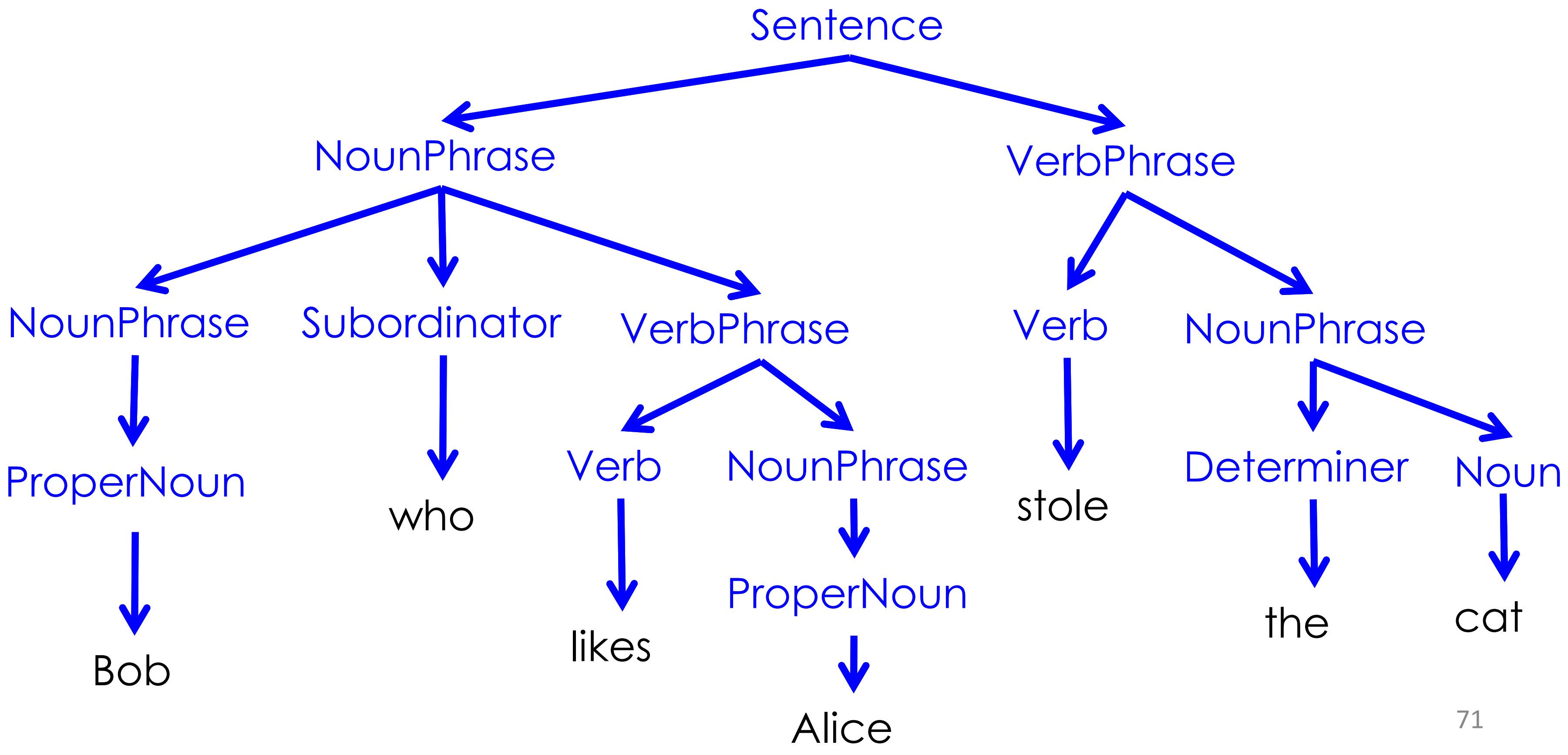
1. Sentence -> NounPhrase VerbPhrase
2. NounPhrase -> ProperNoun
3. VerbPhrase -> Verb NounPhrase
4. NounPhrase -> Det Noun
5. ProperNoun -> Bob
6. Verb -> stole
7. Noun -> cat
8. Det -> the



Recursive Structures

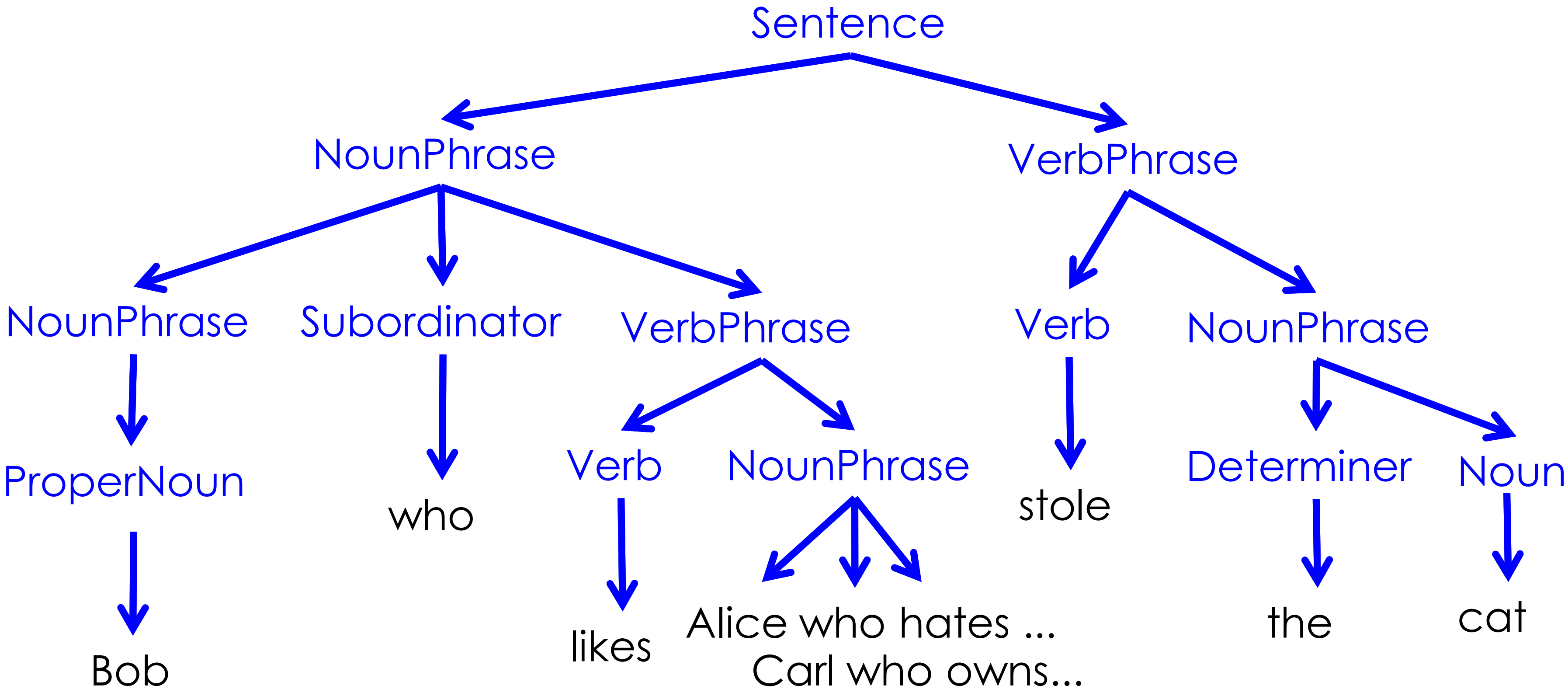
1. Sentence -> NounPhrase VerbPhrase
2. NounPhrase -> ProperNoun
3. NounPhrase -> Determiner Noun
4. NounPhrase -> NounPhrase Subordinator VerbPhrase
5. VerbPhrase -> Verb NounPhrase

Recursive rules: allow a circle in the derivation



Recursive Structures

1. Sentence -> NounPhrase VerbPhrase
2. NounPhrase -> ProperNoun
3. NounPhrase -> Determiner Noun
4. NounPhrase -> NounPhrase Subordinator VerbPhrase
5. VerbPhrase -> Verb NounPhrase



Sentence

NounPhrase

VerbPhrase

NounPhrase

Subordinator

VerbPhrase

Verb

NounPhrase

ProperNoun

who

Verb

NounPhrase

stole

Determiner

Noun

Bob

likes

Alice who hates ...

Carl who owns...

the

cat

Language

The **language of a grammar** is the set of all sentences that can be derived from the start symbol by rule applications.

Bob stole the cat

Bob stole Alice

Alice stole Bob who likes the cat

The cat likes Alice who stole Bob

Bob likes Alice who likes Alice who...

...

The grammar is a finite description of an infinite set of sentences

~~The Bob stole likes.~~

~~Stole stole stole.~~

~~Bob cat Alice likes.~~

...

Grammar Summary

A grammar is a formalism that can generate the sentences of a language.

Even though the grammar is finite, the sentences can be infinitely many.

We have seen a particular kind of grammars (context-free grammars), which produce a parse tree for the sentence they generate.

Parsing

Parsing is the process of, given a grammar and a sentence, finding the phrase structure tree.

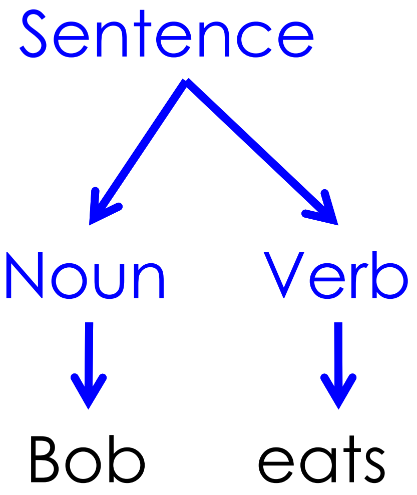
Sentence -> Noun Verb

Noun -> Bob

Verb -> eats

N = {Sentence, Noun, Verb}

T = {Bob, eats}



Parsing

A naïve parser would try all rules systematically from the top to arrive at the sentence.

Sentence -> Noun Verb

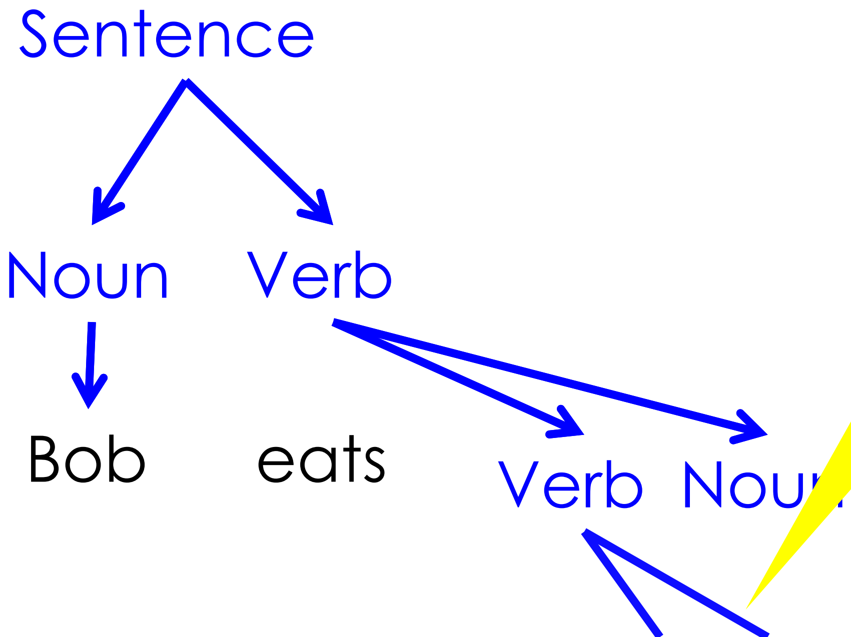
Noun -> Bob

Verb -> eats

Verb -> Verb Noun

N = {Sentence, Noun, Verb}

T = {Bob, eats}



Sentence

Noun Verb

Bob eats Verb Nou

This can go very wrong with recursive rules

Going bottom up is not much smarter

Earley Parser: Prediction

The **Earley Parser** is a parser that parses a sentence in $O(n^3)$ or less, where n is the length of the sentence.

* indicates current position

State 0: * Bob eats.

Sentence -> * Noun Verb, 0

Noun -> * Bob, 0

Put the start rule(s) of the grammar here.

Start index, initially 0

Prediction

If the state *i* contains the rule

X -> … * Y …., *j*

and if the grammar contains the rule

Y -> *something*

then add to state *i* the rule

Y -> * *something*, *i*

Earley Parser: Scanning

State 0: * Bob eats.

Sentence -> * Noun Verb, 0

Noun -> * Bob, 0

State 1: Bob * eats.

Noun -> Bob *, 0

Scanning

If z is a non-terminal and the state is

... * z ...

and if it contains the rule

X -> ... * z ...., i

then add that rule to the following state and advance the * by one in the new rule.

Earley Parser: Completion

State 0: * Bob eats.

Sentence -> * Noun Verb, 0

Noun -> * Bob, 0

State 1: Bob * eats.

Noun -> Bob *, 0

Sentence -> *Noun * Verb, 0

Completion

If the state contains

X -> … *, *i*

and if state *i* contains the rule

Y -> … * X …, *j*

then add that rule to the current state and advance the * by one in the new rule.

Earley Parser: Iteration

Prediction, Scanning and Completion are iterated until saturation. A state cannot contain the same rule twice.

State 0: * Bob eats.

Sentence -> * Noun Verb, 0

Noun -> * Bob, 0

State 1: Bob * eats.

Noun -> Bob *, 0

Sentence -> Noun * Verb, 0

Verb -> * Verb Noun, 1

~~Verb -> * Verb Noun, 1~~

Prediction

If state *i* contains

X -> ... * Y ...., *j*

and if the grammar contains

Y -> *something*

then add

Y -> * *something*, *i*

By prediction

Duplicate state, do not add it again

Earley Parser: Result

The process stops if no more scanner/predictor/completer can be applied.

State 2: Bob eats *.

...

Sentence -> Noun Verb *, 0

Iff the last state contains

Sentence -> *something* *, 0

(with the dot at the end), then the sentence conforms to the grammar.

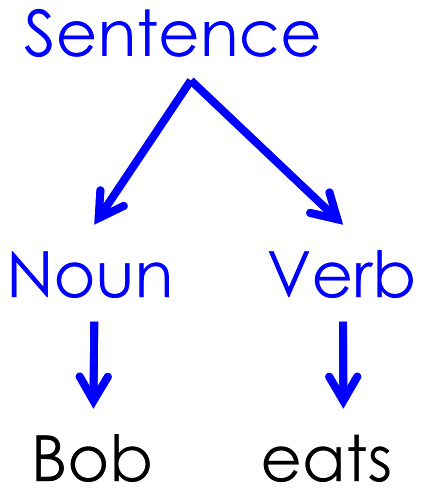
Earley Parser: Result

The parse tree can be read out (non-trivially) from the states by tracing the rules backward.

State 2: Bob eats *.

...

Sentence -> Noun Verb *, 0

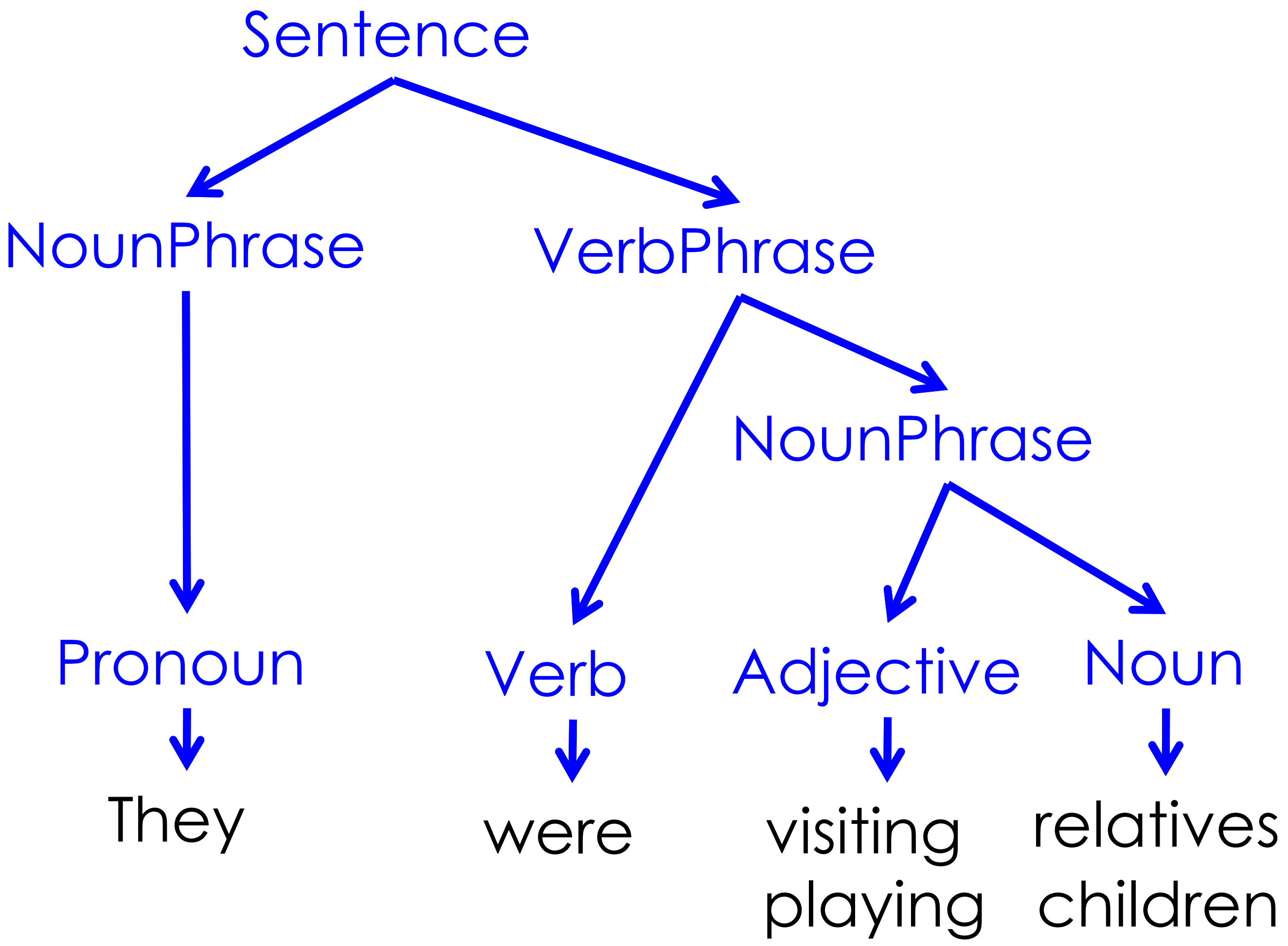


Sentence

Noun Verb

Bob eats

Syntactic Ambiguity



Sentence

NounPhrase

VerbPhrase

NounPhrase

Pronoun

Verb

Adjective

Noun

They

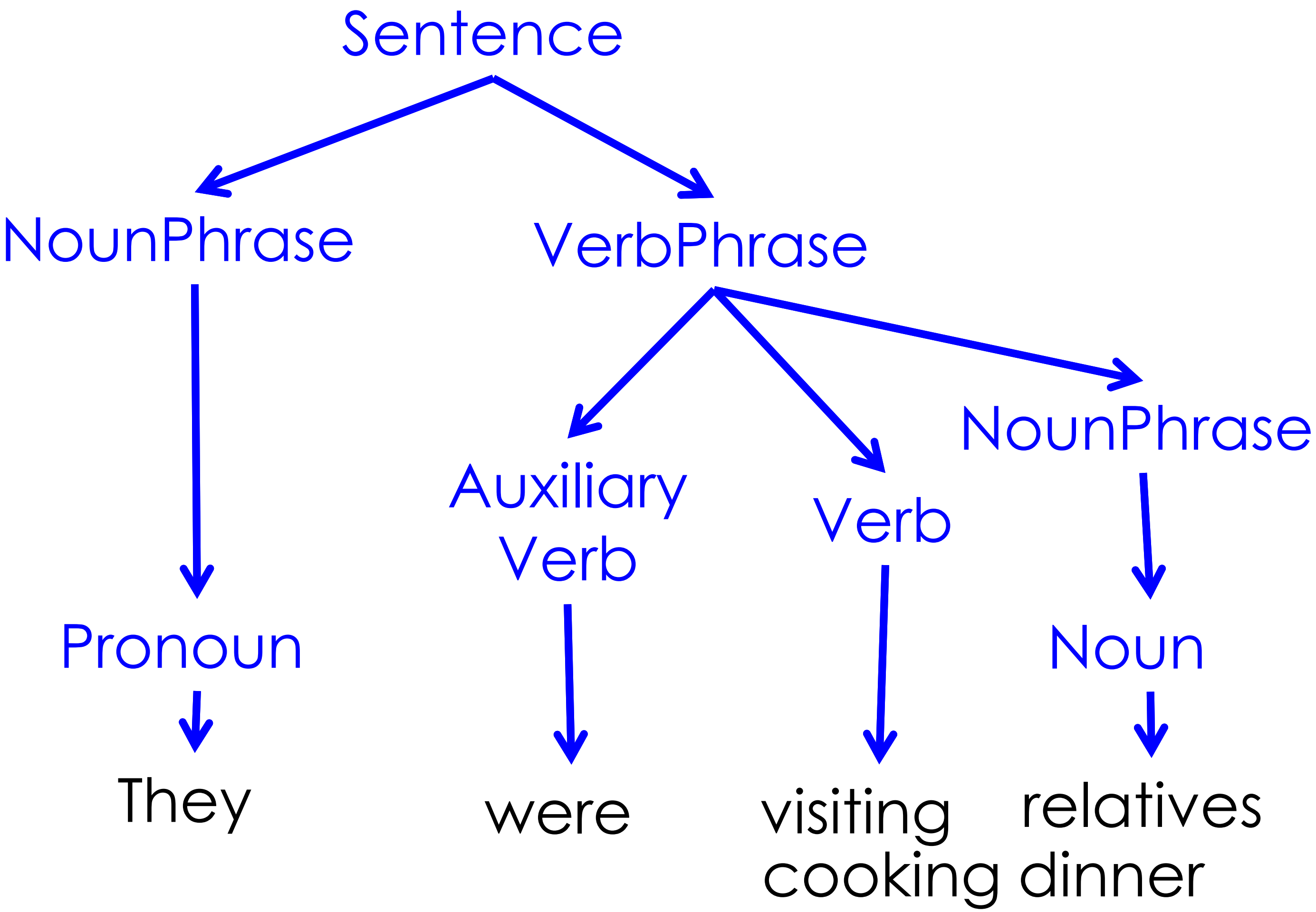
were

visiting
playing

relatives
children

= They were relatives who came to visit.

Syntactic Ambiguity



Sentence

NounPhrase

VerbPhrase

Pronoun

Auxiliary Verb

Verb

NounPhrase

Noun

They

were

visiting

relatives

cooking dinner

= They were on a visit to relatives.

Parsing Summary

Parsing is the process of, given a grammar and a sentence, finding the parse tree.

There may be multiple parse trees for a given sentence (a phenomenon called **syntactic ambiguity**).

The Earley Parser is an efficient parser for context free grammars.

What we cannot (yet) do

What is difficult to do with context-free grammars:

- agreement between words

  Bob kicks the dog.
  I kicks the dog. ✗

We could differentiate VERB3rdPERSON and VERB1stPERSON, but this would multiply the non-terminal symbols exponentially.

- sub-categorization frames

  Bob sleeps.
  Bob sleeps you. ✗

- meaningfulness

  Bob switches the computer off.
  Bob switches the cat off. ✗

Feature Structures

A **feature structure** is a mapping from attributes to values.

Each **value** is an atomic value or a feature structure.

A sample feature structure:

Category = Noun
Agreement = { Number = Singular
Person = Third }

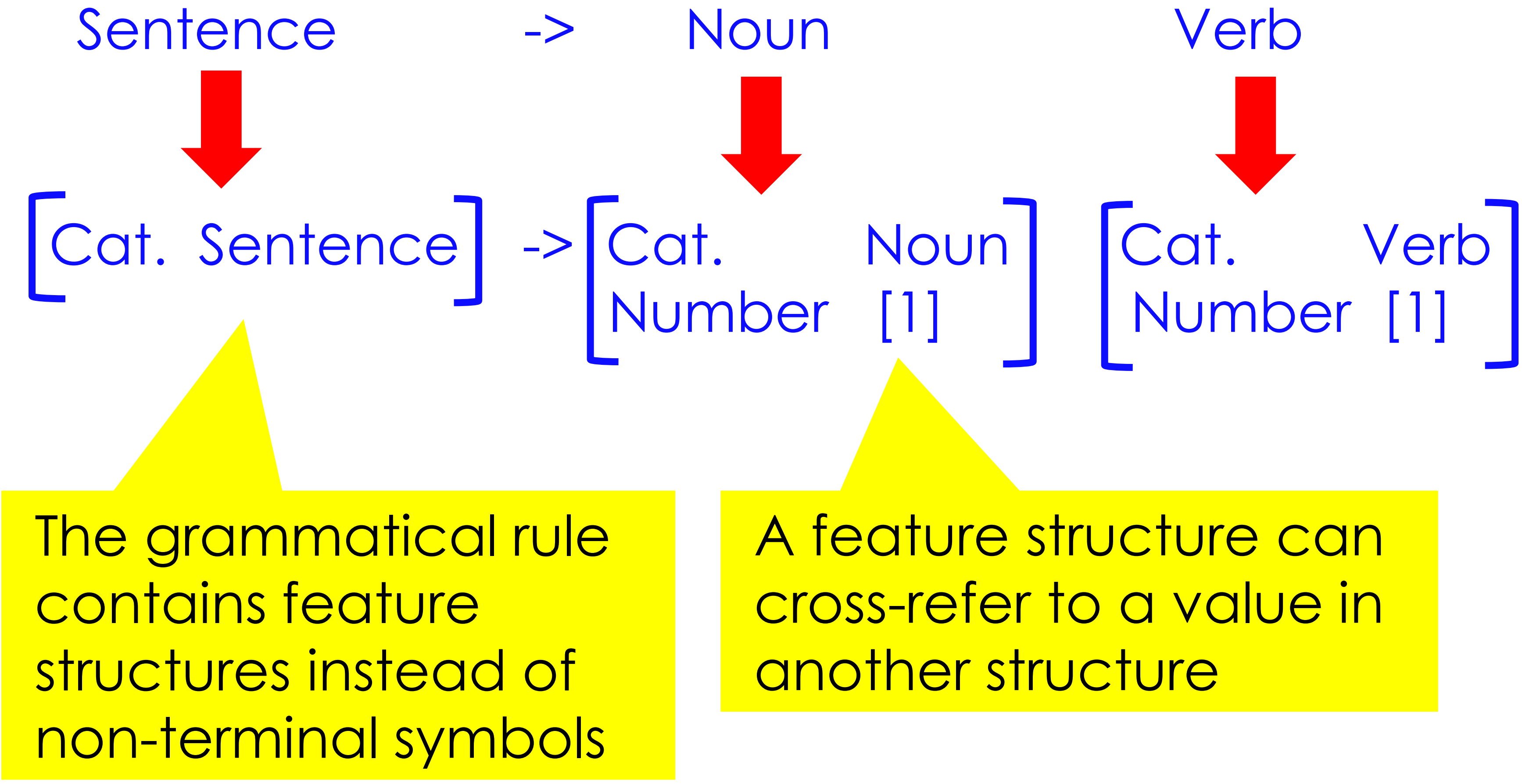
Attribute = Value

Represented differently:

$$\begin{bmatrix} \text{Category} & \text{Noun} \\ \text{Agreement} & \begin{bmatrix} \text{Number} & \text{Singular} \\ \text{Person} & \text{Third} \end{bmatrix} \end{bmatrix}$$

Feature Structure Grammars

A **feature structure grammar** combines traditional grammar with feature structures in order to model agreement.



Sentence -> Noun Verb

$$\begin{bmatrix}\text{Cat.} & \text{Sentence}\end{bmatrix} \rightarrow \begin{bmatrix}\text{Cat.} & \text{Noun}\\ \text{Number} & [1]\end{bmatrix} \begin{bmatrix}\text{Cat.} & \text{Verb}\\ \text{Number} & [1]\end{bmatrix}$$

The grammatical rule contains feature structures instead of non-terminal symbols

A feature structure can cross-refer to a value in another structure

Feature Structure Grammars

A **feature structure grammar** combines traditional grammar with feature structures in order to model agreement.

$$\begin{bmatrix}\text{Cat.} & \text{Sentence}\end{bmatrix} \text{ -> } \begin{bmatrix}\text{Cat.} & \text{Noun}\\ \text{Number} & [1]\end{bmatrix} \begin{bmatrix}\text{Cat.} & \text{Verb}\\ \text{Number} & [1]\end{bmatrix}$$

$$\begin{bmatrix}\text{Cat.} & \text{Noun}\\ \text{Number} & \text{Singular}\\ \text{Gender} & \text{Male}\end{bmatrix} \text{ -> } \text{Bob}$$

Rules with terminals have constant values in their feature structures

Rule Application

Grammar rules are applied as usual.

$$\begin{bmatrix}\text{Cat.} & \text{Sentence}\end{bmatrix}$$

Apply rule [Cat. Sentence] -> ....

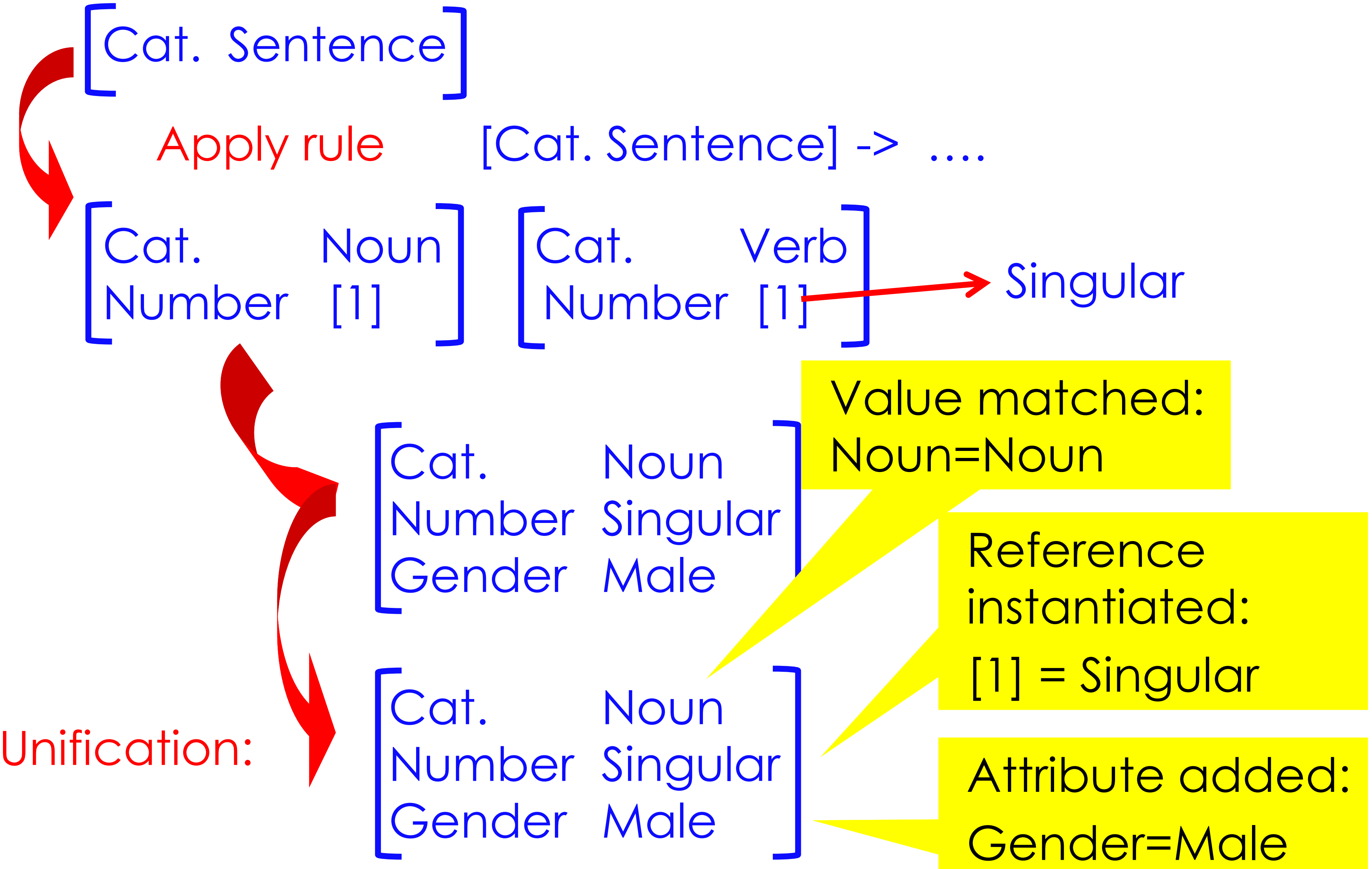
$$\begin{bmatrix}\text{Cat.} & \text{Noun}\\ \text{Number} & [1]\end{bmatrix}\quad\begin{bmatrix}\text{Cat.} & \text{Verb}\\ \text{Number} & [1]\end{bmatrix}$$

Apply rule $\begin{bmatrix}\text{Cat.} & \text{Noun}\\ \text{Number} & \text{Singular}\\ \text{Gender} & \text{Male}\end{bmatrix}$ -> Bob

Feature structures have to be **unified** before applying a rule: Additional attributes are added, references instantiated, and values matched (possibly recursively)

Unification

Grammar rules are applied as usual.



$\begin{bmatrix} \text{Cat.} & \text{Sentence} \end{bmatrix}$

Apply rule [Cat. Sentence] -> ....

$\begin{bmatrix} \text{Cat.} & \text{Noun} \\ \text{Number} & [1] \end{bmatrix} \begin{bmatrix} \text{Cat.} & \text{Verb} \\ \text{Number} & [1] \end{bmatrix}$ → Singular

$\begin{bmatrix} \text{Cat.} & \text{Noun} \\ \text{Number} & \text{Singular} \\ \text{Gender} & \text{Male} \end{bmatrix}$

Unification: $\begin{bmatrix} \text{Cat.} & \text{Noun} \\ \text{Number} & \text{Singular} \\ \text{Gender} & \text{Male} \end{bmatrix}$

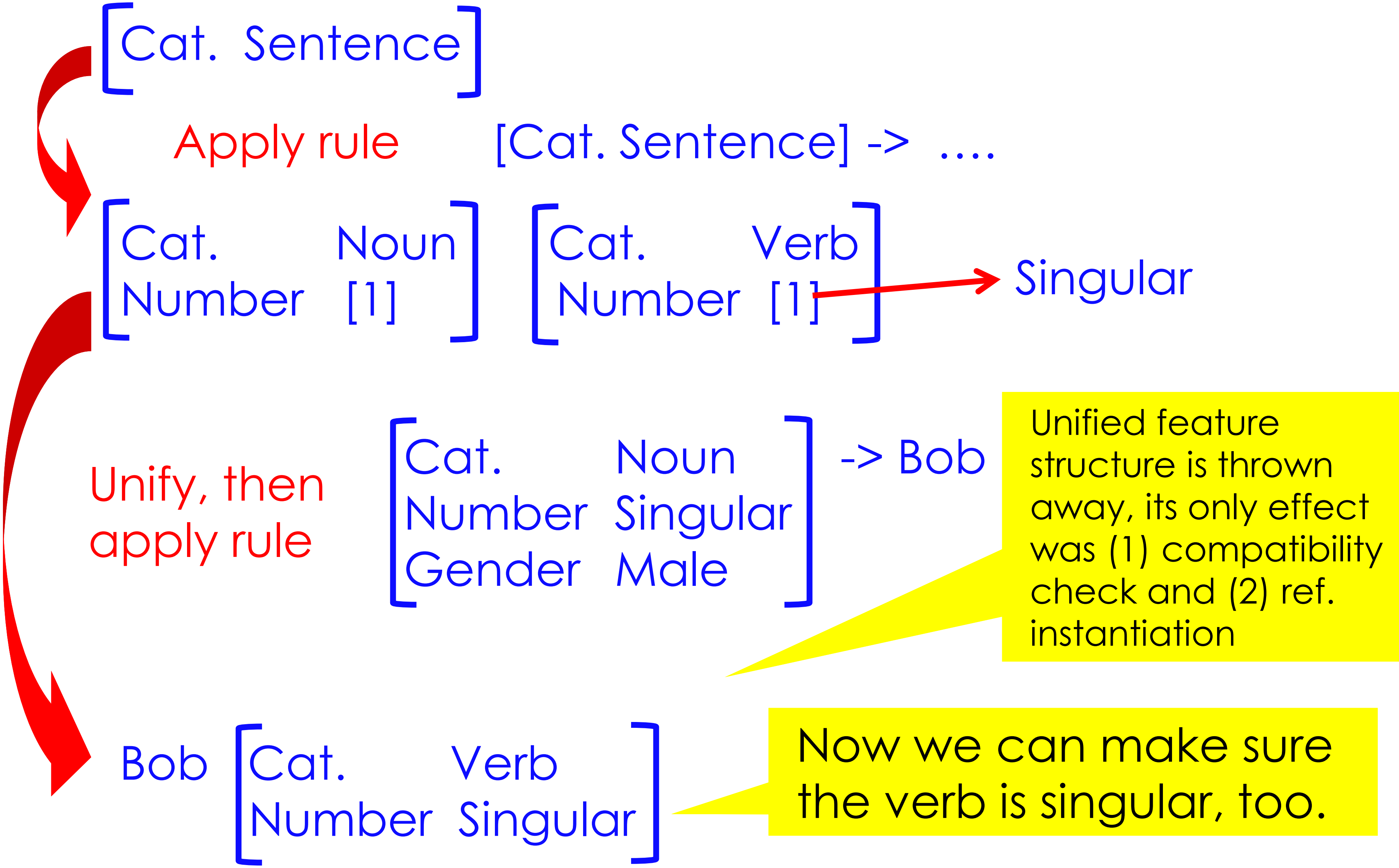
Value matched: Noun=Noun

Reference instantiated: [1] = Singular

Attribute added: Gender=Male

Unification

Grammar rules are applied as usual.



[Cat. Sentence]

Apply rule [Cat. Sentence] -> ....

[Cat. Noun, Number [1]] [Cat. Verb, Number [1]] Singular

Unify, then apply rule [Cat. Noun, Number Singular, Gender Male] -> Bob

Unified feature structure is thrown away, its only effect was (1) compatibility check and (2) ref. instantiation

Bob [Cat. Verb, Number Singular]

Now we can make sure the verb is singular, too.

Feature Structures Summary

Feature structures can represent additional information on grammar symbols and enforce agreement.

We just saw a very naïve grammar with feature structures.

Various more sophisticated grammars use feature structures:

- generalizes phrase structure grammars
- head-driven phrase structure grammars (HPSG)
- Lexical-functional grammars (LFG)

Fields of Linguistics

/ai θot.../ ✓
(Phonology, the study of pronunciation)

go/going ✓
(Morphology, the study of word constituents)

I thought they're never going to hear me ‘cause they're screaming all the time. [Elvis Presley]

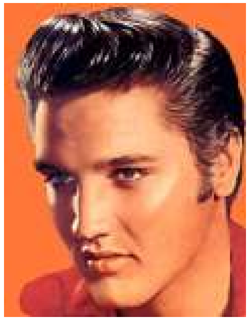
Sentence

Noun phrase

Verbal phrase

(Syntax, the study of grammar)

✓

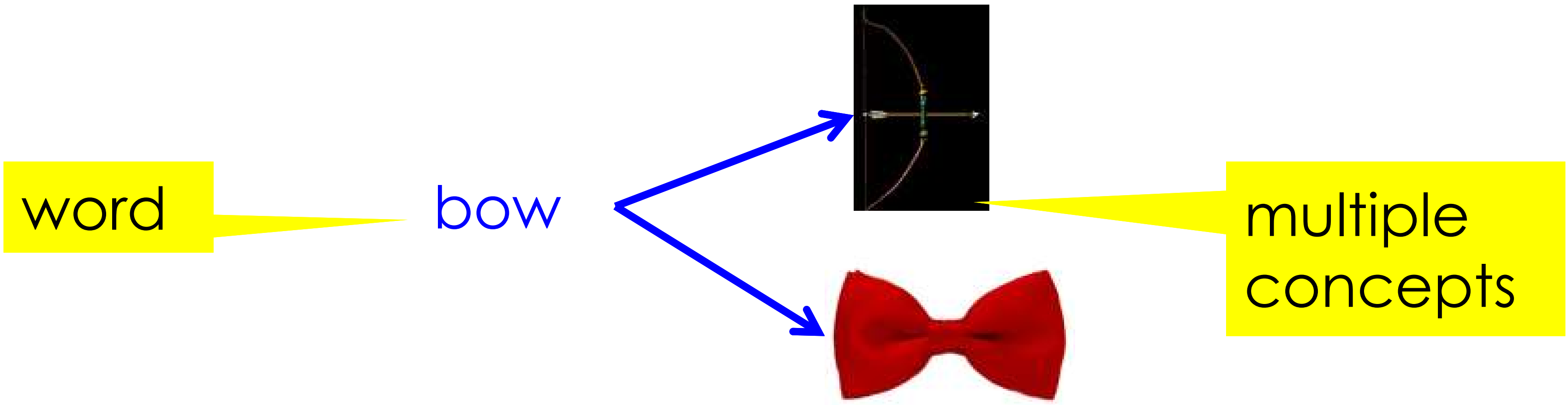
“I” = 

(Semantics, the study of meaning)

It doesn’t matter what I sing.
(Pragmatics, the study of language use)

Meaning of Words

- A word can refer to multiple concepts/meanings/senses (such a word is called a **homonym**)

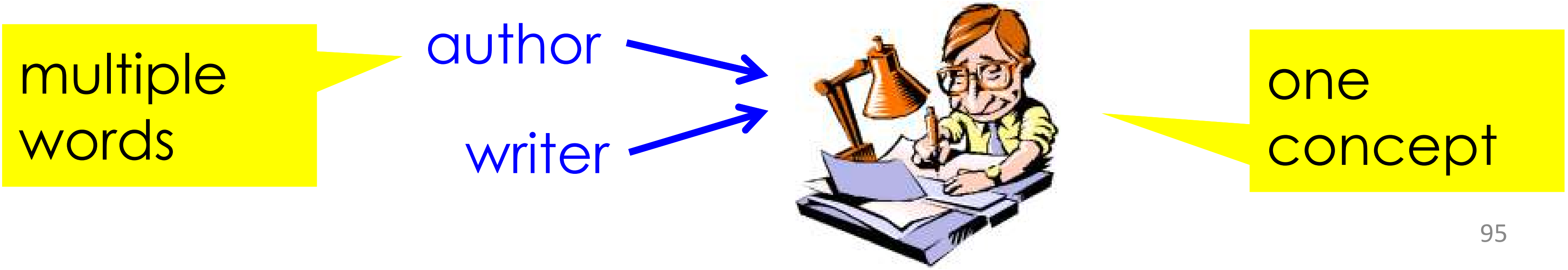


word

bow

multiple concepts

- A concept can be expressed by multiple words (such words are called **synonyms**)



multiple words

author

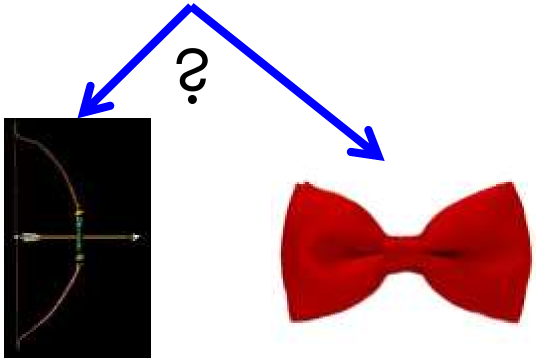
writer

one concept

Word Sense Disambiguation

Word Sense Disambiguation (WSD) is the process of finding the meaning of a word in a sentence.

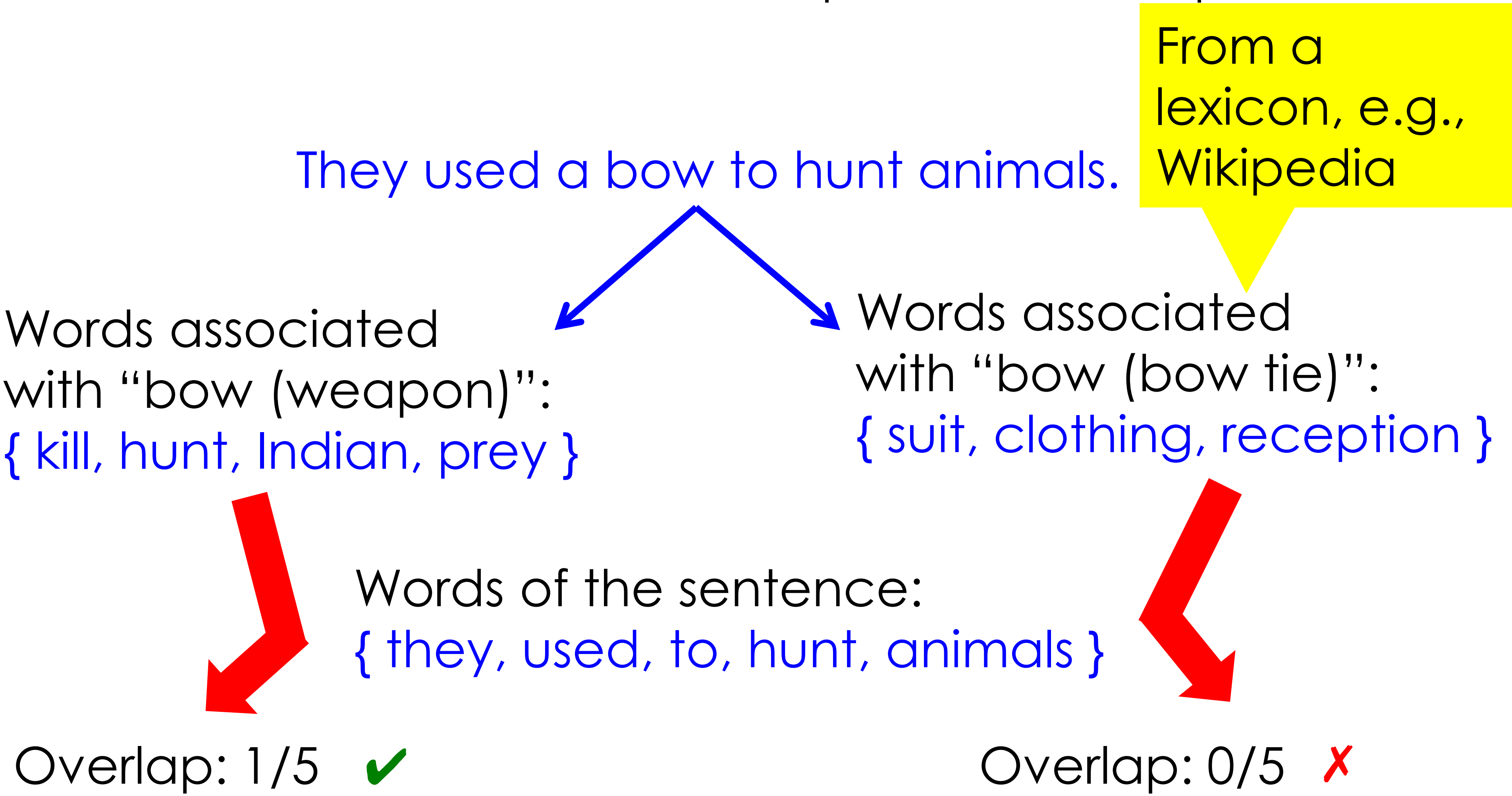
They used a bow to hunt animals.



How can a machine do that without understanding the sentence?

Bag-of-Words WSD

Bag-of-Words WSD compares the words of the sentence to words associated to each of the possible concepts.



From a lexicon, e.g., Wikipedia

They used a bow to hunt animals.

Words associated with "bow (weapon)": { kill, hunt, Indian, prey }

Words associated with "bow (bow tie)": { suit, clothing, reception }

Words of the sentence: { they, used, to, hunt, animals }

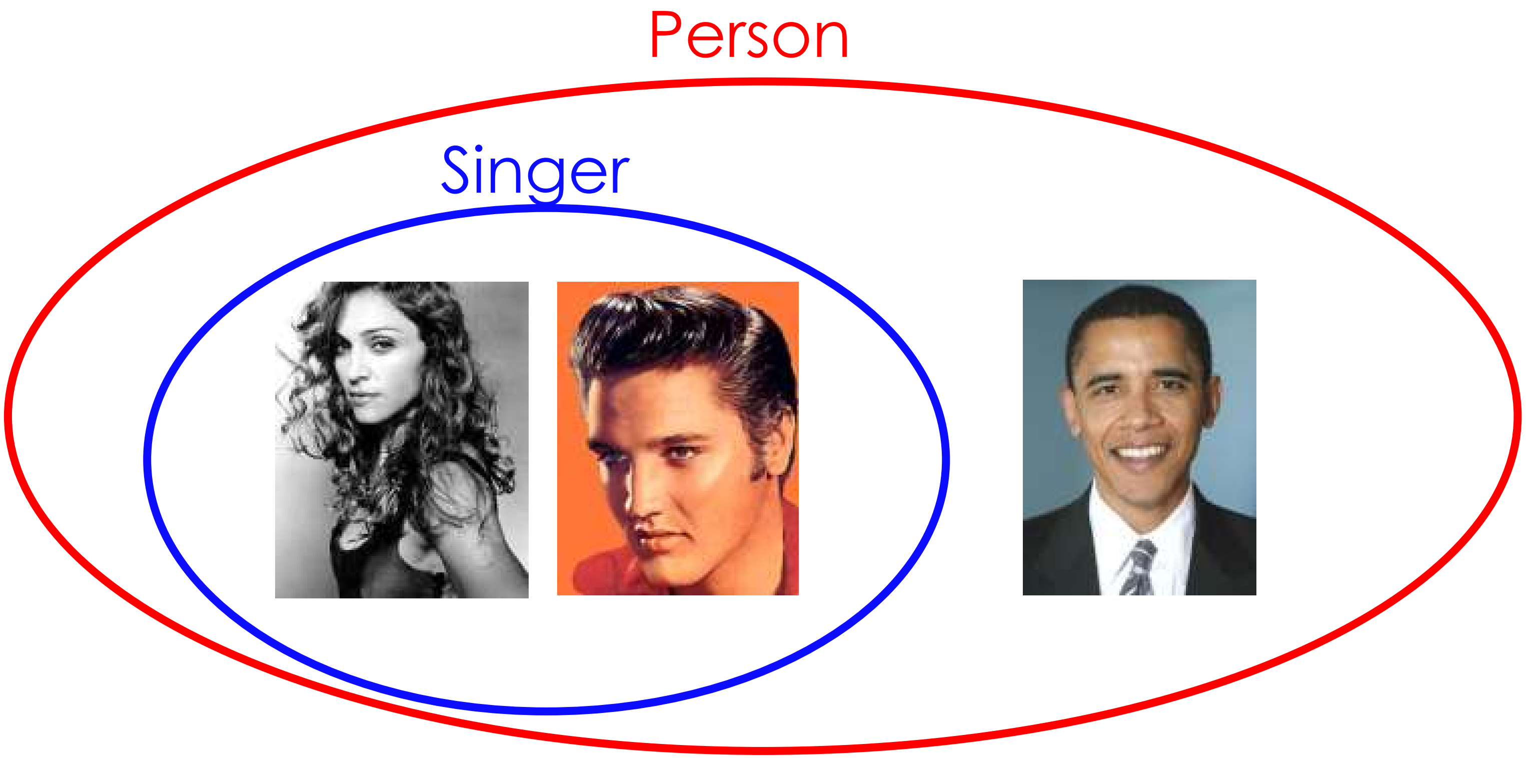
Overlap: 1/5 ✔

Overlap: 0/5 ✗

Hyponymy

A concept is a **hypernym** of another concept, if its meaning is more general that that of the other concept.

The other concept is called the **hyponym**.



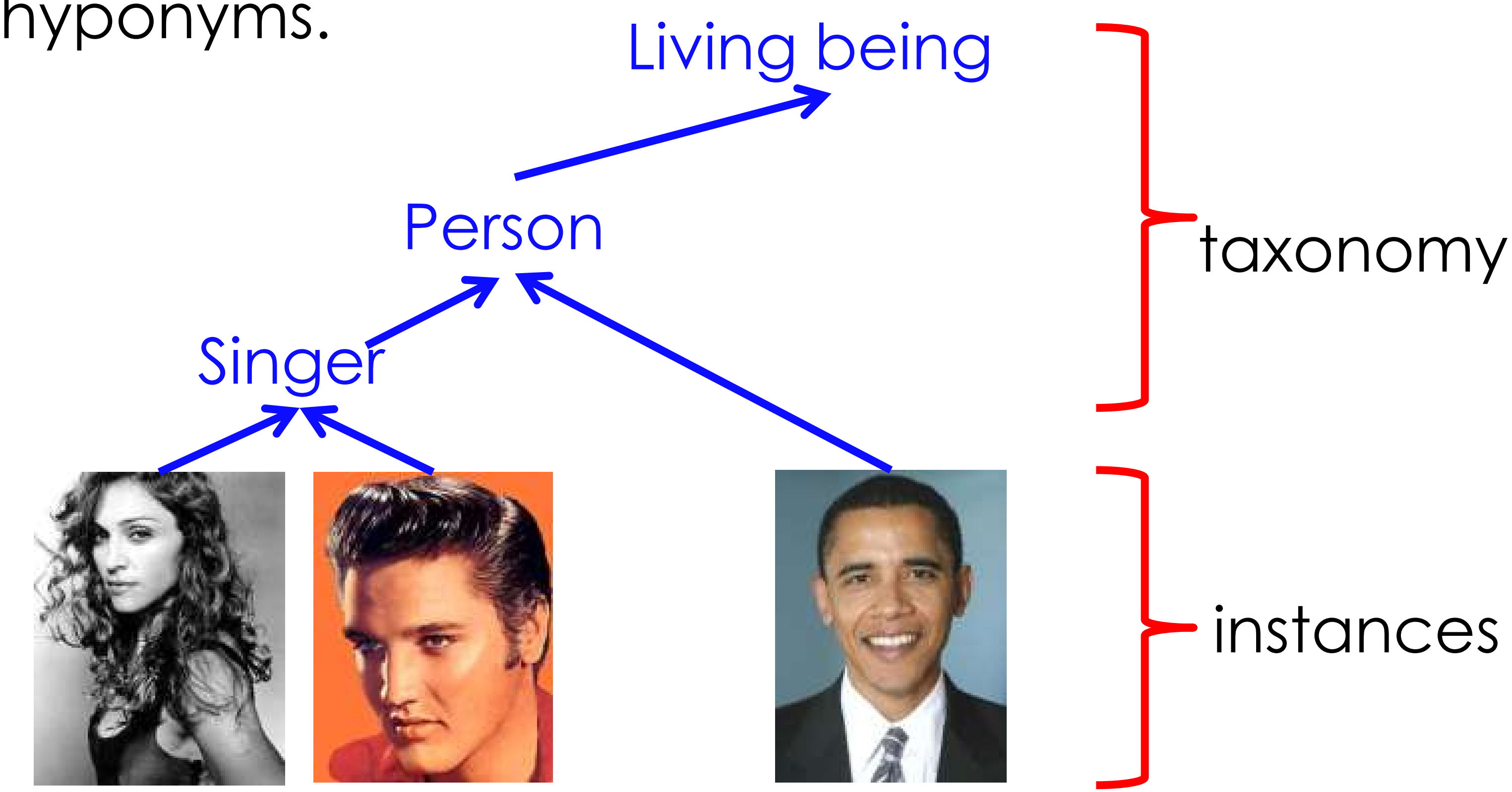
Person

Singer

Every singer is a person => “singer” is a hyponym of “person”

Taxonomy

A **taxonomy** is a directed acyclic graph, in which hypernyms dominate hyponyms.



Living being

Person

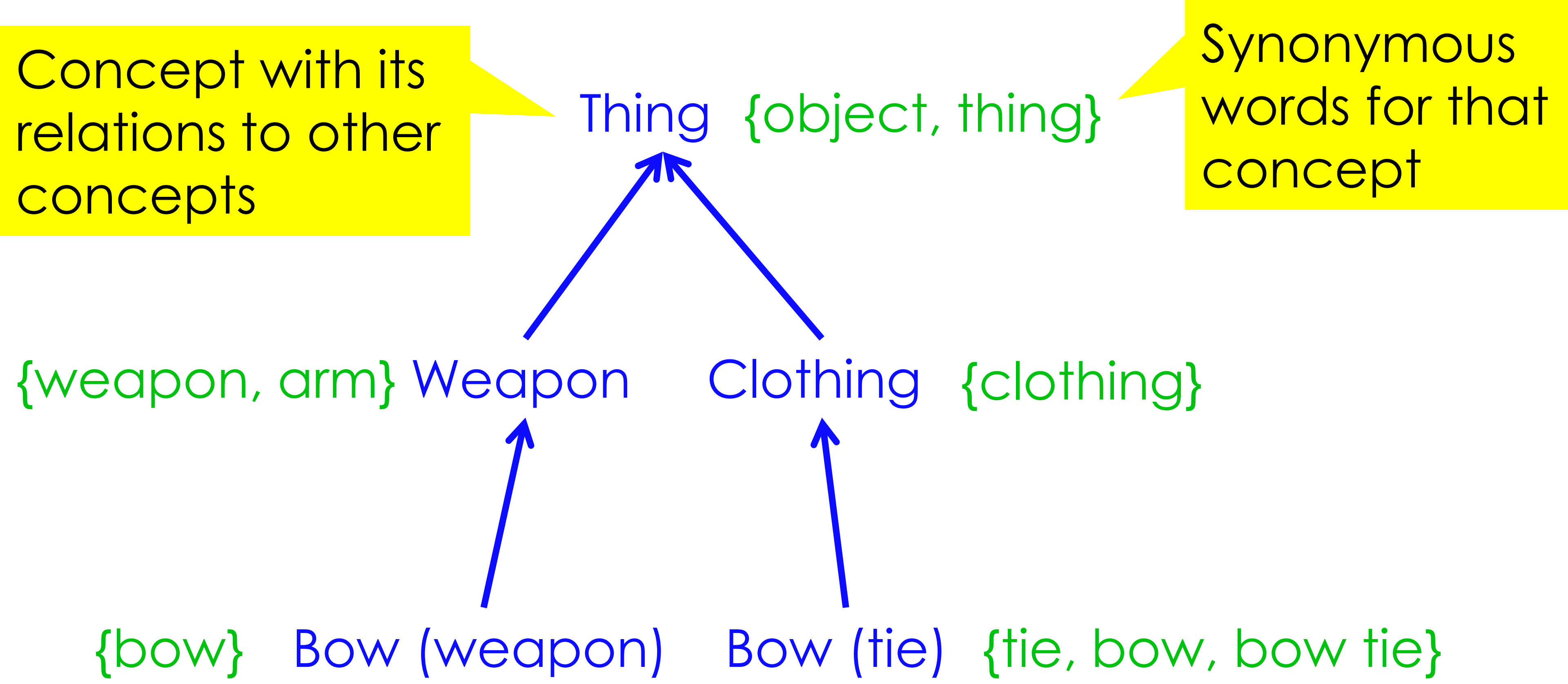
Singer

taxonomy

instances

WordNet

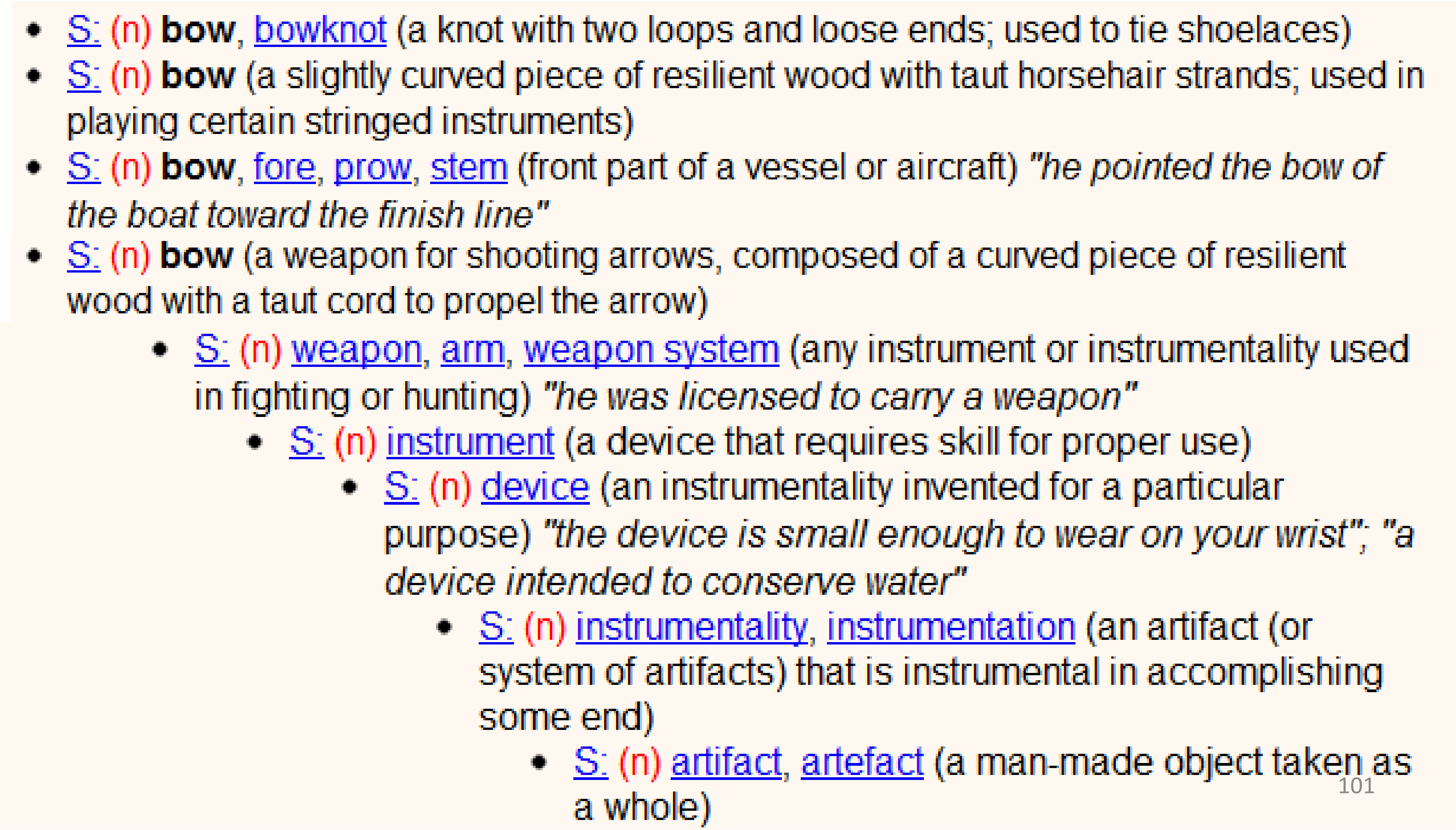
WordNet is a lexicon of the English language, which contains a taxonomy of concepts plus much additional information.



Concept with its relations to other concepts
Thing {object, thing}
Synonymous words for that concept
{weapon, arm} Weapon
Clothing {clothing}
{bow} Bow (weapon)
Bow (tie) {tie, bow, bow tie}

WordNet

Example: the word “bow” in WordNet, http://wordnet.princeton.edu

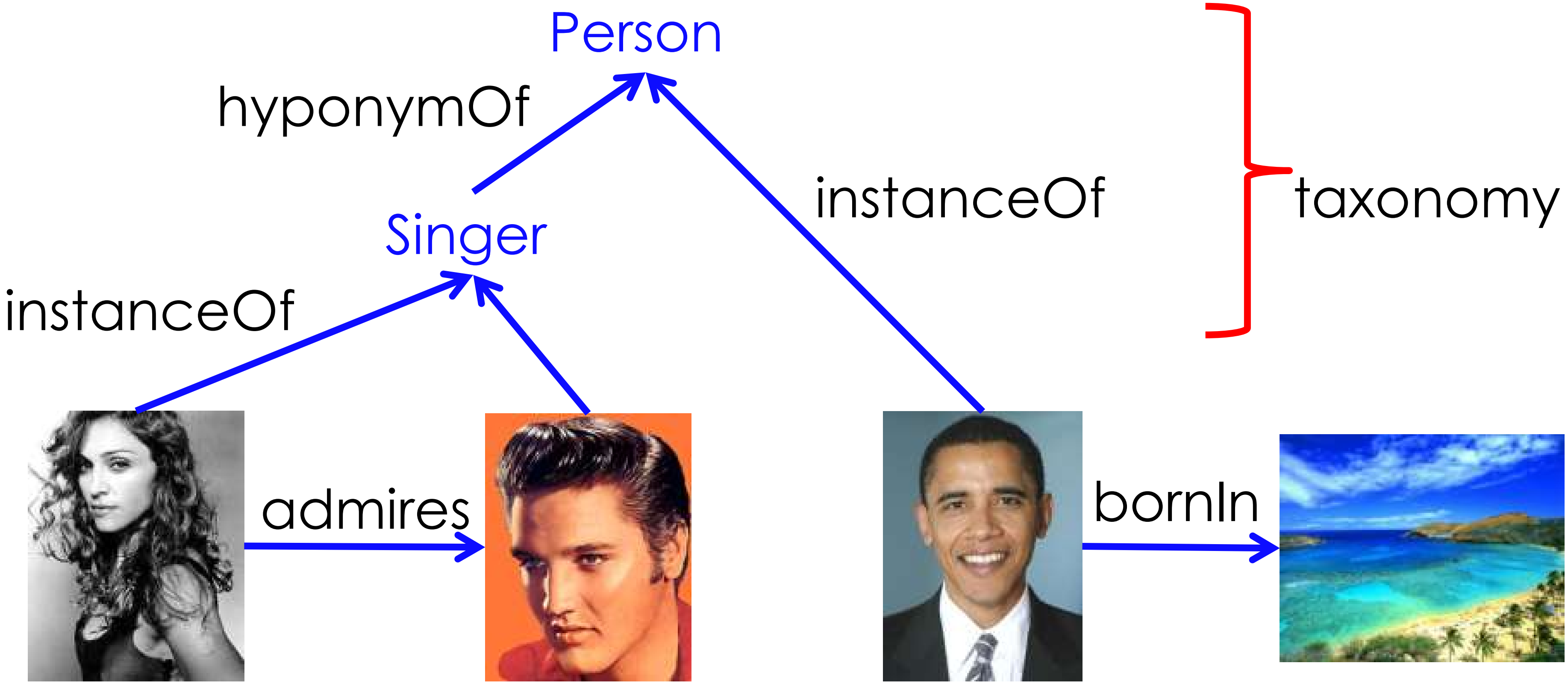


- S: (n) **bow**, bowknot (a knot with two loops and loose ends; used to tie shoelaces)
- S: (n) **bow** (a slightly curved piece of resilient wood with taut horsehair strands; used in playing certain stringed instruments)
- S: (n) **bow**, fore, prow, stem (front part of a vessel or aircraft) *"he pointed the bow of the boat toward the finish line"*
- S: (n) **bow** (a weapon for shooting arrows, composed of a curved piece of resilient wood with a taut cord to propel the arrow)
    - S: (n) weapon, arm, weapon system (any instrument or instrumentality used in fighting or hunting) *"he was licensed to carry a weapon"*
        - S: (n) instrument (a device that requires skill for proper use)
            - S: (n) device (an instrumentality invented for a particular purpose) *"the device is small enough to wear on your wrist"; "a device intended to conserve water"*
                - S: (n) instrumentality, instrumentation (an artifact (or system of artifacts) that is instrumental in accomplishing some end)
                    - S: (n) artifact, artefact (a man-made object taken as a whole)

Ontology

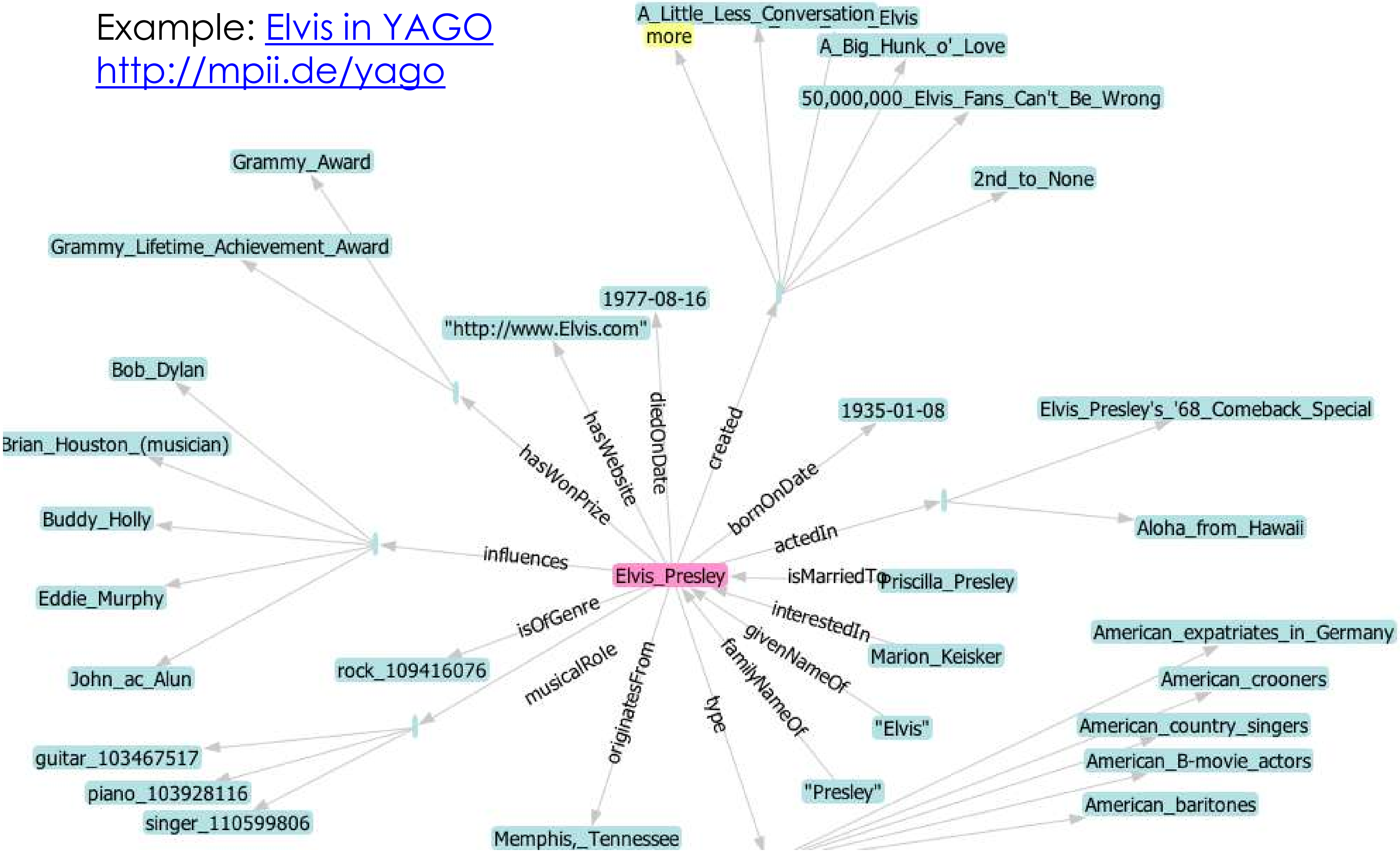
An **ontology** is a graph of instances, concepts and relationships between them.

An ontology includes a taxonomy.



Sample Ontology: YAGO

Example: Elvis in YAGO
http://mpii.de/yago



Meanings of Words – Summary

- One word can have **multiple meanings**
  and one meaning can be represented by multiple words.

- Figuring out the meaning of a word in a sentence is called **Word Sense Disambiguation**.
  A naïve approach just looks at the context of the word.

- Concepts can be arranged in a **taxonomy**.
  (example: **WordNet**)

- **Ontologies** also contain facts about instances.
  (example: YAGO)

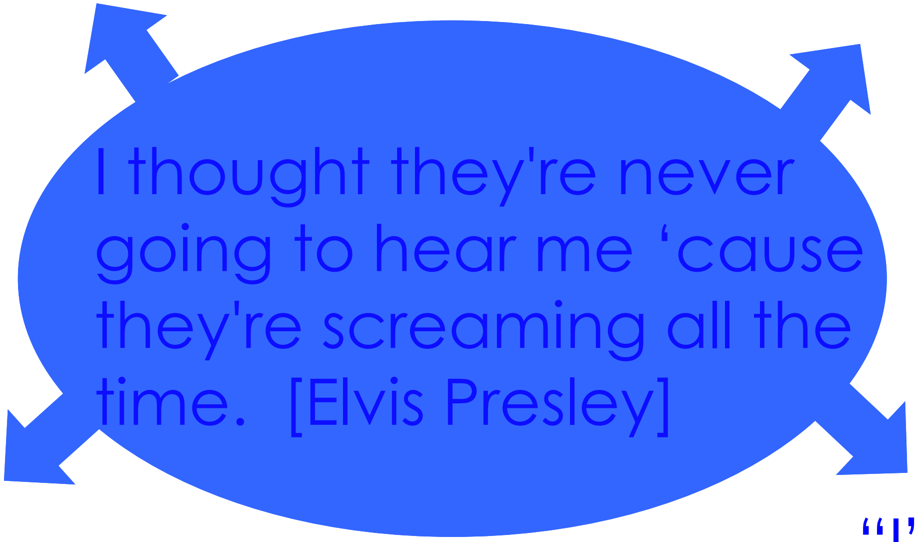
Fields of Linguistics

/ai θot…/ ✓

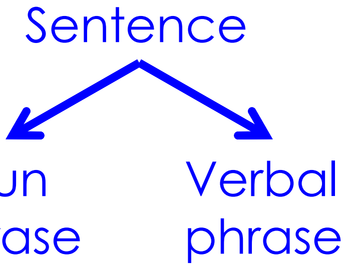
(Phonology, the study of pronunciation)

go/going ✓

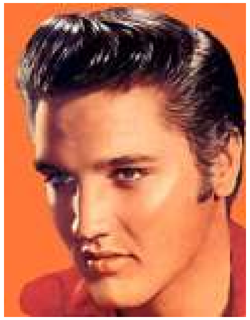
(Morphology, the study of word constituents)



I thought they're never going to hear me ‘cause they're screaming all the time. [Elvis Presley]



(Syntax, the study of grammar) ✓

“I” =  ✓

(Semantics, the study of meaning)

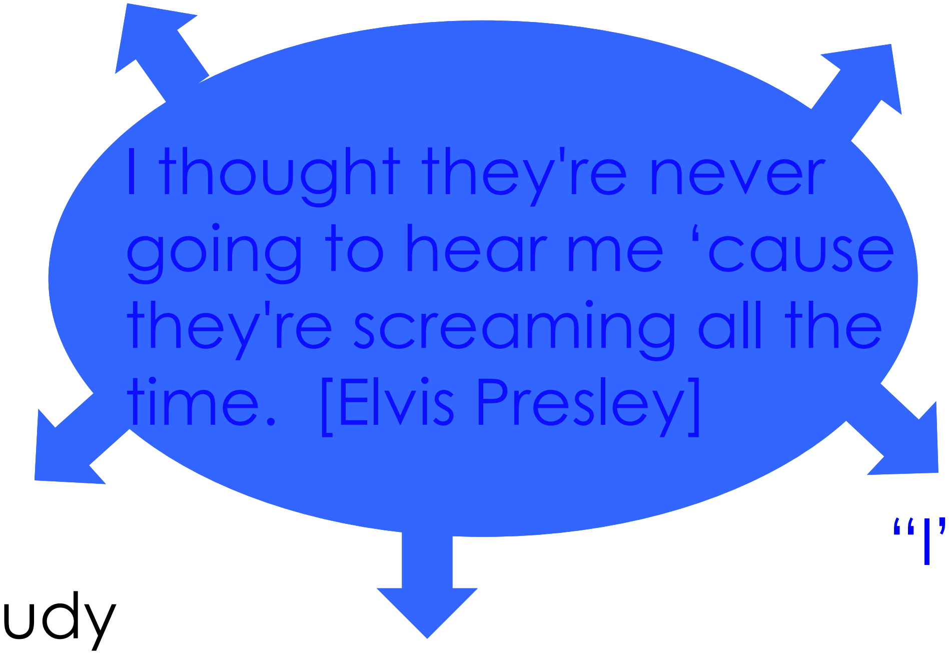
Fields of Linguistics

/ai θot…/ ✓

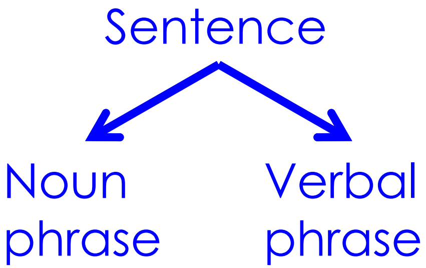
(Phonology, the study of pronunciation)

go/going ✓

(Morphology, the study of word constituents)



I thought they're never going to hear me ‘cause they're screaming all the time. [Elvis Presley]



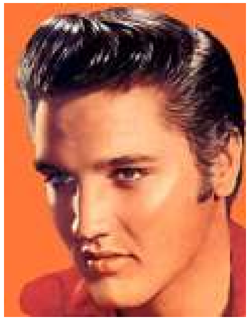
(Syntax, the study of grammar)

✓

It doesn’t matter what I sing.

(Pragmatics, the study of language use)

“I” =

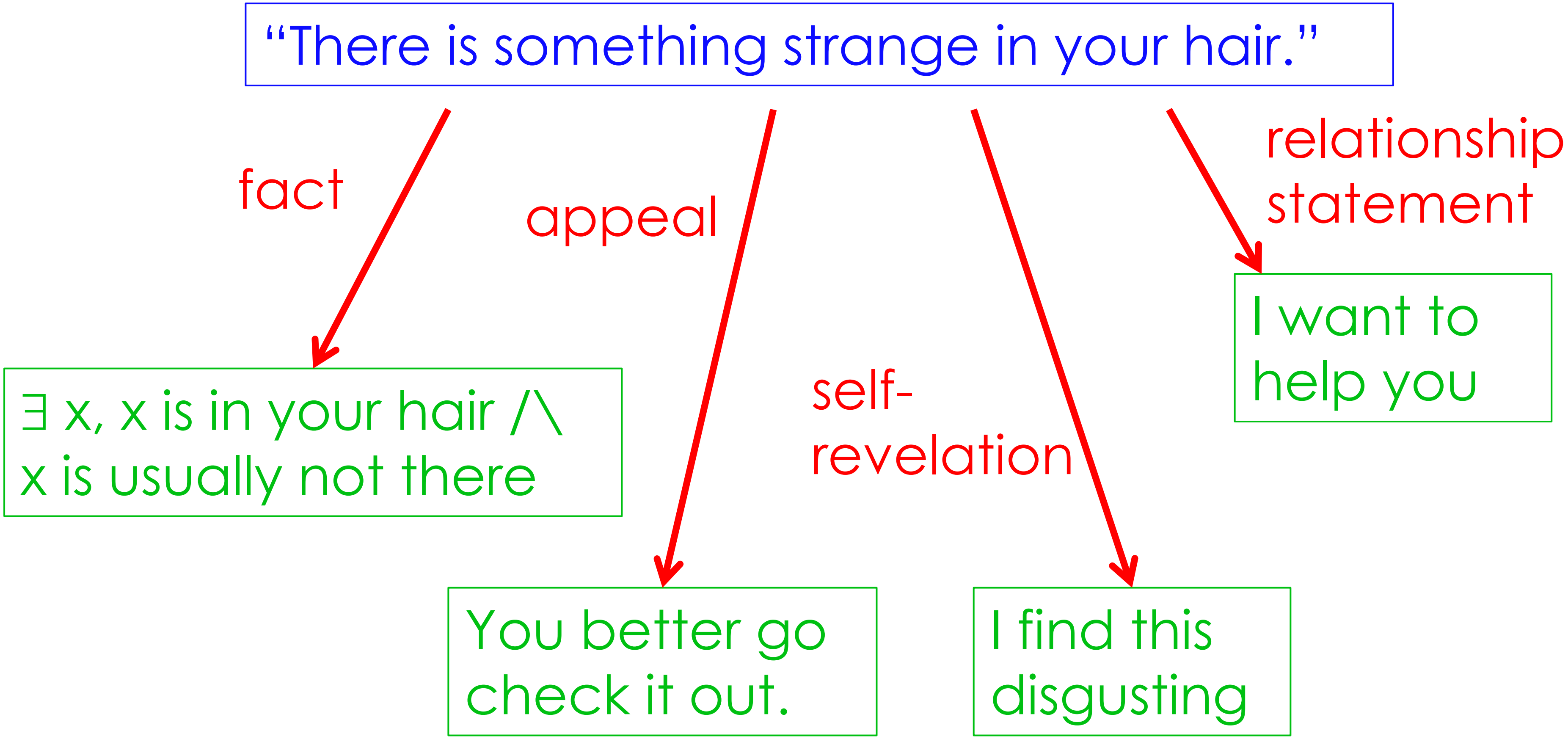


✓

(Semantics, the study of meaning)

Four Sides Model

The **Four Sides Model** hypothesizes that there are 4 messages in every utterance. [Friedemann Schulz von Thun]



“There is something strange in your hair.”

fact

∃ x, x is in your hair /\ x is usually not there

appeal

You better go check it out.

self-revelation

I find this disgusting

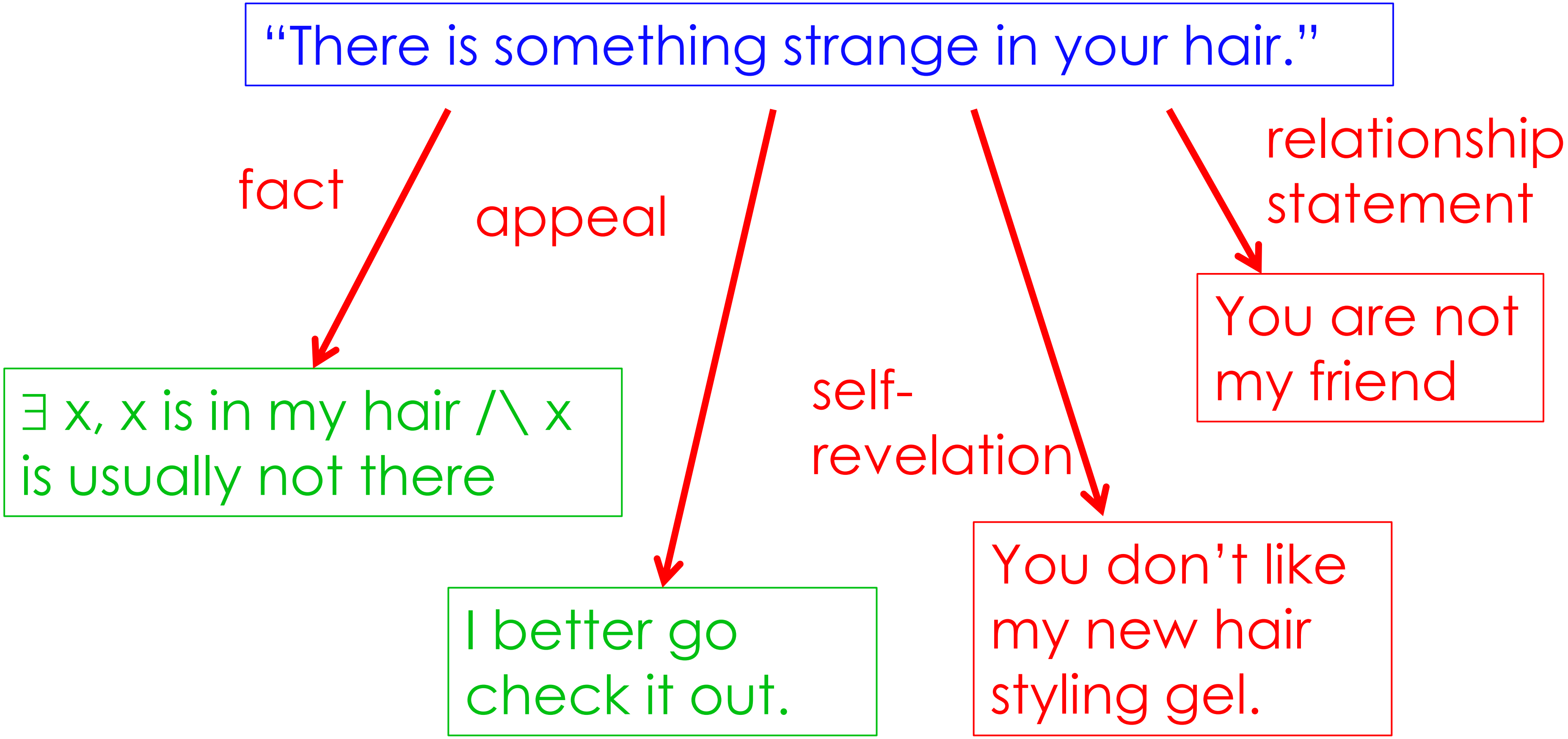
relationship statement

I want to help you

⇒ We say much more than words!

Sender / Receiver

The receiver of the utterance may read different messages.



"There is something strange in your hair."

fact

∃ x, x is in my hair /\ x is usually not there

appeal

I better go check it out.

self-revelation

You don't like my new hair styling gel.

relationship statement

You are not my friend

⇒ What gets sent is not necessarily what is received.

Indirect Speech Act

An **indirect speech act** is an utterance that intentionally transmits an implicit message. [John Searle]

What is said…

Bob: Do you want to go?
Alice: It is raining outside…

What it means…

Alice: No.

Searle proposes the following algorithm:

1. Collect the factual meaning of the utterance

   It is raining outside.

2. If that meaning is unrelated

   The fact that it rains is unrelated to Bob's question.

3. Then assume that the utterance means something else.

   Alice probably does not want to go.

Presuppositions

A **presupposition** is an implicit assumption about the world that the receiver makes when receiving a message.

What is said…	*What it presupposes…*
I stopped playing guitar.	I played guitar before.
The King of England laughs.	England has a king.
I realized that she was there. (cf.: I thought that she was there)	She was indeed there.
Bob managed to open the door. (cf: Bob happened to open the door)	Bob wanted to open the door.

Illocutionary Speech Acts

An **illocutionary speech act** is an utterance that does more than transferring a message. [John L. Austin]

What is said…	*How the world changes…*
Bob: "I will buy the car!"	Legal effect: a promise
Bob: "I just escaped from prison and I have a gun!"	Psychological effect on the audience.
Bob: "I hereby legally pronounce you husband and wife"	Elvis and Priscilla are married.

Pragmatics Summary

A sentence says much more than the actual words

- It may carry an appeal, a self-revelation and a relationship statement.
- It may carry an intended implicit message
- It carries presuppositions
- It may have a tangible effect on the world.

Computers are still far from catching these messages.

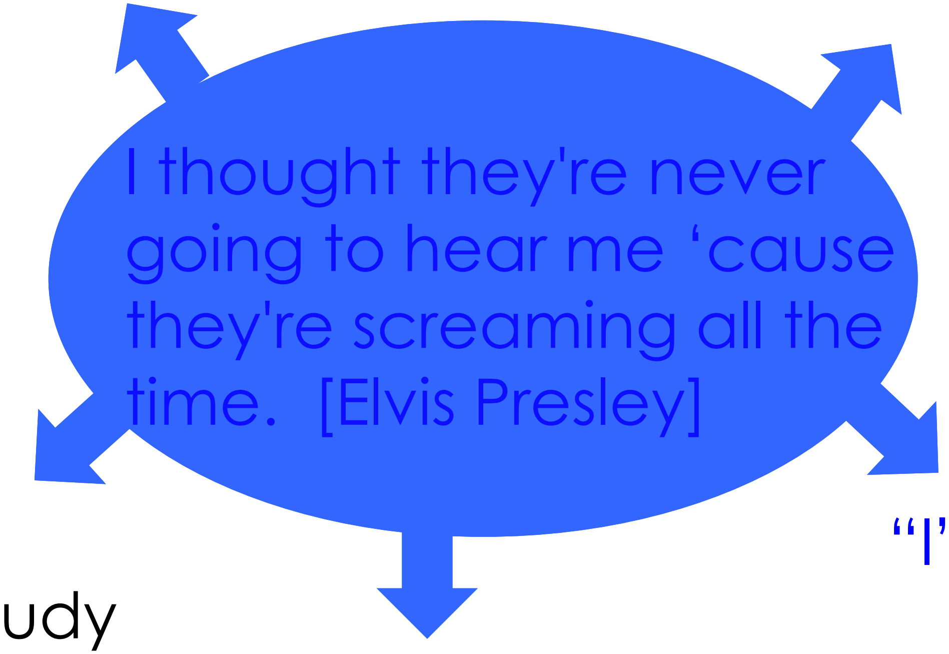
Fields of Linguistics

/ai θot…/ ✓

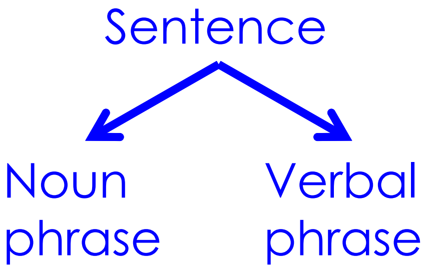
(Phonology, the study of pronunciation)

go/going ✓

(Morphology, the study of word constituents)



I thought they're never going to hear me 'cause they're screaming all the time. [Elvis Presley]

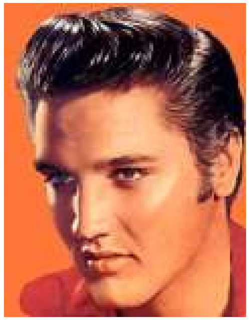


(Syntax, the study of grammar)

✓

It doesn't matter what I sing.

✓ (Pragmatics, the study of language use)

"I" =  ✓

(Semantics, the study of meaning)

Homework

- Phonology: Find two French words that sound the same, but are written differently (homophones)
- Morphology: Find an example Web search query where stemming to the stem (most aggressive variant) is too aggressive.
- Semantics: Make a taxonomy of at least 5 words in a thematic domain of your choice
- Syntax: POS-tag the sentence

  The quick brown fox jumps over the lazy dog

⇒ Hand in by e-mail to f.m.suchanek@gmail.com
or on paper in the next session