CS 671 NLP
COMPUTATIONAL MORPHOSYNTAX

amitabha mukerjee
iit kanpur

Structure in language

पांच फिरंगी अफसरों __ फांसी पर ___ दिया

what can go in the blanks?

Is one of the slots easier to fill than the other?

MorphoSyntax

Sentences are built from “words”.

Boy[s] like[ϕ] girl[s]
german[s] drink[ϕ] beer[ϕ]

sentence = noun verb noun
N-pl = N + [-s]

Words are built from morphemes

ENTROPY

Structure in language

पांच फि रं गी अफ स रों को फां सी पर लट का दि या

Which syllables follow which others?

Shannon Entropy

- Predict the next word/ letter / syllable, given (*n-1)* previous letters or words : Fn = entropy = SUM_i (p_i log p_i)

Claude E. Shannon. “Prediction and Entropy of Printed English”, 1951.

Shannon Entropy : Human

- Ask human to guess the next letter:

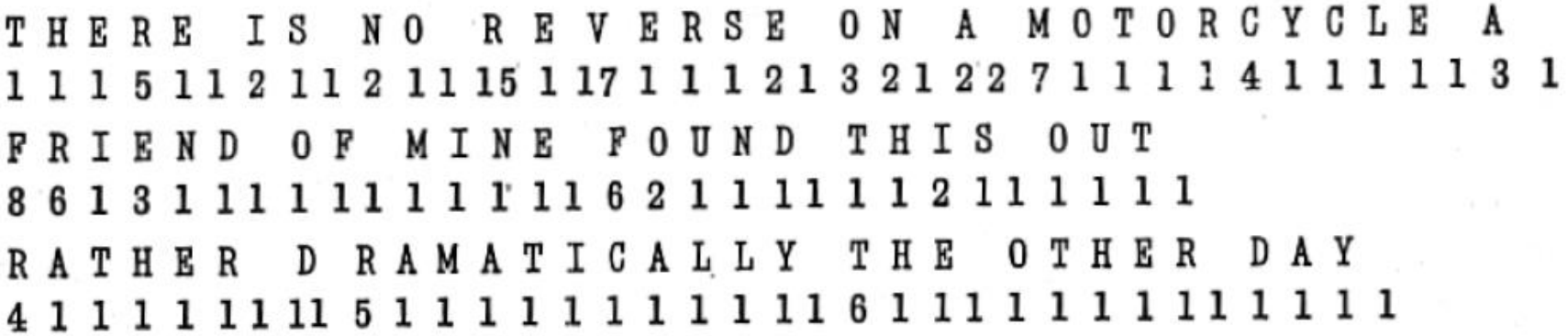
```
THE ROOM WAS NOT VERY LIGHT A SMALL OBLONG
----ROO------NOT-V-----I------SM----OBL---

READING LAMP ON THE DESK SHED GLOW ON
REA----------O------D----SHED-OLD--O-

POLISHED WOOD BUT LESS ON THE SHABBY RED CARPET
P-L-S-----O---BU--L-S—O--------SH-----RE—C-----
```

- 69% guessed on 1st attempt ["-" = 1st attempt]

Claude E. Shannon. "Prediction and Entropy of Printed English", *Bell System Technical Journal* 30:50-64. 1951.

Shannon Entropy : Human

- Count number of attempts:



- Entropy: F_1 =3.2, 4.0 F_{10} =1.0, 2.1 F_{100} = 0.6, **1.3**

Claude E. Shannon. “Prediction and Entropy of Printed English”, *Bell System Technical Journal* 30:50-64. 1951.

Shannon Entropy

- Predict the next word/ letter / syllable, given (*n-1)* previous : Fn = entropy = SUM_i (p_i log p_i)
- probabilities p_i (of n-grams) from corpus:
  - F_0 (only alphabet) = $\log_2 27$ = 4.76 bits per letter
  - F_1 (1-gram frequencies p_i) = 4.03 bits
  - F_2 (bigram frequencies) = 3.32 bits
  - F_3 (trigrams) = 3.1 bits
  - F_{word} = 2.62 bits
    (avg word entropy = 11.8 bits per 4.5 letter word)

Claude E. Shannon. “Prediction and Entropy of Printed English”, 1951.

The Shannon Generation Method

- Choose a random bigram (<s>, w) according to its probability
- Now choose a random bigram (w, x) according to its probability
- And so on until we choose </s>
- Then string the words together

```
<s> I
    I want
      want to
           to eat
              eat Chinese
                  Chinese food
                           food
  </s>
I want to eat Chinese food
```

Shannon generation: English

1. Zero-order
   - XFOML RXKHR JFF JU J ZLPWCFWKCY JFFJEYVKCQSGXYD QI'AAMKBZAACIBZLHJQD
2. First-order (unigram frequencies as English)
   - OCR0 HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI ALHENH'ITPA OOBTTVA NAH BRL
3. Second-order (bigram).
   - ON IE ANTSOUTINYS ARE T INCTORE ST BE S DEAMY ACHIN D ILONASIVE TUCOOWE AT TEASONARE FUSO TIZIN ANDY TOBE SEACE CTISBE

Shannon generation: English

- 4. Third-order (trigram)
  - IN NO IST LAT WHEY CRATICT FROURE BIRS GROCID PONDENOME OF DEMONSTURES OF THE REPTAGIN IS REGOACTIONA OF CRE

Shannon generation: English

- A. Word models: First-Order
- REPRESENTING AND SPEEDILY IS AN GOOD APT OR COME CAN DIFFERENT NATURAL HERE HE THE A IN CAME THE TO OF TO EXPERT GRAY COME TO FURNISHES THE LINE MESSAGE HAD BE THESE

- B. Word Model: Second-Order (bigram)
- THE HEAD AND IN FRONTAL ATTACK ON AN ENGLISH WRITER THAT THE CHARACTER OF THIS POINT IS THEREFORE ANOTHER METHOD FOR THE LETTERS THAT THE TIME OF WHO EVER TOLD THE PROBLEM FOR AN UNEXPECTED T

Claude E. Shannon. *A Mathematical Theory of Communication*, 1948.

The Corpus matters

- What corpus was used to generate these:

Bigram

What means, sir. I confess she? then all sorts, he is trim, captain.

Why dost stand forth thy canopy, forsooth; he is this palpable hit the King Henry. Live king. Follow.

What we, hath got so she that I rest and sent to scold and nature bankrupt, nor the first gentleman?

Trigram

Sweet prince, Falstaff shall die. Harry of Monmouth's grave.

This shall forbid it should be branded, if renown made it empty.

Indeed the duke; and had a very good friend.

Fly, and will rid me these news of price. Therefore the sadness of parting, as they say, 'tis done.

Quadrigram

King Henry.What! I will go seek the traitor Gloucester. Exeunt some of the watch. A great banquet serv'd in;

Will you not tell me who I am?

It cannot be but so.

Indeed the short and the long. Marry, 'tis a noble Lepidus.

The Corpus matters

- A more modern corpus (WSJ)

Bigram

Last December through the way to preserve the Hudson corporation N. B. E. C. Taylor would seem to complete the major central planners one point five percent of U. S. E. has already old M. X. corporation of living on information such as more frequently fishing to keep her

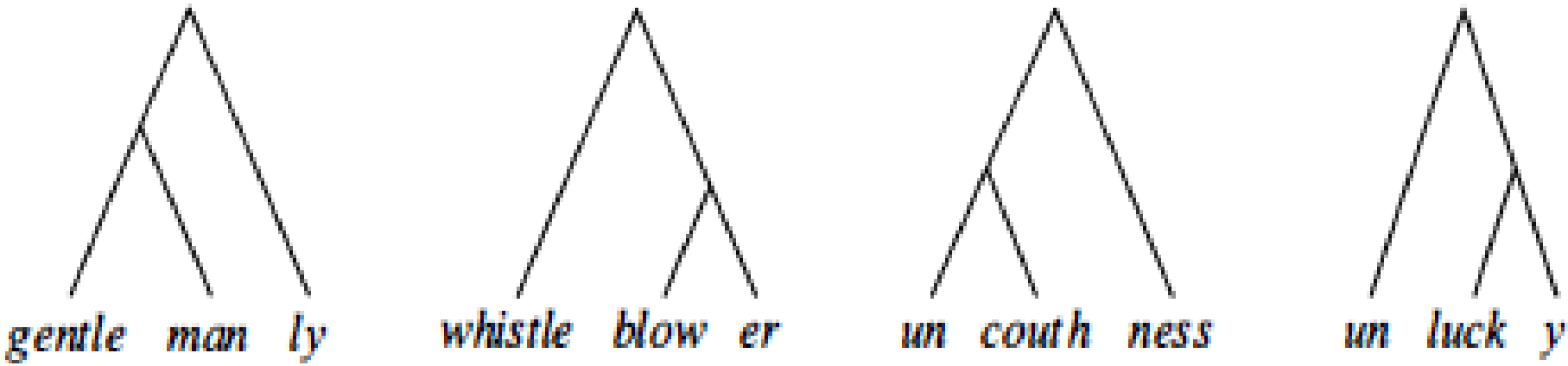
Trigram

They also point to ninety nine point six billion dollars from two hundred four oh six three percent of the rates of interest stores as Mexico and Brazil on market conditions

FINITE STATE MORPHOLOGY

(RULE-BASED)

Derivations : Parsing



- Differing parses → different semantics :
- e.g. unlockable

  “can’t be locked” or “can be unlocked”?

Huddleston & Pullum 05

Two challenges

- Morphotactics
  - Words are composed of smaller elements that must be combined in a certain order:
    - **piti-less-ness** is English
    - **piti-ness-less** is not English

- Phonological alternations
  - The shape of an element may vary depending on the context
    - **pity** is realized as **piti** in **pitilessness**
    - **die** becomes **dy** in **dying**

Morphology is regular (=rational)

- The relation between the **surface forms** of a language and the corresponding **lexical forms** can be described as a **regular relation**.
- A regular relation consists of ordered pairs of strings.
  - *leaf+N+Pl* : *leaves* *hang+V+Past : hung*
- Any finite collection of such pairs is a regular relation.
- *Regular relations are **closed** under operations such as **concatenation**, **iteration**, **union**, and **composition**.*
  - *Complex regular relations can be derived from simple relations.*

Morphology is finite-state

- A regular relation can be defined using the metalanguage of **regular expressions**.

- `[{talk} | {walk} | {work}]`
- `[%+Base:0 | %+SgGen3:s | %+Progr:{ing} | %+Past:{ed}];`

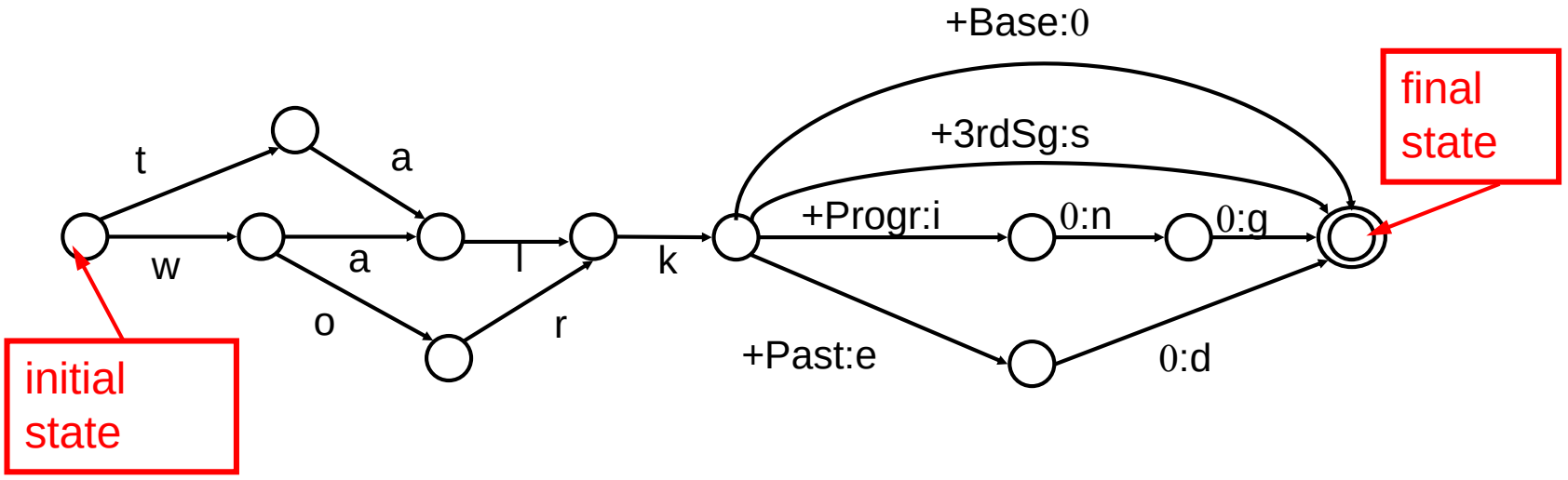
- A regular expression can be compiled into a **finite-state transducer** that implements the relation computationally.

Compilation

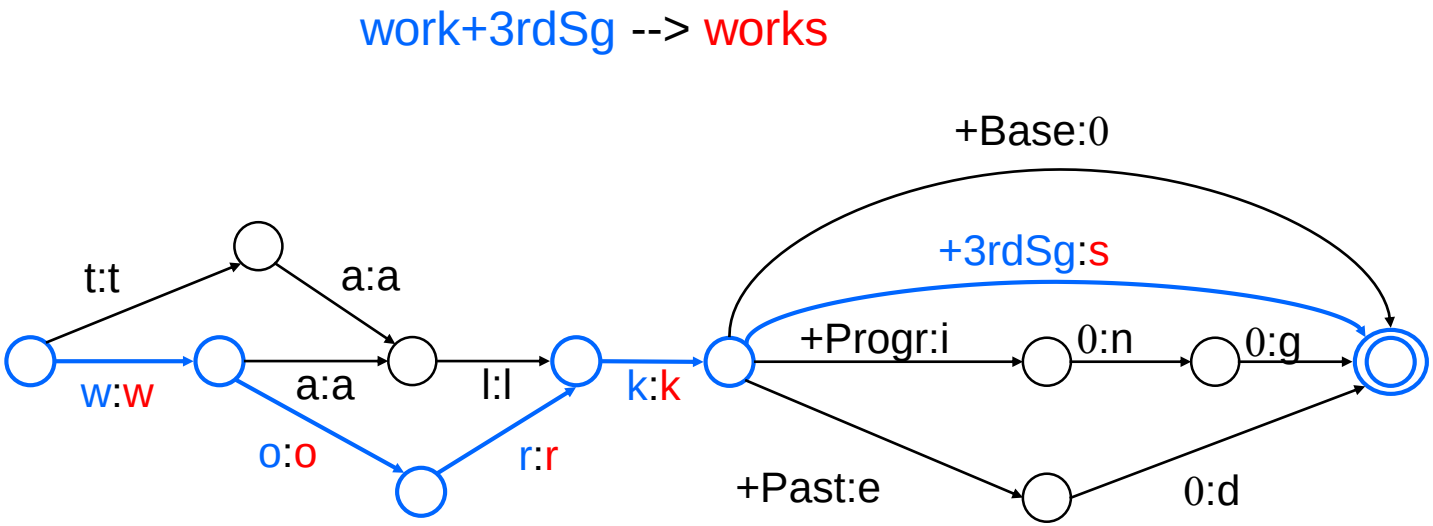
Regular expression

- `[{talk} | {walk} | {work}]`
- `[%+Base:0 | %+SgGen3:s | %+Progr:{ing} | %+Past:{ed}];`

Finite-state transducer



Generation


work+3rdSg --> works
+Base:0
+3rdSg:s
t:t
a:a
w:w
a:a
l:l
k:k
+Progr:i
0:n
0:g
o:o
r:r
+Past:e
0:d

Statistical Morphosyntax

Language Modeling

- Examine short sequences of
  - letters
  - syllables
  - morphemes
  - words
- How likely is each sequence?
- **Markov Assumption** – word is affected only by its "prior local context" (last few words)

Probabilistic Language Models

- Probability of a sequence of words:

$$P(W) = P(w_1, w_2, ..., w_{t-1}, w_T)$$

- **Conditional probability** of an upcoming word:

$$P(w_T | w_1, w_2, ..., w_{t-1})$$

- Chain rule of probability:

$$P(w_1, w_2, ..., w_{t-1}, w_T) = P(w_1)P(w_2 | w_1)P(w_3 | w_1, w_2)...P(w_T | w_1, w_2, .., w_T)$$

$$P(w_1, w_2, ..., w_{t-1}, w_T) = \prod_{t=1}^{T} P(w_t | w_1, w_2, ..., w_{t-1})$$

- $(n\text{-}1)^{\text{th}}$ order Markov assumption

$$P(w_1, w_2, ..., w_{t-1}, w_T) \approx \prod_{t=1}^{T} P(w_t | w_{t-n+1}, w_{t-n+2}, ..., w_{t-1})$$

Probabilistic Language Models

- Learn joint likelihood of training sentences under $(n\text{-}1)^{\text{th}}$ order Markov assumption using **n-grams**

$$P(w_1, w_2, \ldots, w_{t-1}, w_T) = \prod_{t=1}^{T} P(w_t \mid w_1, w_2, \ldots, w_{t-1}) \approx \prod_{t=1}^{T} P(w_t \mid \mathbf{w}_{t-n+1}^{t-1})$$

target word w_t

word history $\mathbf{w}_{t-n+1}^{t-1} = w_{t-n+1}, w_{t-n+2}, \ldots, w_{t-1}$

- Maximize the **log-likelihood**:

$$\sum_{t=1}^{T} \log P(w_t \mid \mathbf{w}_{t-n+1}^{t-1}, \boldsymbol{\theta})$$

  - Assuming a **parametric model θ**

Computational Analysis

- [Harris 1955]

  /hiyzkwikor/ *He's quicker*

  will have the segmentation: /hiy.z.kwik.or/;

  → To be done "purely by comparing this phonemic sequence with the phonemic sequences of other utterances."

- [Keshava Pitler 06] : Based on transition frequencies – How many starting syllables are *un-?*
  - Best results for English - *2006 PASCAL challenge*

Computational Analysis

- [Goldsmith 01]

  Information-Theoretic ideas - Minimum Description Length

  Which “signature” (pattern) will results in the most compact description of the corpus?

Signature	Example	Stem # (type)	Counts Token
NULL.ed.ing	betray betrayed betraying	69	864
NULL.ed.ing.s	remain remained remaining remains	14	516
NULL.s.	cow cows	253	3414
e.ed.es.ing	notice noticed notices noticing	4 62	

Computational Analysis

- [Dasgupta & V.Ng 07]
  - Simple concatenation not enough for more agglutinated languages.
  - Attempt to discover root word form. (*denial →deny*)
  - Assumption: if compound word is common,then root word will also : Word-Root Frequency Ratios (WRFR)

Correct Parses			Incorrect Parses		
Word	**Root**	**WRFR**	**Word**	**Root**	**WRFR**
bear-able	bear	0.01	candid-ate	candid	53.6
attend-ance	attend	0.24	medic-al	medic	483.9
arrest-ing	arrest	0.06	prim-ary	prim	327.4
sub-group	group	0.0002	ac-cord	cord	24.0
re-cycle	cycle	0.028	ad-diction	diction	52.7
un-settle	settle	0.018	de-crease	crease	20.7

Computational Analysis

- [Dasgupta & V.Ng 07]

	English				Bengali			
	A	P	R	F	A	P	R	F
Linguistica	68.9	84.8	75.7	80.0	36.3	58.2	63.3	60.6
Morphessor	64.9	69.6	85.3	76.6	56.5	89.7	67.4	76.9
Basic induction	68.1	79.4	82.8	81.1	57.7	79.6	81.2	80.4
Relative frequency	74.0	86.4	82.5	84.4	63.2	85.6	79.9	82.7
Suffix level similarity	74.9	88.6	82.3	85.3	66.1	89.7	78.8	83.9
Allomorph detection	**78.3**	88.3	86.4	**87.4**	**68.3**	89.3	81.3	**85.1**

+ेन्द्र or = ा + िन्द्र ?

+ेन्द्र (1575):

- राजे**न्द्र** 137 ← **राज+** 978 ? □ **राजा+** 874;
  राजनीतिक 2236, राजनीति 1537, राज्य 5532
- नरे**न्द्र** 124 ← **नर+** 41,
  नरसिंह 40, नरक 37, नर्मदा 35, नर्सिंग 31, नरूला 30
- महे**न्द्र** 88
  ← **मह+** 0
  महिला 2682, महीने 2276, महसूस 856, महंगाई 737, महतो 645
  ← **महा+** 33 महाराष्ट्र 794, महासचिव 794, महान 400, महात्मा 275, महानिदेशक 199, महाराज 182, महानगर 179
  ?? महेश 283, महोत्सव 161

• note: के**न्द्र** 680 क 164, के 261214 की 163858 को 120489

Phrase structure

Morphosyntax

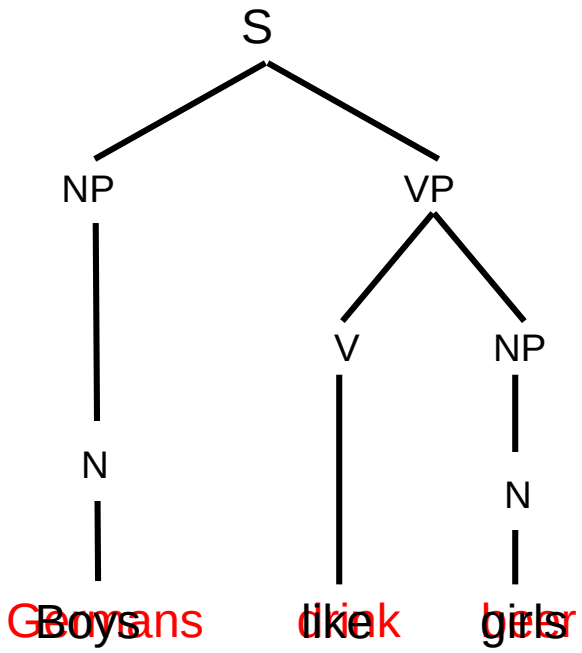
1. Break down sentence into relevant parts (**constituents**)

2. Assign **grammatical category** to constituents
   [e.g. “noun phrase”, “coordinator”]
   words → **POS (part of speech)** tags

3. Phrase structure: relation between words
   Boys like girls | A boy likes girls
   S ← NP VP ; VP ← V NP; NP ← det N | N

   verb **agreement** : (number, person) of subject

Syntactic Analysis



Phrase structure rules

S → NP VP
NP → N
VP → V NP
NP → det N

Lexicon

N → german[s], boy[s], girl[s], beer
V → like, drink

Hierarchy in Grammar

discourse — Germans drink beer. They love it.

sentence — [$_S$ Germans drink beer]

clause — S → NP VP

phrase — [$_S$ [$_{NP}$ Germans] [$_{VP}$ drink beer]]

NP → N

word — [$_S$ [$_{NP}$ [$_N$ Germans]] [$_{VP}$ [$_V$ drink [$_{NP}$[$_N$ beer]]]]

NP → N

VP → V NP

morpheme — [$_S$ [$_{NP}$ [$_N$ [$_{pl}$ German [-s]]]] [$_{VP}$ [$_V$ [$_{pl}$ drink [-ø]]]
[$_{NP}$[$_N$ beer]]]]

Clauses and Sentences

Single-clause
Sentence: Germans drink beer

Coordination
Sentence: The snake killed the rat
and swallowed it

Subordinate
Clause: No one doubts that the rat was killed

Grammatical Function vs Grammatical Category

	Germans	like beer
function:	subject	predicate
category:	NP	VP

function: relation with other parts (subject of a clause)

category: grammatically similar expressions

Grammatical Function vs Grammatical Category

Germans is the subject of the clause

Germans like beer

Subject : w.r.t. a clause (not just subject)

Noun Phrase: is a category - may have different functions

Grammatical Function vs Grammatical Category

Same function, different categories:

[His guilt] was obvious. [NP]

[That he was guilty] was obvious.

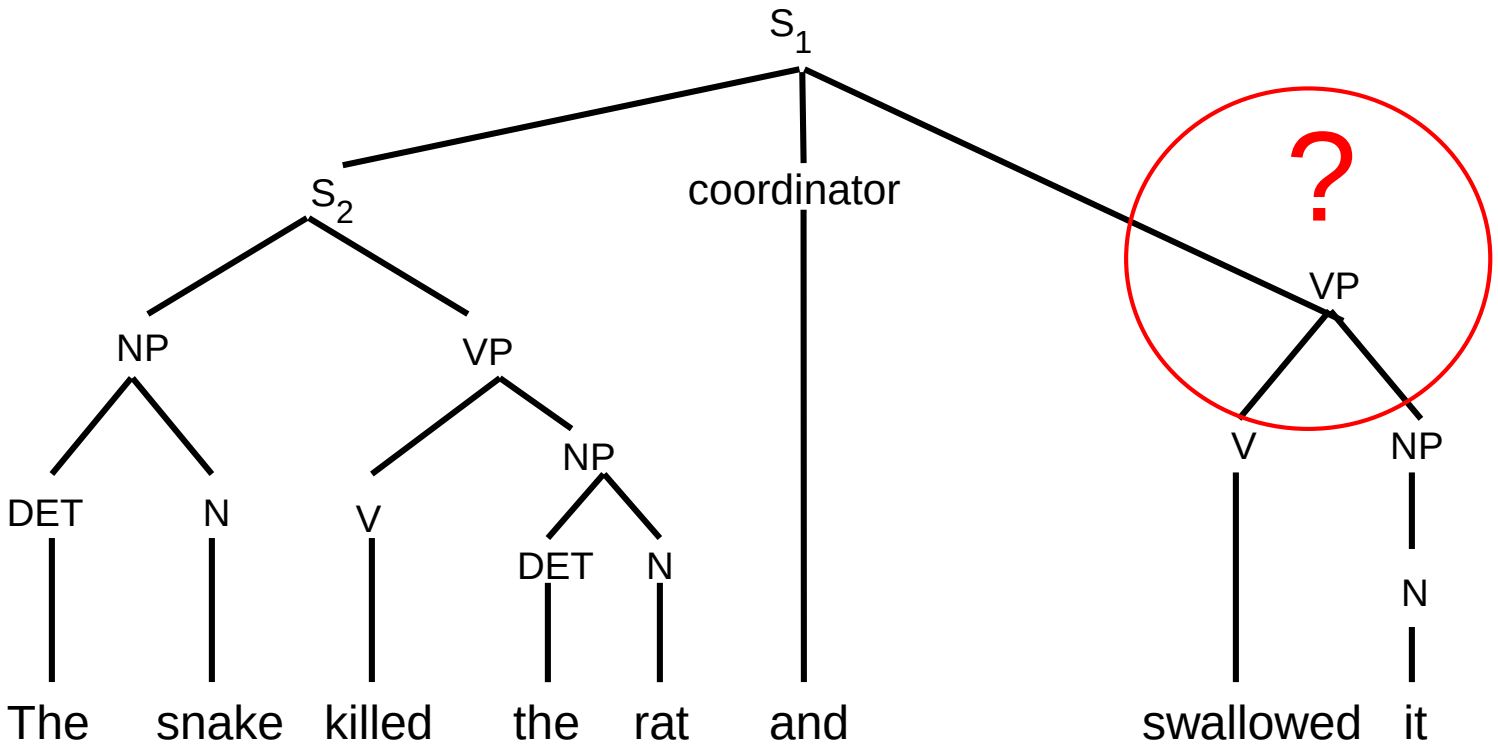
[Subordinate clause, with own subj/pred]

Same category, different functions:

[Some customers] complained. [subject]

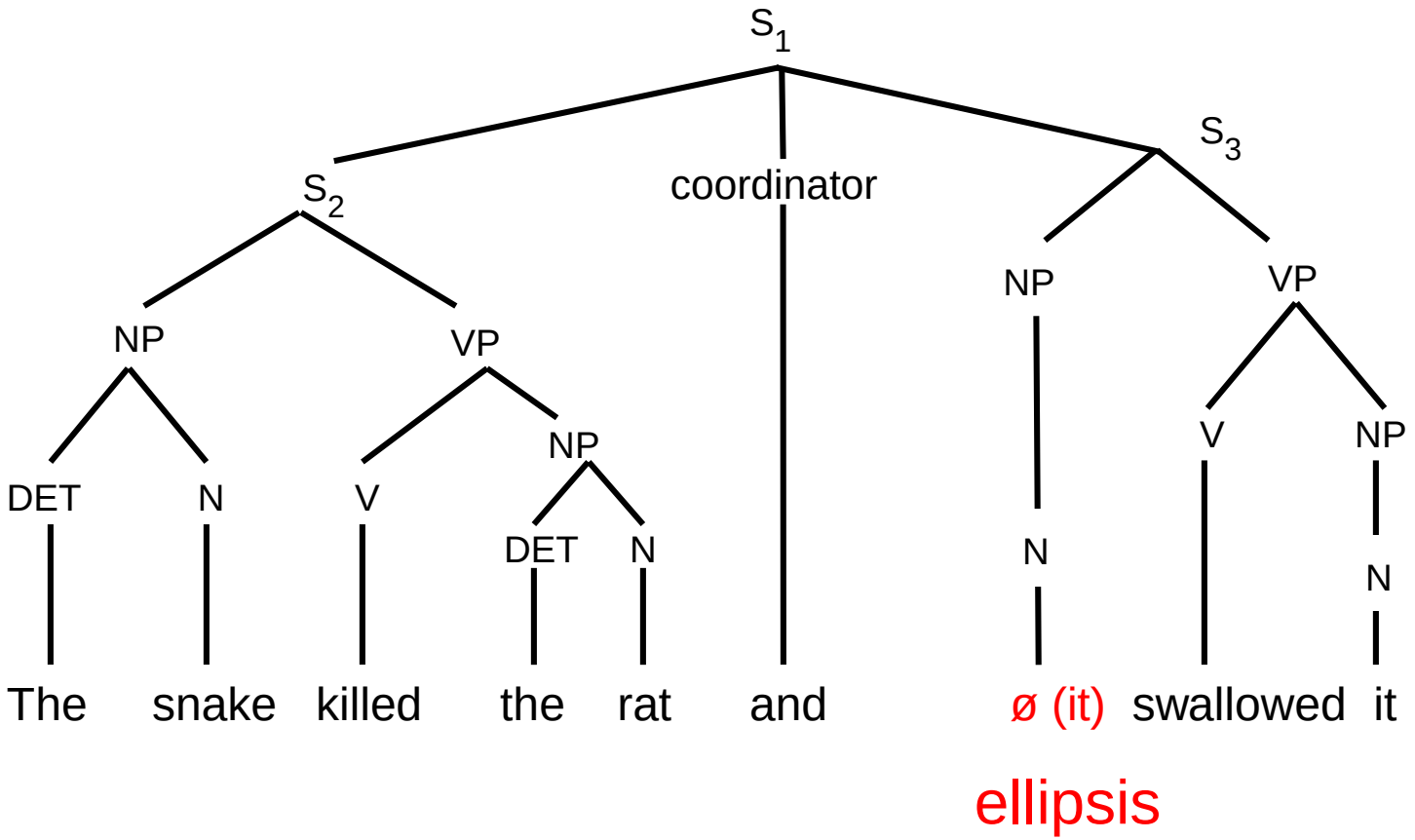
Kim insulted [some customers] [object]

Missing Elements?



[haegeman wekker 03] modern course in english syntax

Missing Elements : Ellipsis



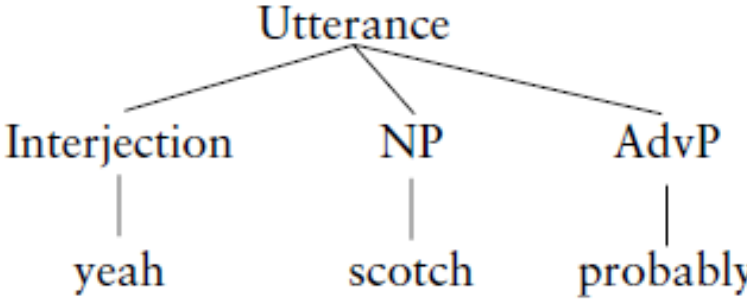
[haegeman wekker 03] modern course in english syntax

Bare argument ellipsis (BAE)

A: I hear Harriet's been drinking again.
B: *Yeah, scotch, probably*

Generative Grammar analysis (ellipsis):
B: *Yeah, [Harriet has been drinking] scotch probably*
[$_{ADVP}$ *Yeah*] [$_{NP}$ e] [$_{VP}$ e *scotch*]] [$_{ADVP}$ *probably*]

Culicover / Jackendoff 02*:*
Accept fragment as is
use semantics / pragmatics
to judge grammaticality



Ellipsis Ambiguity

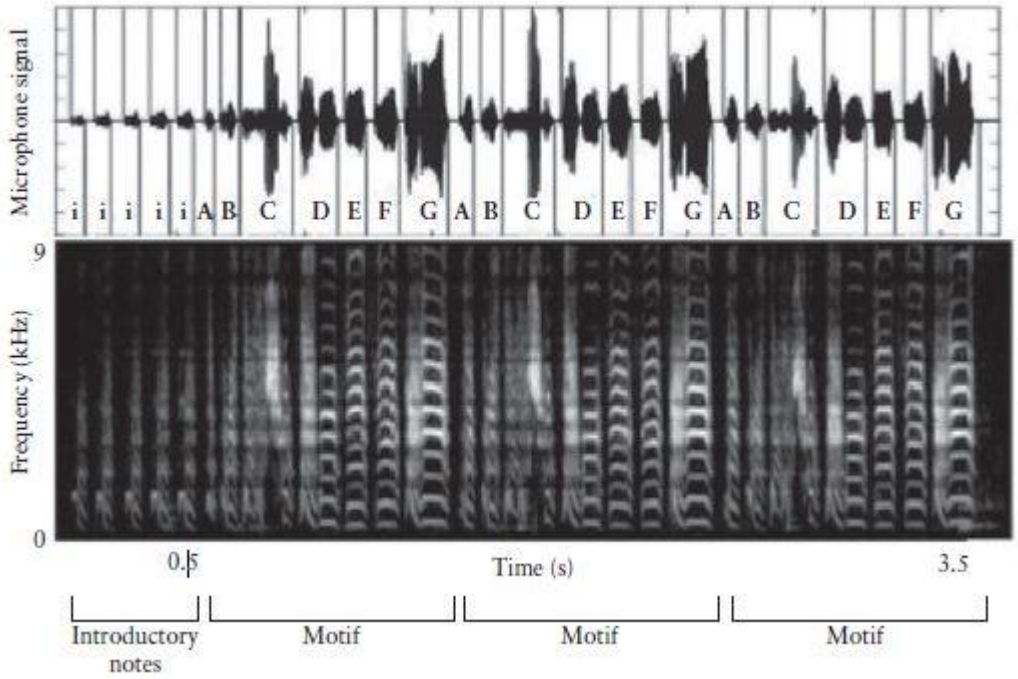
Q: Should I have a baby after 35?

A: No. 35 children is enough.

Semantics – Syntax – Pragmatics divide

- CARNAPIAN division of the theory of language:
  - SYNTAX - relations between expressions
  - SEMANTICS - relations between expressions and what they stand for
  - PRAGMATICS - relations between expressions and those who use them
- [Peregrin 1998, The pragmatization of semantics] :
  - Internal Challenge: context – Deictic (pronouns, demonstratives); indef article “a” = introduces new element ; “the” = old item
  - External Challenge: **language is not a set of labels stuck on things**; not "what does a word mean?" but "how is it used?“ [Wittgenstein PI 53]
- Langacker : Composition based on Syntax + Semantics + Pragmatics

Zebra finch song



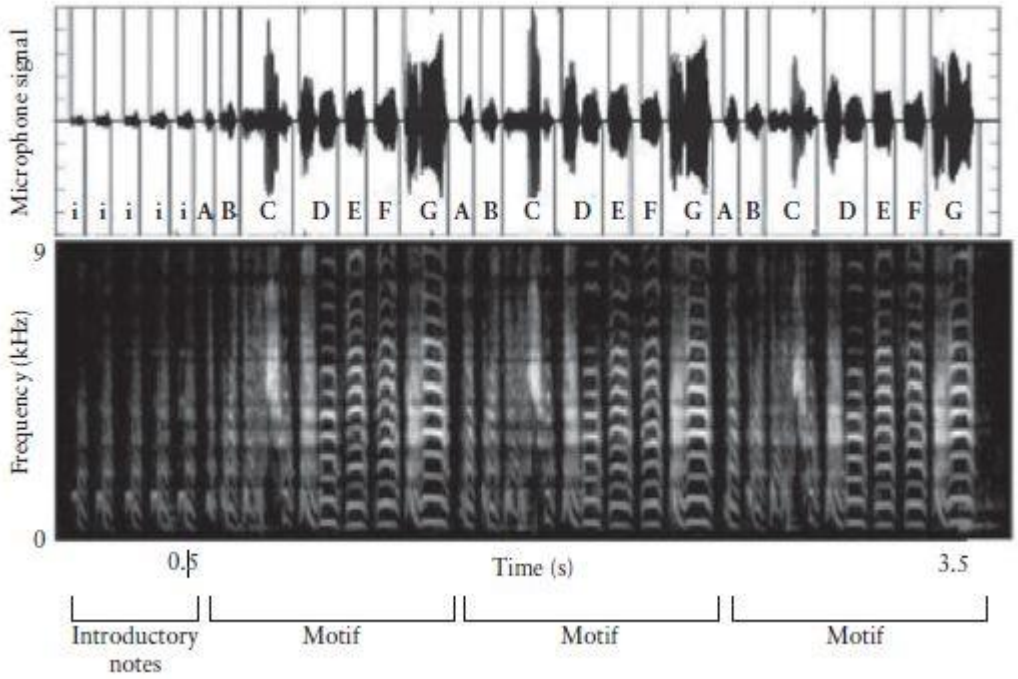
www.youtube.com : zebra finch song

initial notes - "i" - repeated a few times

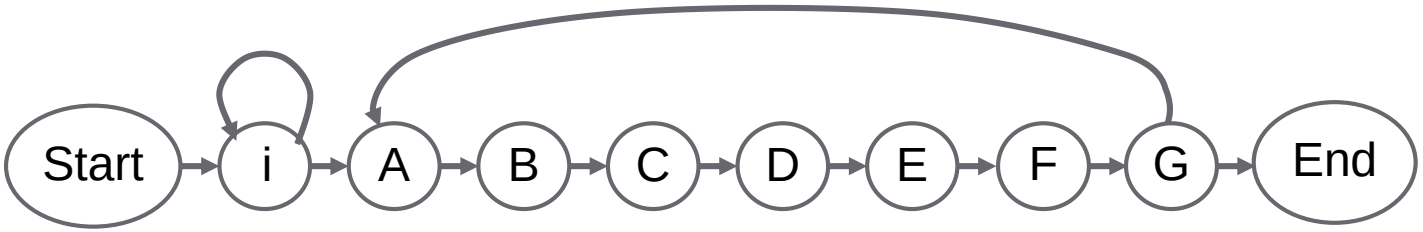
motif of syllables - ABCDEFG - repeated variable # of times.

[hurford 12] origins of grammar

Regular Grammar?



www.youtube.com : zebra finch song



[hurford 12] origins of grammar

APPROACHES TO NLP PROBLEMS

Approaches

Word segmentation:

- Chinese: 浮法像蝴蝶.

  ("float like a butterfly)

- Hindi

  पांचफिरंगीअफसरोंकोफांसीपरलटकादिया

  - Q. Letter-or Syllable- based?
  - Which positions have low "sequence" probability?

NLP Tasks

- Rule-based
  - Discrete categories (Boolean)
- Stochastic
  - Based on discrete structures (e.g. PCFG)
  - Discovery of structures
- Cognitive
  - Unsupervised, but needs semantic models

NLP tasks and Probabilistic Models

- Machine Translation:
  - P(**high** winds tonite) > P(**large** winds tonite)
- Spell Correction
  - The office is about fifteen **minuets** from my house
    - P(about fifteen **minutes** from) > P(about fifteen **minuets** from)
- Speech Recognition
  - P(I saw a van) >> P(eyes awe of an)

NLP tasks and Probabilistic Models

- Verb argument structure discovery
  - Via factorization of syntactic parses to discover
  - Argument structure (syntax ?)
  - Selection preference (semantics)
- Summarization, question-answering, etc.,
  - Paraphrasing
- Semantics : Role labelling, Similarity

Word similarity : plagiarism detection

MAINFRAMES

Mainframes are primarily referred to large computers with rapid, advanced processing capabilities that can execute and perform tasks equivalent to many Personal Computers (PCs) machines networked together. It is characterized with high quantity Random Access Memory (RAM), very large secondary storage devices, and high-speed processors to cater for the needs of the computers under its service.

Consisting of advanced components, mainframes have the capability of running multiple large applications required by many and most enterprises and organizations. This is one of its advantages. Mainframes are also suitable to cater for those applications (programs) or files that are of very high demand by its users (clients). Examples of such organizations and enterprises using mainframes are online shopping websites such as Ebay, Amazon, and computing-giant

MAINFRAMES

Mainframes usually are referred those computers with fast, advanced processing capabilities that could perform by itself tasks that may require a lot of Personal Computers (PC) Machines. Usually mainframes would have lots of RAMs, very large secondary storage devices, and very fast processors to cater for the needs of those computers under its service.

Due to the advanced components mainframes have, these computers have the capability of running multiple large applications required by most enterprises, which is one of its advantage. Mainframes are also suitable to cater for those applications or files that are of very large demand by its users (clients). Examples of these include the large online shopping websites -i.e. : Ebay, Amazon, Microsoft, etc.

from Jurafsky lectures stanford 2015

Word Sense Disambiguation

- For example, with Google translate
  http://laylita.com/recetas/2008/02/28/platanos-maduros-fritos/

A veces siento que no como suficiente plátanos maduros fritos, quizás es porque los comía casi todos los días cuando vivía en Ecuador.

Sometimes I feel like not enough fried plantains, perhaps because he ate almost every day when I lived in Ecuador.

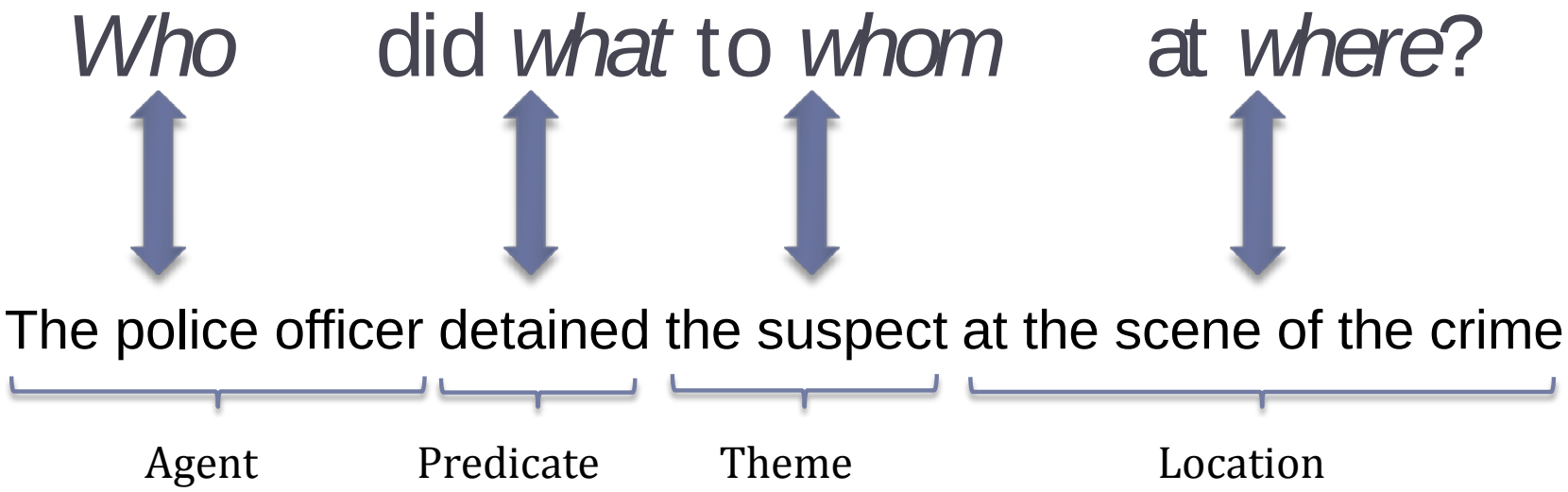
como: “like”, como : “I eat”

Question Answering

"Analysts have been expecting a GM-Jaguar pact that would give the U.S. car maker an eventual 30% stake in the British company."

- How do we answer questions about who did what to whom?

Semantic Role Labeling

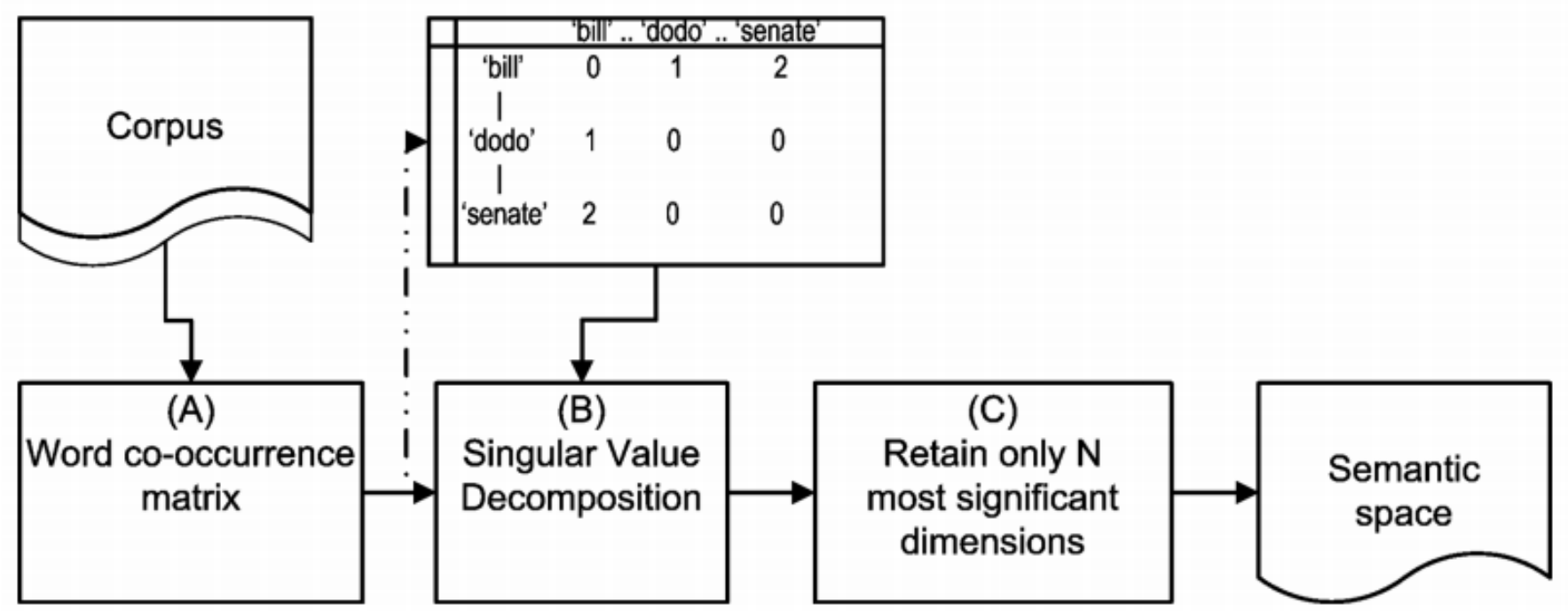


DISTRIBUTIONAL LANGUAGE MODELS

Distributional Hypothesis

- Bhartrihari (6th c.) : Words by themselves may have no meaning –
  meaning = contexts of use (holism)
- Wittgenstein (1953): The meaning of a word is its usage in language
- J. R. Firth (1957) : Word is known by the company it keeps (*Modes of Meaning, 1965*)
- Word meaning= set of contexts in which it may be used.

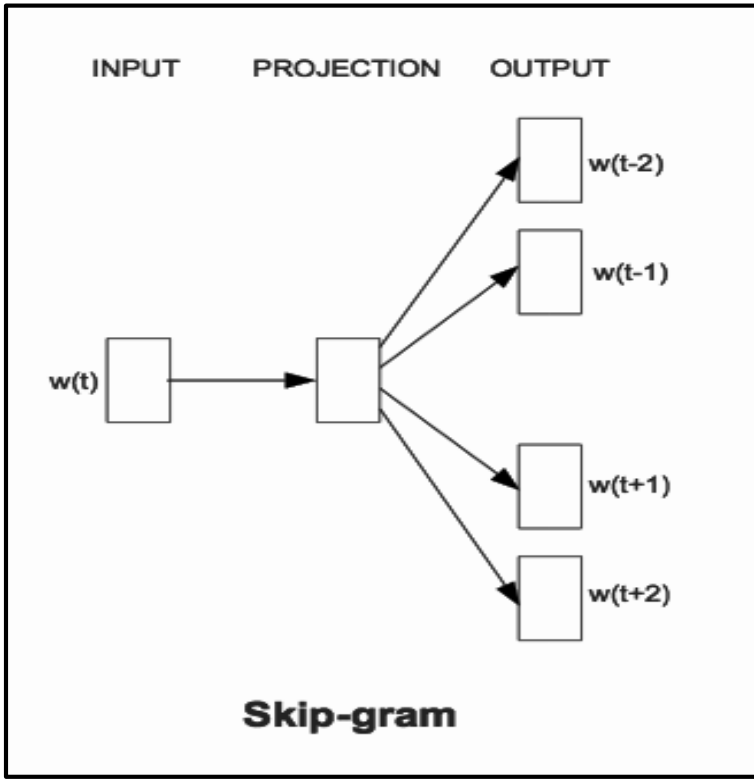
Word Vectors : WORDSPACE



sagi-diermeier-13_identifying-issue-frames-in-text

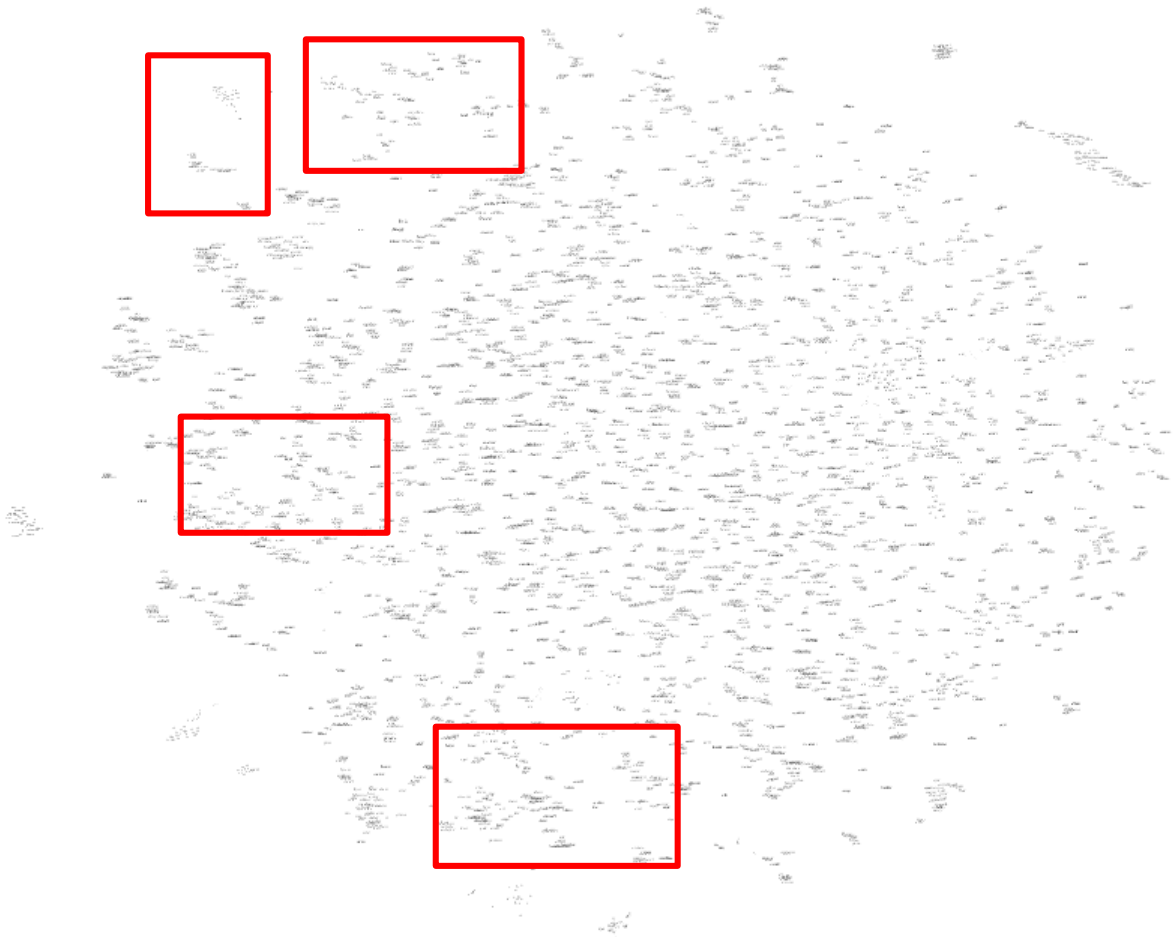
Skip-gram Model [Mikolov 13]

- No hidden layer
- Projection layer shared for all words
- All words get projected into the same position (vectors are averaged).
- Skip-gram : Given w in a phrase, attempt to predict left and right context (k words each) from projection layer.
- Efficient: Softmax replaced by Hierarchical softmax



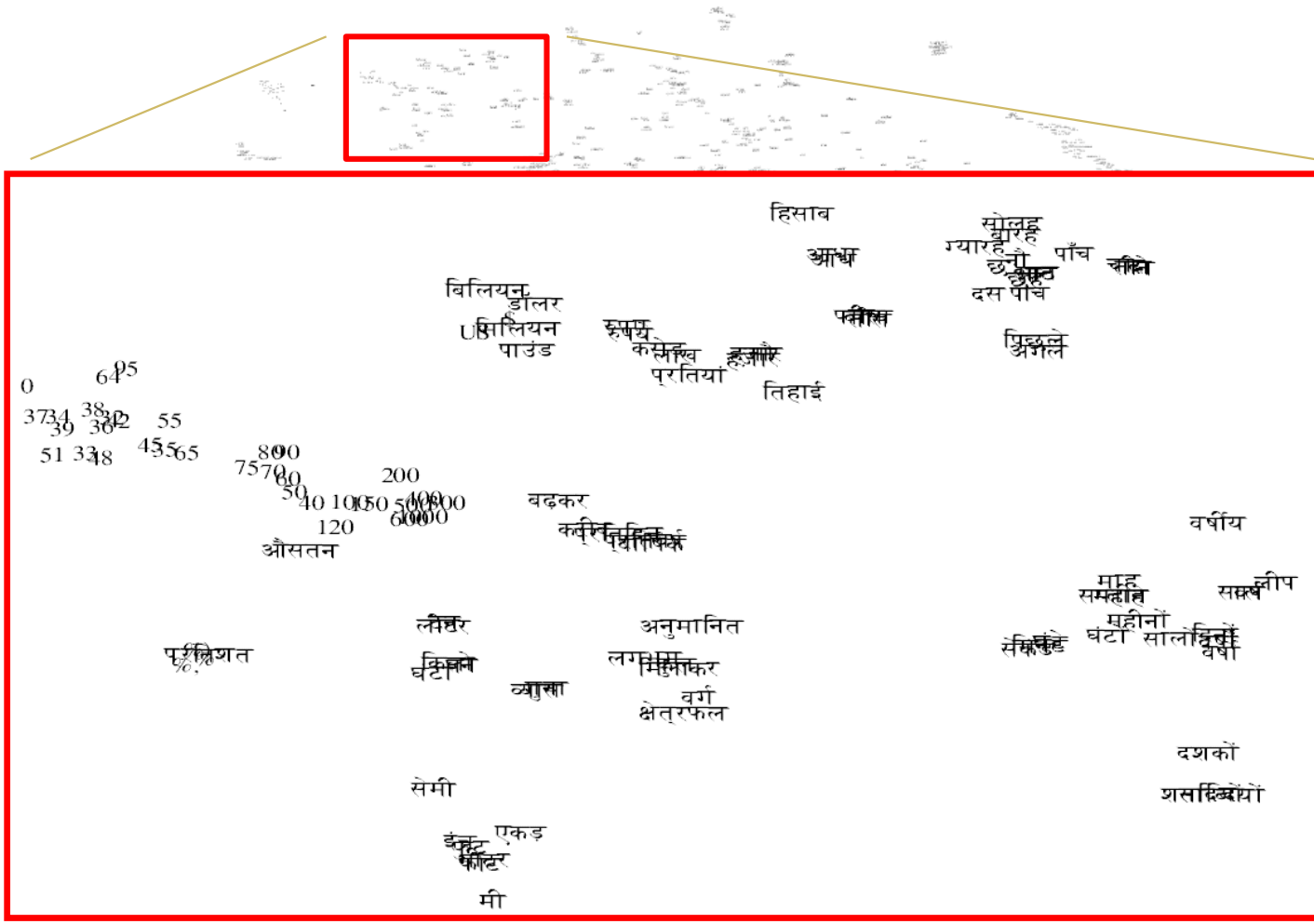
Predicts surrounding words given current word.

Word Vector Space: Hindi (top 5000)

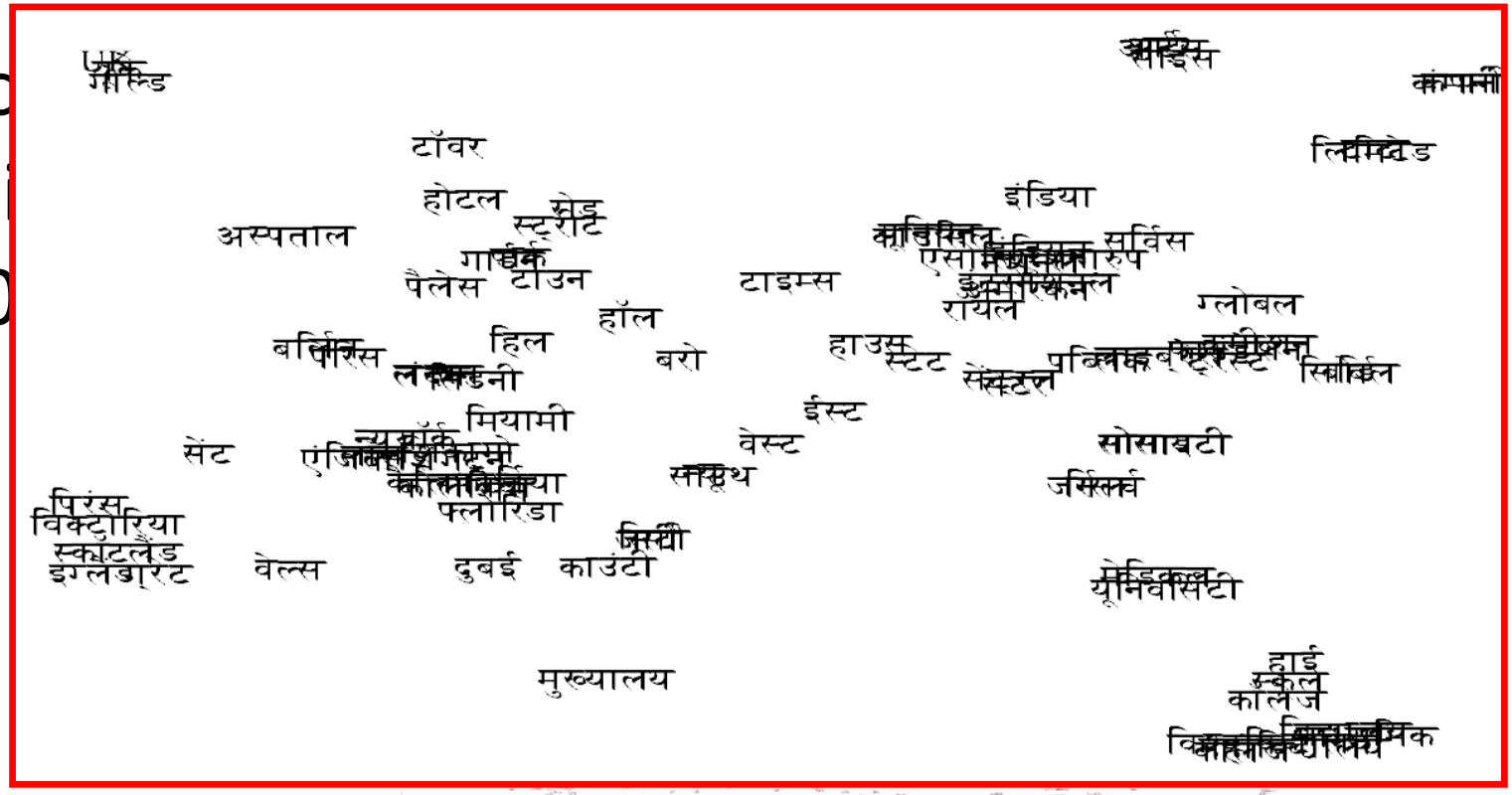


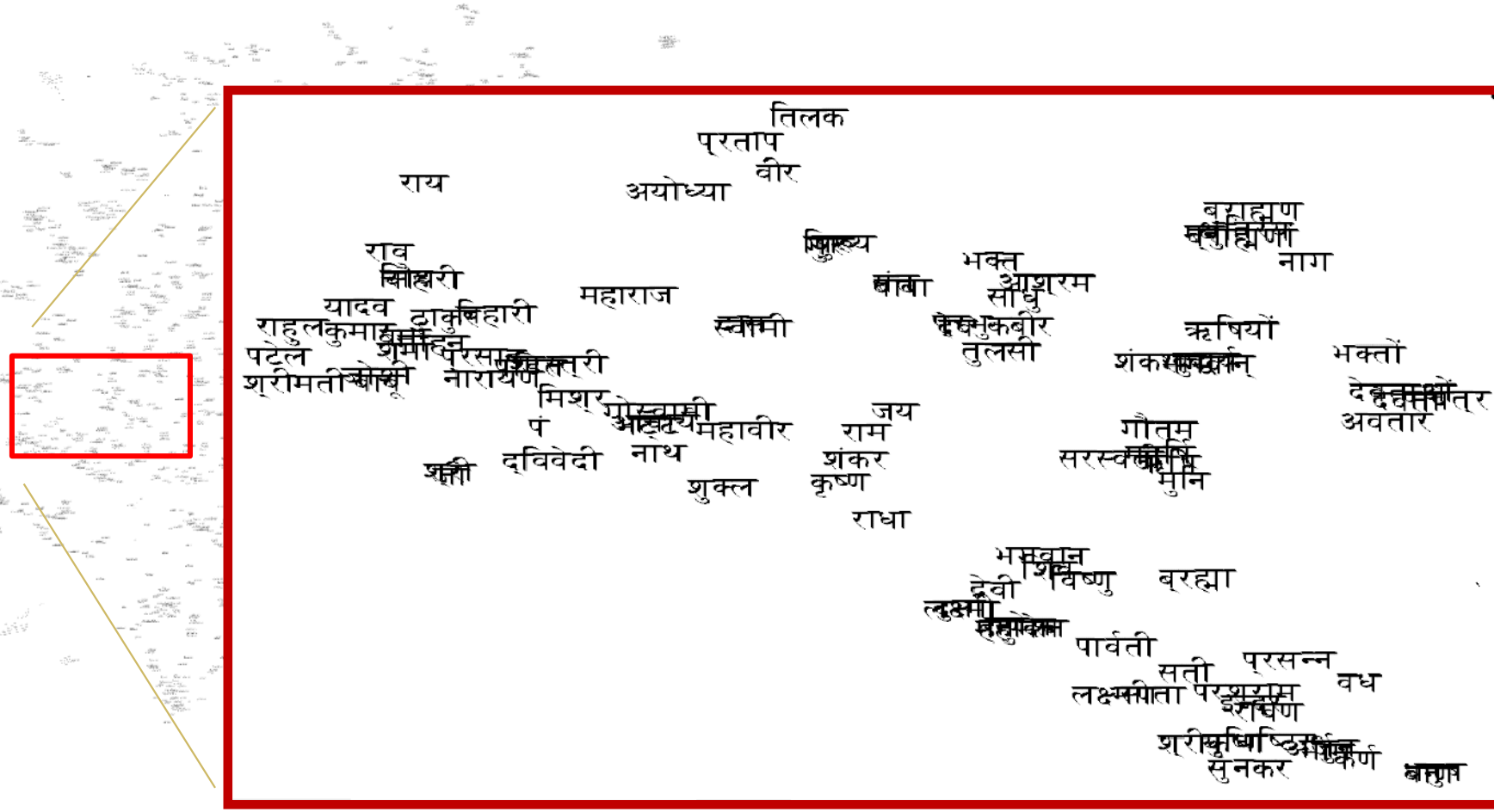
Word Vector Space: Hindi (top 5000)

Word Vector Space: Hindi (top 5000)



Word Vec
Space: Hi
(top 5000




तिलक
प्रताप
वीर
राय
अयोध्या
ब्राह्मण
नाग
राव
भक्त
आश्रम
साधु
महाराज
यादव
ठाकुर
बिहारी
स्वामी
राहुल
कुमार
पटेल
तुलसी
ऋषियों
शंकर
भक्तों
नारायण
मिश्र
जय
राम
महावीर
गौतम
अवतार
पं
नाथ
शंकर
द्विवेदी
शुक्ल
कृष्ण
मुनि
राधा
भगवान
शिव
विष्णु
ब्रह्मा
देवी
पार्वती
सती
प्रसन्न
वध
लक्ष्मी
सीता
परशुराम
सुनकर

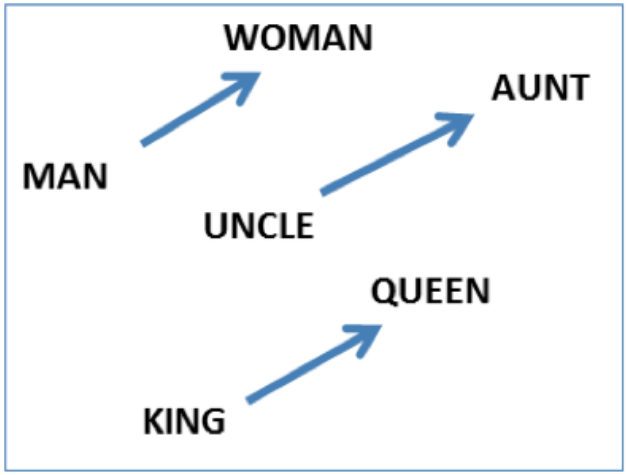
রংপুর ময়মনসিংহ যশোর
কুড়িগ্রাম কারমাইকেল

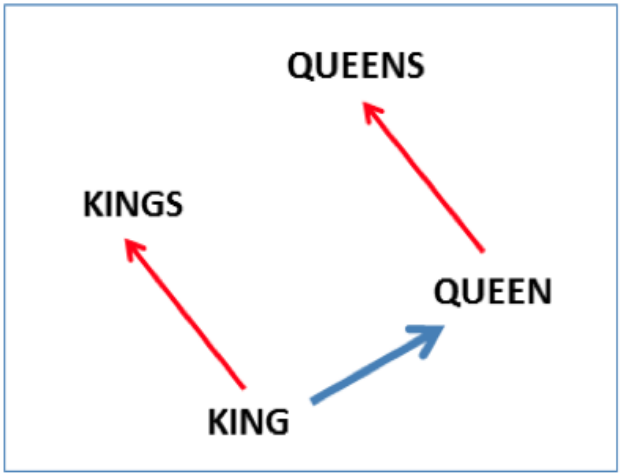
কিলোমিটার মাইল ফুট
মিটার বর্গ

লাল গোলাপি হলুদ সবুজ নীল

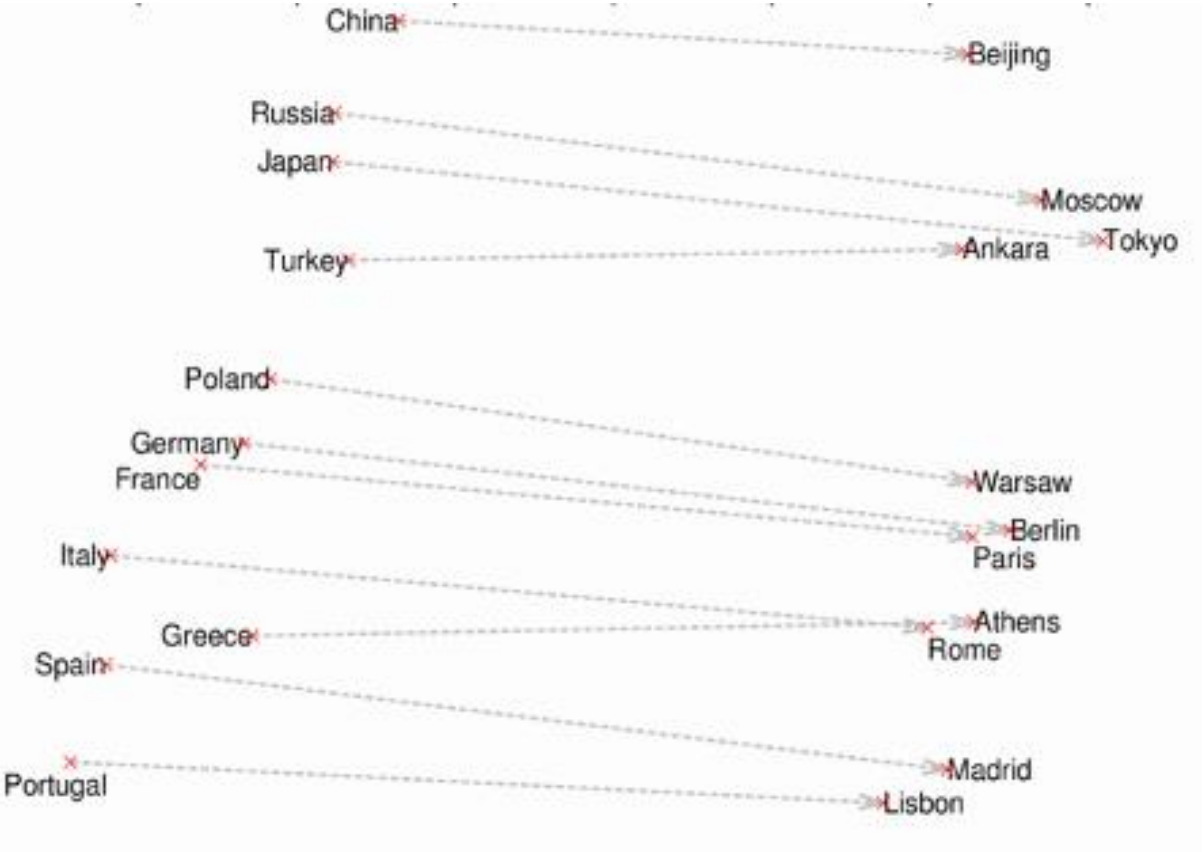
কম্পিউটার লয়্যাপটপ হার্ডওয়্যার
কম্পিউটারে সফটওয়্যার

Gender and Number Relations





Ontological Relations



AK Zehady, Purdue U

RULE-BASED SYNTAX

What is Syntax?

- *Compositionality* Assumption: Larger phrases built up from smaller ones

- Construct rules for how words compose into phrases and sentences = **Grammar**
  - may also apply to morphemes

- Map to semantics:
  - Assumption: words have meaning
  - Syntax : Composes words into new composite meaning

Why is Syntax Important?

- Grammar checkers
- Question answering
- Word sense Disambiguation

- Information retrieval (?)
- Machine translation

- Map to semantics

Theories of Syntax?

- Unfortunately, no consensus on a theory of grammar - aggressive debates :
  - Chomskyan – formalist, autonomous from semantics, we are born with syntax
  - Cognitive linguistics – semantics has a role, language is learned by discovering patterns in usage
  - Computational : Use what works

Syntax : Composability

- Are sentences constructed by combining words? [decomposability]
- Or are words obtained by breaking up sentences? [holism]
- At least some times, while learning a language, babies understand the sentence *before* the words

Chomskyan (Generative) view

- Syntax is independent of meaning. Perception, action, etc. are not relevant to grammar
- Of course, language is compositional
- Lexicon = list of words → *arbitrary*
- Syntax: Words are composed via deterministic, formal rules → *systematic*

Chomskyan Language Acquisition

- Babies acquire language with very little guidance. (Poverty of Stimulus)
- Possible only if we have an innate *Language Faculty* with a built-in *Universal Grammar* (Nativism)
- Language learning = filling language-specific parameters in the UG

Autonomous Syntax

- Are grammaticality judgments based on form alone?

colourless green ideas sleep furiously

vs

furiously sleep ideas green colorless

→ ***autonomy of syntax*** argument

[chomsky 57]: syntactic structures

Autonomous Syntax : Assumptions

- Rules determining the syntax (form) of language are formulated without reference to meaning, or language use.

- Related : **Grammar is not statistical**

  "There appears to be no particular relation between statistical relations and grammaticalness" p.17

  [chomsky 57]: syntactic structures

  see P. Norvig: On Chomsky and the Two Cultures of Statistical Learning [http://norvig.com/chomsky.html]

Ambiguity : Newspaper headlines

- Ban on Nude Dancing on Governor's Desk
- Kids Make Nutritious Snacks
- Iraqi Head Seeks Arms
- Juvenile Court to Try Shooting Defendant
- Stolen Painting Found by Tree
- Local High School Dropouts Cut in Half
- Red Tape Holds Up New Bridges

HAND-CRAFTED (RULE-BASED) GRAMMARS

Grammars for Syntax

- Syntax = systematcity in composing words
- A. Words as forms (tokens in finite alphabet)
  - Generative grammars : GBT / MP) [Chomsky]
  - HPSG (Sag and Pollard, 87, 94)
  - Categorial grammars : CCG (Steedman 87)
  - Dependency grammars (Tesniere 59, Kubler/Nivre 09)
- B. Words as Symbols / Signs = form-meaning pairs
  - Construction Grammar (Goldberg 95)
  - Cognitive Grammars (Langacker 84)

Grammar for NLP : Approaches

- Rule-based vs Machine learning / Probabilistic
  - Hand-Crafted grammar
  - Supervised: Based on annotated corpus with intermediate tags :
    - parts of speech (Brown), parse tree (Treebank),
    - semantic maps (Framenet)
  - Unsupervised : Attempt to learn syntax + semantics from grounded input (embedded in context)
    - Task driven: input → response. (No need to analyze input)

Context Free grammar

- Syntax = systematcity in composing words
- Grammar G = (V, Σ , R, S)
  - V = variables (non-terminals)
  - Σ = vocabulary (terminals)
  - R = finite relation from V to $(V \cup \Sigma)^*$
    from non-terminal to seq of terminals+non+ϕ

    S = start symbol
- Productions or rewrite rules :

  S → NP VP NP → Det N VP → V N

  NP → N VP → V

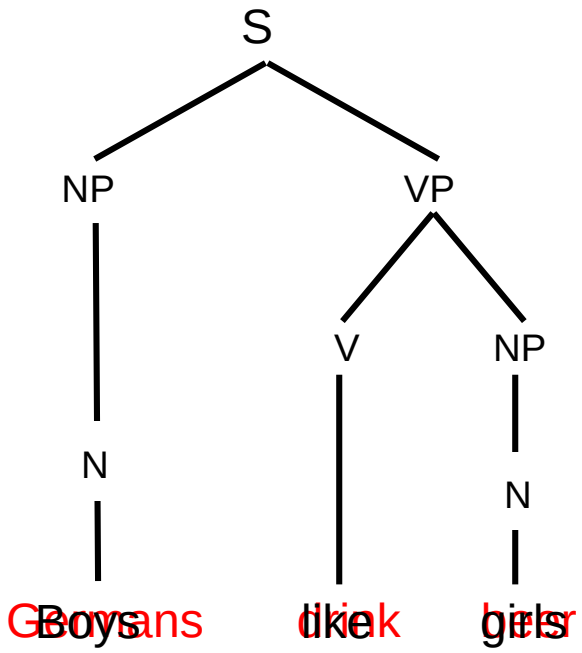
Context Free grammar

Can generate sentences:

boys like girls
germans drink beer

Sentence → NP VP
→ noun [verb noun]

Syntactic Analysis



Phrase structure rules

S → NP VP
NP → N
VP → V NP
NP → det N

Lexicon

N → german[s], boy[s], girl[s], beer
V → like, drink

Creating grammar rules

- Hand-crafted grammar and lexicon
  - S → NP VP
  - NP → (DT) NN
  - NP → NN NNS
  - NP → NNP
  - VP → V NP
  - …

  NN → *interest*
  NNS → *rates*
  NNS → *raises*
  VBP → *interest*
  VBZ → *rates*
- Proof systems : establish parses from words
- Scales poorly. Little coverage
  - Lots of parses - for real-size broad-coverage grammar: millions of parses

Probabilistic CFG

- Hand-crafted grammar and lexicon
  - S → NP VP
  - NP → (DT) NN
  - NP → NN NNS
  - NP → NNP
  - VP → V NP
  - …

  NN → *interest*

  NNS → *rates*

  NNS → *raises*

  VBP → *interest*

  VBZ → *rates*
- Proof systems : establish parses from words
- Scales poorly. Little coverage
  - Lots of parses - for real-size broad-coverage grammar: millions of parses

Probabilistic Grammar PCFG

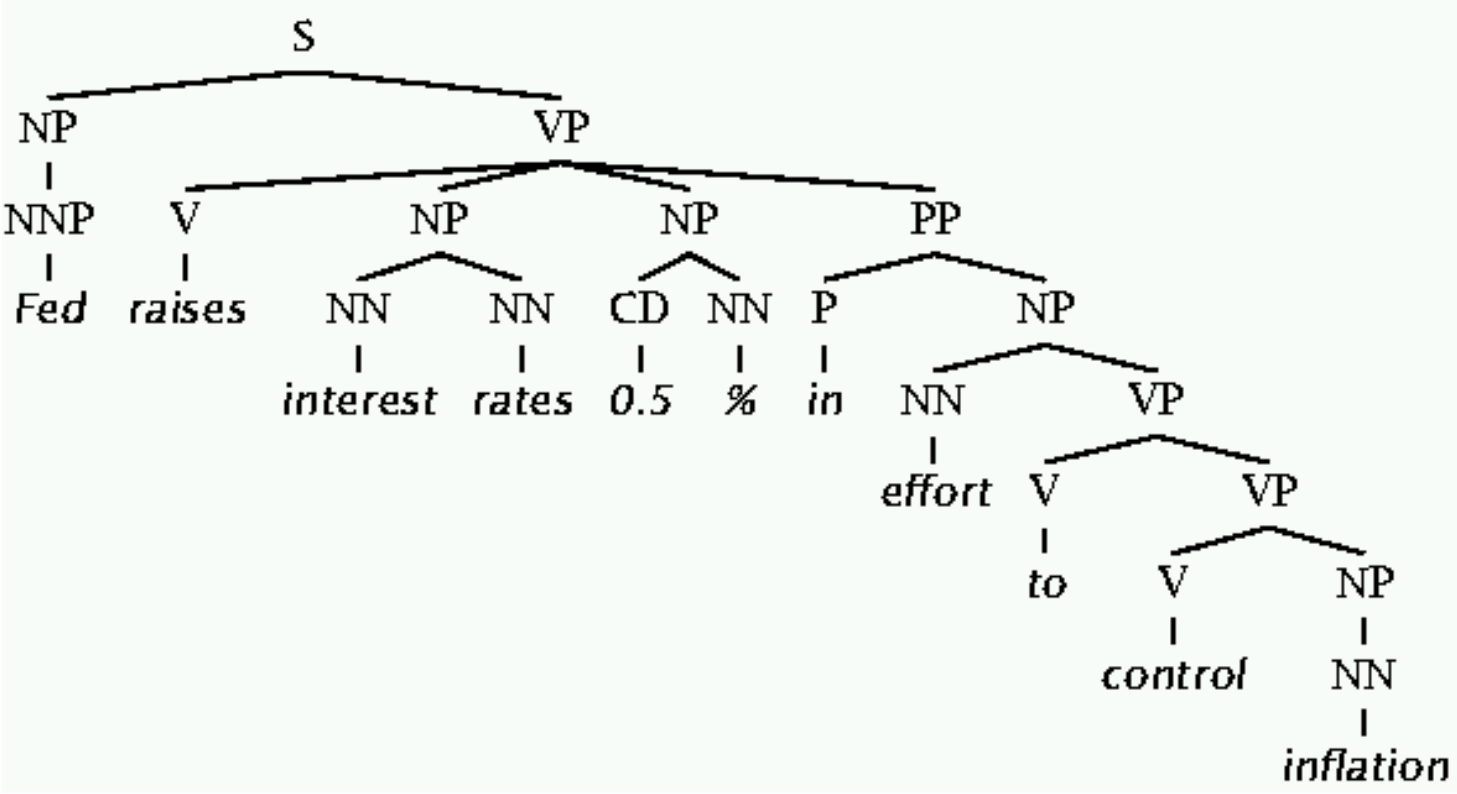
- Grammar G = (V, Σ , R, S, P)
  - R = rules e.g. NP → N N
  - P(r) = probability for each in R; Σ(r) = 1

- Top-down (matches from LHS – start from goal), vs
- Bottom-up (matches from RHS – start w data)

AMBIGUITY

Parse ambiguities

- Tree for: *Fed raises interest rates 0.5% in effort to control inflation* (*NYT* headline 5/17/00)



slide from: manning 07

Parse ambiguities

Part of speech ambiguities

		VB					
	VBZ	VBP	VBZ				
NNP	NNS	NN	NNS	CD	NN		
Fed	*raises*	*interest*	*rates*	*0.5*	*%*	*in*	*effort*
						to	*control*
							inflation

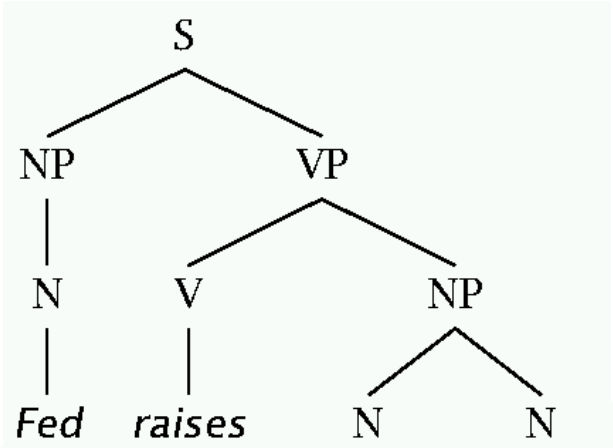
Syntactic attachment ambiguities

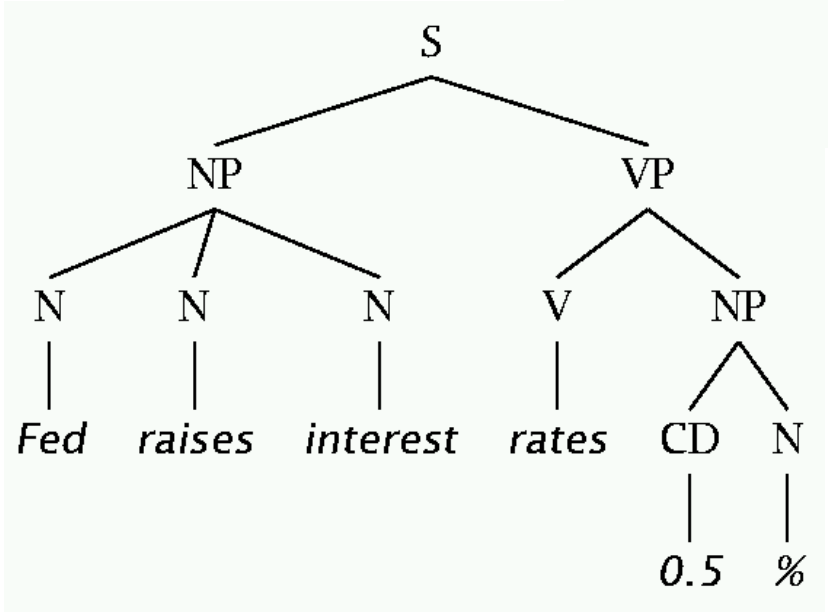
Word sense ambiguities: Fed → *"federal agent"*
interest → *a feeling of wanting to know or learn more*

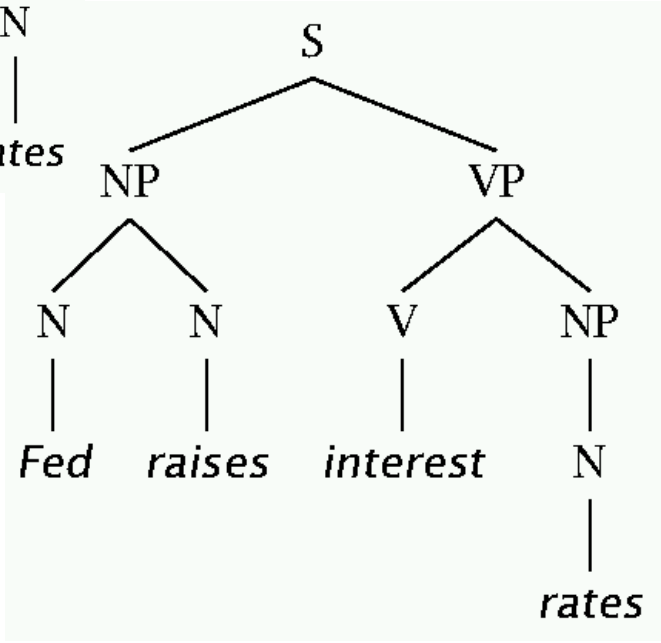
Semantic interpretation ambiguities above the word level

slide from: manning 07

V/N ambiguities







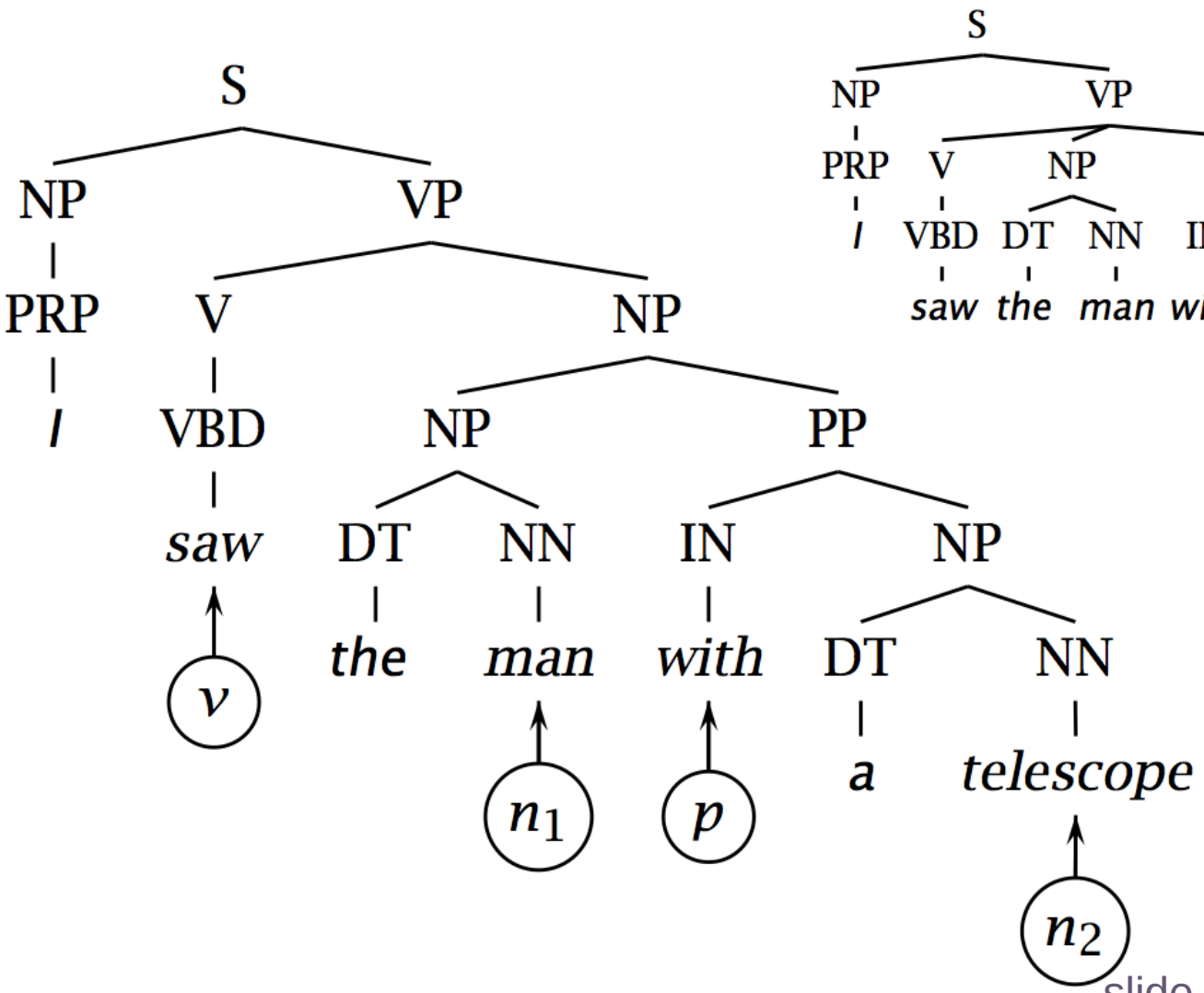
Attachment ambiguities

- Prepositional phrase attachment:

I saw the man with a telescope

- What does *with a telescope* modify?
  - The verb *saw*?
  - The noun *man*?

Attachment ambiguities: Two possible PP attachments



slide from: manning 07

Attachment ambiguities

- In the V NP PP context, right attachment usually gets right 55–67% of cases.
- → wrong 33–45% of cases.

Selectional Restriction

- Specific Words select specific attachments
  - The children ate the cake with a spoon
  - The children ate the cake with frosting
  - Moscow sent more than 100,000 soldiers into Afghanistan …
  - Sydney Water breached an agreement with NSW Health …

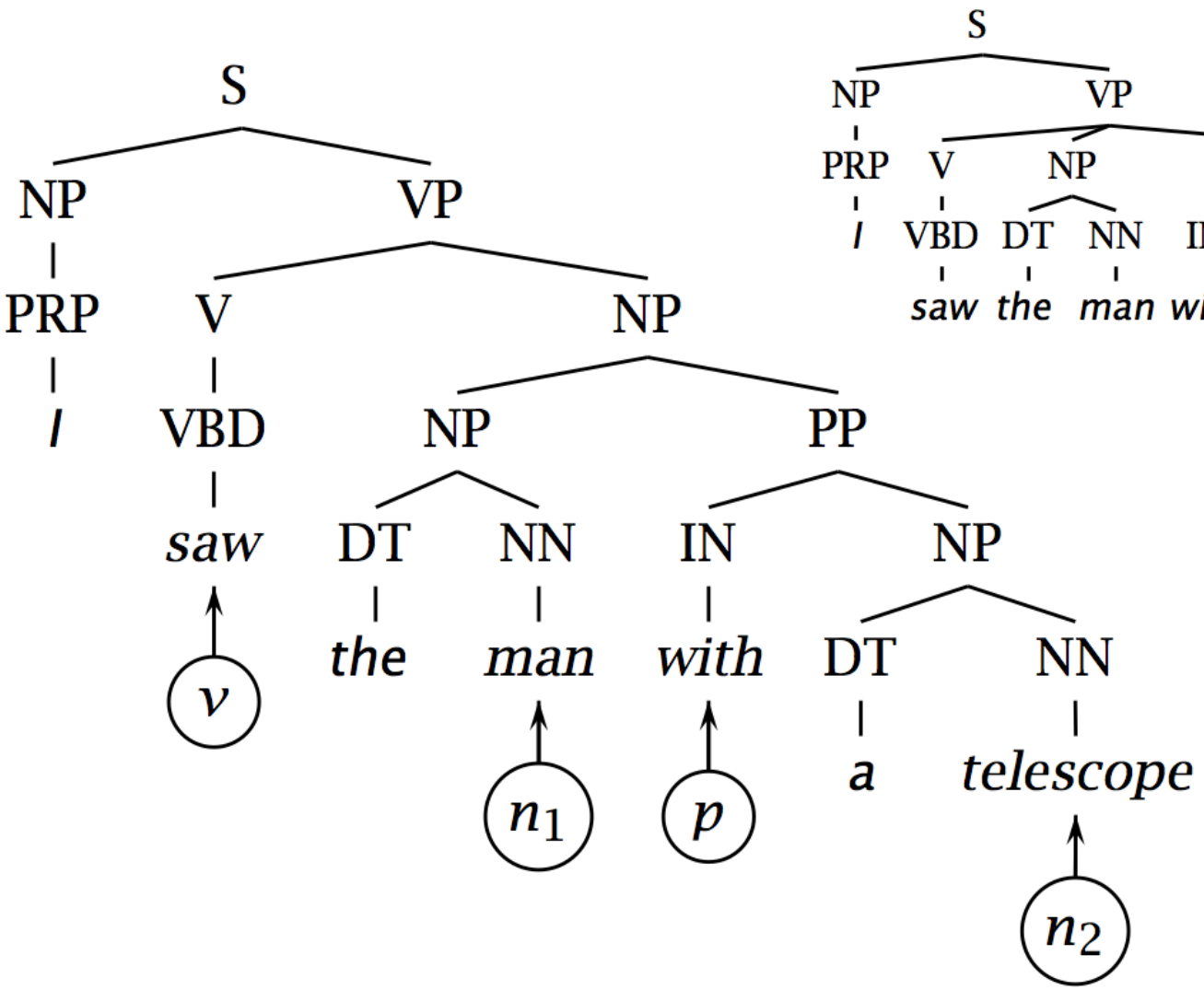
A simple prediction

- Moscow sent more than 100,000 soldiers into Afghanistan …
- Sydney Water breached an agreement with NSW Health …

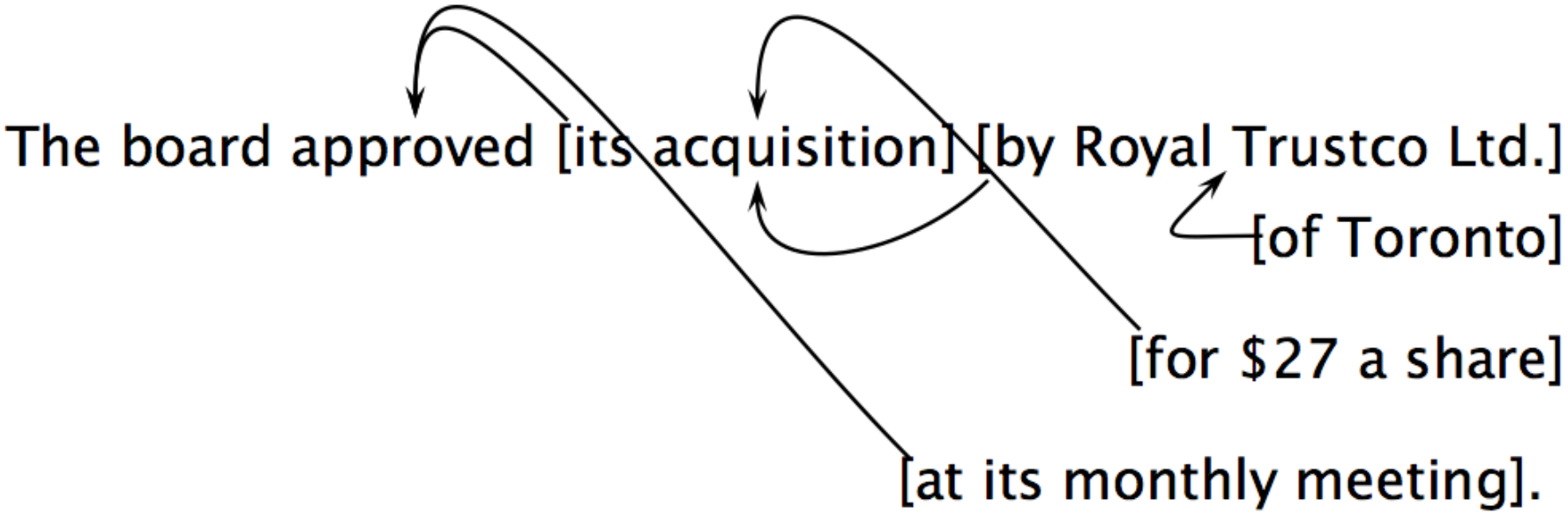
- P(*with*|*agreement*) = 0.15 p|n
- P(*with*|*breach*) = 0.02 p|v

- Ratio = p|v by p|n = 0.13 → prefer p-n attachment

Broader context is better



Attachment ambiguities in a real sentence



- Catalan numbers
  - $C_n = (2n)!/[(n+1)!n!]$
- An exponentially growing series, which arises in many tree-like contexts:
  - E.g., the number of possible triangulations of a polygon with $n+2$ sides

PARTS OF SPEECH

Parts of speech

- What are the English parts of speech?
  - 8 parts of speech?
    - Noun (person, place or thing)
    - Verb (actions and processes)
    - Adjective (modify nouns)
    - Adverb (modify verbs)
    - Preposition (on, in, by, to, with)
    - Determiners (a, an, the, what, which, that)
    - Conjunctions (and, but, or)
    - Particle (off, up)

Parts of Speech inventory (English)

NOUN	The DOG barked.	WE saw YOU.
VERB	The dog BARKED.	It IS impossible.
ADJECTIVE	He's very OLD.	I've got a NEW car.
DETERMINATIVE	THE dog barked.	I need SOME nails.
ADVERB	She spoke CLEARLY.	He's VERY old.
PREPOSITION	It's IN the car.	I gave it TO Sam.
COORDINATOR	I got up AND left.	It's cheap BUT strong.
SUBORDINATOR	It's odd THAT they were late.	I wonder WHETHER it's still there.
INTERJECTOR	OH, HELLO, WOW, OUCH	

Coordinator / subordinator: markers for coordinate / subordinate clauses
POS distinctions based on analysis of syntax and semantics

from [huddleston-pullum 05] *Student's intro to English Grammar*

POS categories

"parts-of-speech" : not sharply defined
some may be more **prototypical**:

prototypical	non-prototypical
noun: *cat, dog*	*equipment* (plural form?)
verb: *go, tell*	*must* (**musted, *to must*)
adj: *big, old,*	*asleep* (**an asleep dog*)

Parts of Speech inventory (Hindi)

1. Noun :	*billi* cat F, *kutta* dog M
2. Determiner :	*koi laRkA* some boy
3. Pronoun	*mai, tu, yeh, vah*
4. Adjective	*acchhA* (inflects for Gender, number, case); -tam/-tarin for superlative
5. Verb	*gir, girA, girvAyA* ; LIGHT: gir paRi, gA uThA
6. Adverb	*dhire, idhar,* COMPLEX: *dhyAn se, skul tak*
7. Postposition	*shyam ko, rAt mein,* COMPOUND: *ke sAmne*
8. Conjunction	*aur, lekin* SUBORDINATING: *agar, yadi, jo*
9. Particle	*hAn, na, to, matr*
10. Interjection	*are vah, bAp re*

from [Kachru 06] *Hindi*

English parts of speech

- Brown corpus: 87 POS tags
- Penn Treebank: ~45 POS tags
  - Derived from the Brown tagset
  - Most common in NLP
  - Many of the examples we'll show us this one
- British National Corpus (C5 tagset): 61 tags
- C6 tagset: 148
- C7 tagset: 146
- C8 tagset: 171

Closed vs. Open Class

- ***Closed class*** categories are composed of a small, fixed set of grammatical function words for a given language.
  - Pronouns, Prepositions, Modals, Determiners, Particles, Conjunctions
- Open class categories have large number of words and new ones are easily invented.
  - Nouns (Googler, futon, iPad), Verbs (Google, futoning), Adjectives (geeky), Abverb (chompingly)

Part of speech tagging

- Annotate each word in a sentence with a part-of-speech marker
- Lowest level of syntactic analysis

John saw the saw and decided to take it to the table.

NNP VBD DT NN CC VBD TO VB PRP IN DT NN

Penn Tagset

Tag	Description	Example	Tag	Description	Example
CC	Coordin. Conjunction	*and, but, or*	SYM	Symbol	+,%, &
CD	Cardinal number	*one, two, three*	TO	"to"	*to*
DT	Determiner	*a, the*	UH	Interjection	*ah, oops*
EX	Existential 'there'	*there*	VB	Verb, base form	*eat*
FW	Foreign word	*mea culpa*	VBD	Verb, past tense	*ate*
IN	Preposition/sub-conj	*of, in, by*	VBG	Verb, gerund	*eating*
JJ	Adjective	*yellow*	VBN	Verb, past participle	*eaten*
JJR	Adj., comparative	*bigger*	VBP	Verb, non-3sg pres	*eat*
JJS	Adj., superlative	*wildest*	VBZ	Verb, 3sg pres	*eats*
LS	List item marker	*1, 2, One*	WDT	Wh-determiner	*which, that*
MD	Modal	*can, should*	WP	Wh-pronoun	*what, who*
NN	Noun, sing. or mass	*llama*	WP$	Possessive wh-	*whose*
NNS	Noun, plural	*llamas*	WRB	Wh-adverb	*how, where*
NNP	Proper noun, singular	*IBM*	$	Dollar sign	*$*
NNPS	Proper noun, plural	*Carolinas*	#	Pound sign	*#*
PDT	Predeterminer	*all, both*	"	Left quote	(' or ")
POS	Possessive ending	*'s*	"	Right quote	(' or ")
PP	Personal pronoun	*I, you, he*	(	Left parenthesis	([, (, {, <)
PP$	Possessive pronoun	*your, one's*	)	Right parenthesis	( ],), }, >)
RB	Adverb	*quickly, never*	,	Comma	,
RBR	Adverb, comparative	*faster*	.	Sentence-final punc	(. ! ?)
RBS	Adverb, superlative	*fastest*	:	Mid-sentence punc	(: ; ... – -)
RP	Particle	*up, off*			

Figure 8.6 Penn Treebank Part-of-Speech Tags (Including Punctuation)

Penn Treebank
[Marcus etal 93]

Figure: jurafsky-martin ch.8 (2000)

English POS Subcategories

- Adjective (modify nouns)
  - Basic (JJ): red, tall
  - Comparative (JJR): redder, taller
  - Superlative (JJS): reddest, tallest
- Adverb (modify verbs)
  - Basic (RB): quickly
  - Comparative (RBR): quicker
  - Superlative (RBS): quickest
- Preposition (IN): on, in, by, to, with
- Determiner:
  - Basic (DT) a, an, the
  - WH-determiner (WDT): which, that
- Coordinating Conjunction (CC): and, but, or,
- Particle (RP): off (took off), up (put up)

Hindi Parts of Speech - Base

- 1. Noun (N)
- 2. Pronoun (P)
- 3. Demonstrative (D)
- 4. Nominal Modifier (J)
- 5. Verb (V)
- 6. Adverb (A)
- 7. Postposition (PP)
- 8. Particle (C)
- 9. Numeral (NUM)
- 10. Reduplication (RDP)
- 11. Residual (RD)
- 12. Unknown (UNK)
- 13. Punctuation (PU)

Hindi Parts of Speech - Details

- Noun (N)
  - Common(NC) Gender, Number, Case, Distributive, Honorificity
  - Proper(NP) Gender, Number, Case, Honorificity
  - Verbal(NV) Case ex: जाने\NV के\PP लिए\PP
  - Spatio-temporal (NST) Case, Distributive, Emphatic, Dimension ex: आज, समक्ष
- Nominal Modifier (J)
  - Adjective (JJ) Gender, Number, Case, Distributive
  - Quantifier (JQ) Gender, Number, Case, Numeral, Distributive
  - Intensifier (JINT) Gender, Number, Case

Hindi Parts of Speech - Details

Particle (C)

- Co-ordinating (CCD)
- Subordinating (CSB)
- Interjection (CIN)
- (Dis)Agreement (CAGR)
- Emphatic (CEMP)
- Topic (CTOP)
- Delimitive (CDLIM)
- Honorific (CHON)
- Dedative (CDED)
- Exclusive (CEXCL)
- Interrogative (CINT)
- Dubitative (CDUB)
- Similative (CSIM) Gender, Number
- Others (CX) Gender, Number, Case

Syntax-Semantics Continuum

- What is a noun?
  - Parts of speech categories – are they purely syntactic?
- What about deictics : *you*, *the vase there*
- Some grammatical categories (e.g. plural-singular, mass-count, tense)
  – correlated with meaning?
- What is language about, if not about meaning

[pinker 94]: language instinct

Universal POS categories

petrov etal 11

A Universal Part-of-Speech Tagset

Slav Petrov
Google Research
New York, NY, USA
slav@google.com

Dipanjan Das
Carnegie Mellon University
Pittsburgh, PA, USA
dipanjan@cs.cmu.edu

Ryan McDonald
Google Research
New York, NY, USA
ryanmcd@google.com

Abstract

To facilitate future research in unsupervised induction of syntactic structure and to standardize best-practices, we propose a tagset that consists of twelve universal part-of-speech categories. In addition to the tagset, we develop a mapping from 25 different treebank tagsets to this universal set. As a re-

forms across languages. These categories are often called *universals* to represent their cross-lingual nature (Carnie, 2002; Newmeyer, 2005). For example, Naseem et al. (2009) used the Multext-East (Erjavec, 2004) corpus to evaluate their multi-lingual POS induction system, because it uses the same tagset for multiple languages. When corpora with common tagsets are unavailable, a standard approach is

Universal POS categories

sentence:	The	oboist	Heinz	Holliger	has	taken	
original:	DT	NN	NNP	NNP	VBZ	VBN.	
universal:	DET	NOUN	NOUN	NOUN	VERB	VERB.	

a	hard	line	about	the	problems	.
DT	JJ	NN	IN	DT	NNS	.
:DET	ADJ	NOUN	ADP	DET	NOUN	.

Language	Source	# Tags	O/O	U/U	O/U
Arabic	PADT/CoNLL07 (Hajič et al., 2004)	21	96.1	96.9	97.0
Basque	Basque3LB/CoNLL07 (Aduriz et al., 2003)	64	89.3	93.7	93.7
Bulgarian	BTB/CoNLL06 (Simov et al., 2002)	54	95.7	97.5	97.8
Catalan	CESS-ECE/CoNLL07 (Martí et al., 2007)	54	98.5	98.2	98.8
Chinese	Penn ChineseTreebank 6.0 (Palmer et al., 2007)	34	91.7	93.4	94.1
Chinese	Sinica/CoNLL07 (Chen et al., 2003)	294	87.5	91.8	92.6
Czech	PDT/CoNLL07 (Böhmová et al., 2003)	63	99.1	99.1	99.1
Danish	DDT/CoNLL06 (Kromann et al., 2003)	25	96.2	96.4	96.9
Dutch	Alpino/CoNLL06 (Van der Beek et al., 2002)	12	93.0	95.0	95.0
English	PennTreebank (Marcus et al., 1993)	45	96.7	96.8	97.7
French	FrenchTreebank (Abeillé et al., 2003)	30	96.6	96.7	97.3
German	Tiger/CoNLL06 (Brants et al., 2002)	54	97.9	98.1	98.8
German	Negra (Skut et al., 1997)	54	96.9	97.9	98.6
Greek	GDT/CoNLL07 (Prokopidis et al., 2005)	38	97.2	97.5	97.8
Hungarian	Szeged/CoNLL07 (Csendes et al., 2005)	43	94.5	95.6	95.8
Italian		28	94.9	95.8	95.8
Japanese		80	98.3	98.0	99.1
Japanese		42	97.4	98.7	99.3
Korean		187	96.5	97.5	98.4
Portuguese		22	96.9	96.8	97.4
Russian		11	96.8	96.8	96.8
Slovene		29	94.7	94.6	95.3
Spanish		47	96.3	96.3	96.9
Swedish		41	93.6	94.7	95.1

[petrov das mcdonald 11]

25 languages.
Train / Test on O : original tags
U : universal

Japanese – much easier than Turkish
(avg sentence 11.6 tokens)

STATISTICAL LANGUAGE MODELS :

N-GRAMS

Probabilistic Language Modeling

- Goal: determine if a sentence or phrase has a high acceptability in the language

  → compute the probability of the sequence of words

  E.g. “its water is so transparent that”

  - P(its, water, is, so, transparent, that)

Probabilistic Language Modeling

$$P(W) = P(w_1,w_2,w_3,w_4,w_5 \ldots w_n)$$

- Related task: probability of an upcoming word:

$$P(w_5|w_1,w_2,w_3,w_4)$$

Reliability vs. Discrimination

- larger n: more information about the context of the specific instance (greater discrimination)
- smaller n: more instances in training data, better statistical estimates (more reliability)

How to compute P(W)

- Intuition: let's rely on the Chain Rule of Probability

Bayes -> The Chain Rule

- Recall the definition of conditional probabilities:
  $P(B|A) = P(A,B) / P(A) \rightarrow$

  $P(A,B) = P(A)\, P(B|A)$ [Assume: $P(A) > 0$]

- More variables:

  $P(A,B,C,D) = P(A)\, P(B|A)\, P(C|A,B)\, P(D|A,B,C)$

  Proof: Induction on the form:

  $P((A,B),C)) = P(A,B)\, P(C|(A,B)) = P(A)\, P(B|A)\, P(C|A,B)$

The Chain Rule

- Chain Rule in General

$$P(x_1,x_2,x_3,\ldots,x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1,x_2)\ldots P(x_n|x_1,\ldots,x_{n-1})$$

- Proof:
  - Holds for n=2 (Product rule)
  - Assume is true for $X = x_1 \ldots x_{n-1}$.

$$P(X , x_n) = P(X)\, P\,(x_n|X) \quad \rightarrow \text{General chain rule}$$

The Chain Rule

$$P(w_1 w_2 \ldots w_n) = \prod_i P(w_i \mid w_1 w_2 \ldots w_{i-1})$$

P("its water is so transparent") =

P(its) × P(water|its) × P(is|its water)

× P(so|its water is) × P(transparent|its water is so)

The Chain Rule

- Chain Rule in General

$$P(x_1,x_2,x_3,\ldots,x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1,x_2)\ldots P(x_n|x_1,\ldots,x_{n-1})$$

- Most useful when dependency of x_k is limited to only a few recent terms
  - First-order Markovian: x_k depends only on x_{k-1}

Estimating the probabilities

- Could we just count and divide?

$$P(\text{the} \mid \text{its water is so transparent that}) = \frac{Count(\text{its water is so transparent that the})}{Count(\text{its water is so transparent that})}$$

- Unlikely to find ANY instances in corpus!

Markov Assumption

Andrei Markov
1856-1922, Russia

- Simplifying assumption:

  Depends only on *k*-nearby text

- *First-order* Markov Process (k= 1):

$$P(\text{the} \mid \text{its water is so transparent that}) \approx P(\text{the} \mid \text{that})$$

- or *Second-order* (k=2):

$$P(\text{the} \mid \text{its water is so transparent that}) \approx P(\text{the} \mid \text{transparent that})$$

Markov Assumption

$$P(w_1 w_2 \ldots w_n) \approx \prod_i P(w_i \mid w_{i-k} \ldots w_{i-1})$$

- In other words, we approximate each component in the product

$$P(w_i \mid w_1 w_2 \ldots w_{i-1}) \approx P(w_i \mid w_{i-k} \ldots w_{i-1})$$

Estimating bigram probabilities

- The Maximum Likelihood Estimate

$$P(w_i \mid w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})}$$

$$P(w_i \mid w_{i-1}) = \frac{c(w_{i-1}, w_i)}{c(w_{i-1})}$$

Sentence Genration

Unigram Model: No dependencies on previous words

$$P(w_1 w_2 \ldots w_n) \approx \prod_i P(w_i)$$

Bigram Model : Depends on 1 previous word

$$P(w_i \mid w_1 w_2 \ldots w_{i-1}) \approx P(w_i \mid w_{i-1})$$

Unseen N-grams : Generalization and zeros

The perils of overfitting

- N-grams only work well for word prediction if the test corpus looks like the training corpus
  - In real life, it often doesn't
  - We need to train robust models that generalize!
  - One kind of generalization: Zeros!
    - Things that don't ever occur in the training set
      - But occur in the test set

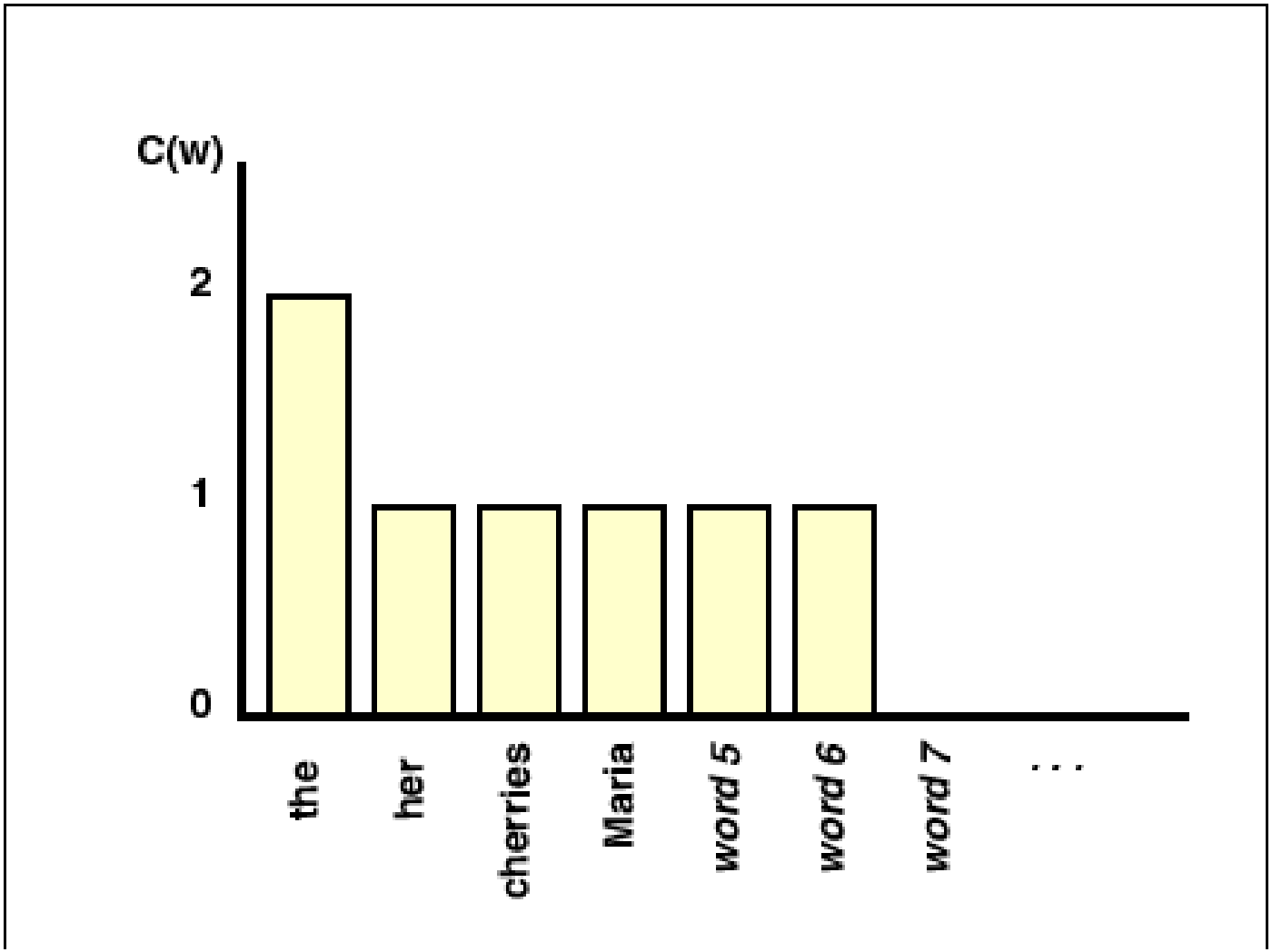
Zeros

- Training set:
  … denied the allegations
  … denied the reports
  … denied the claims
  … denied the request

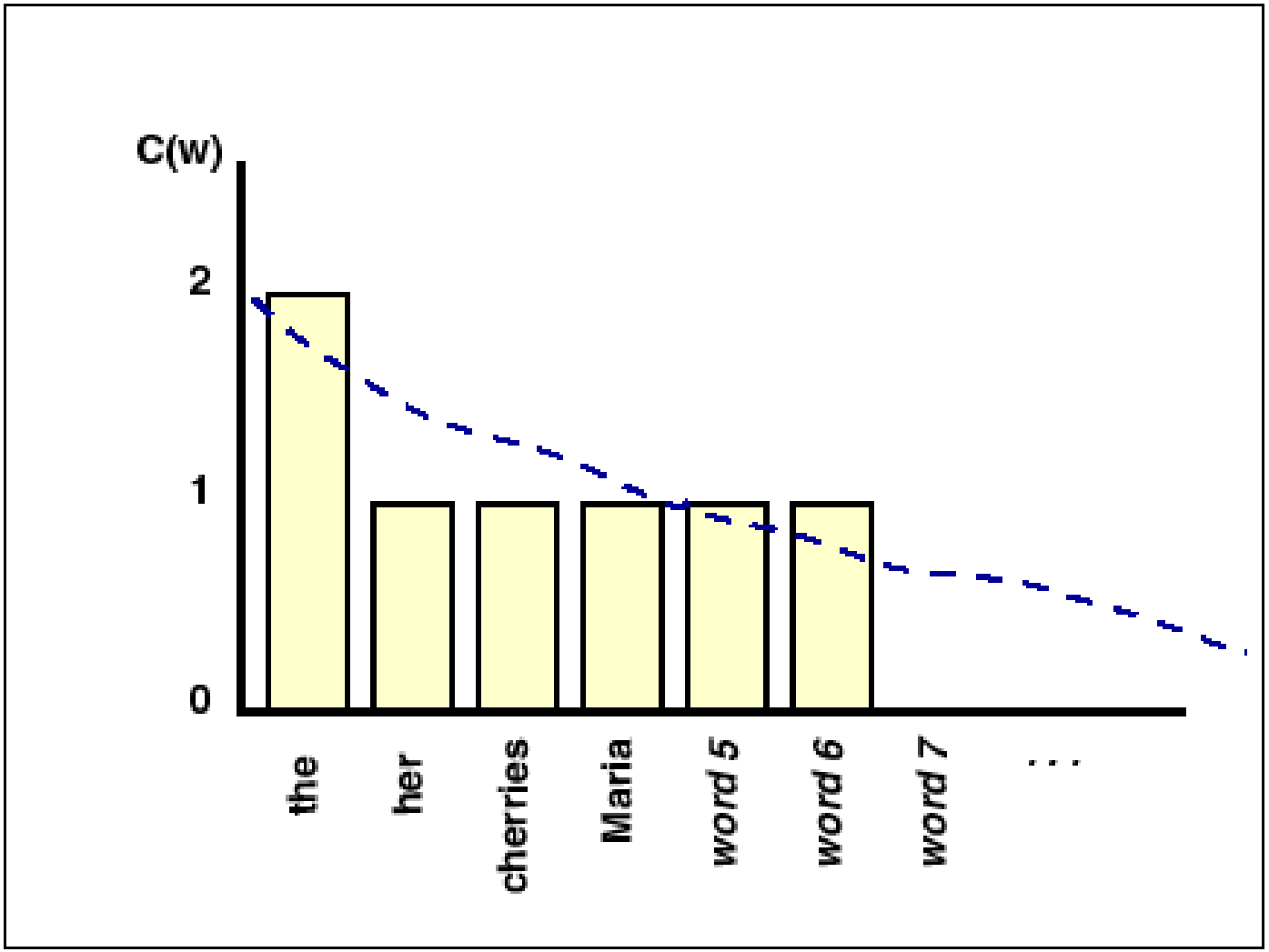
  P(“offer” | denied the) = 0

- Test set
  … denied the offer
  … denied the loan

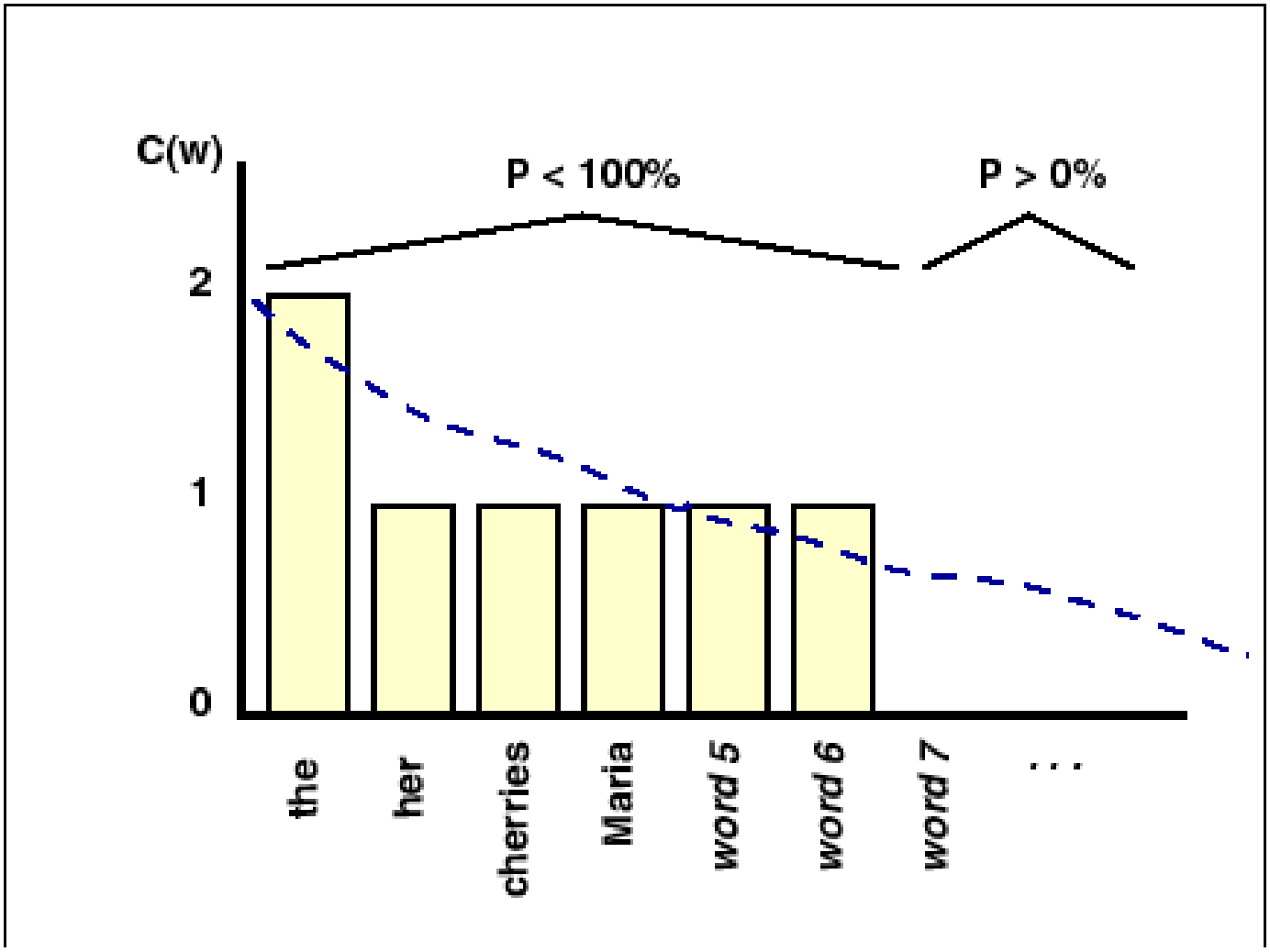
Smoothing



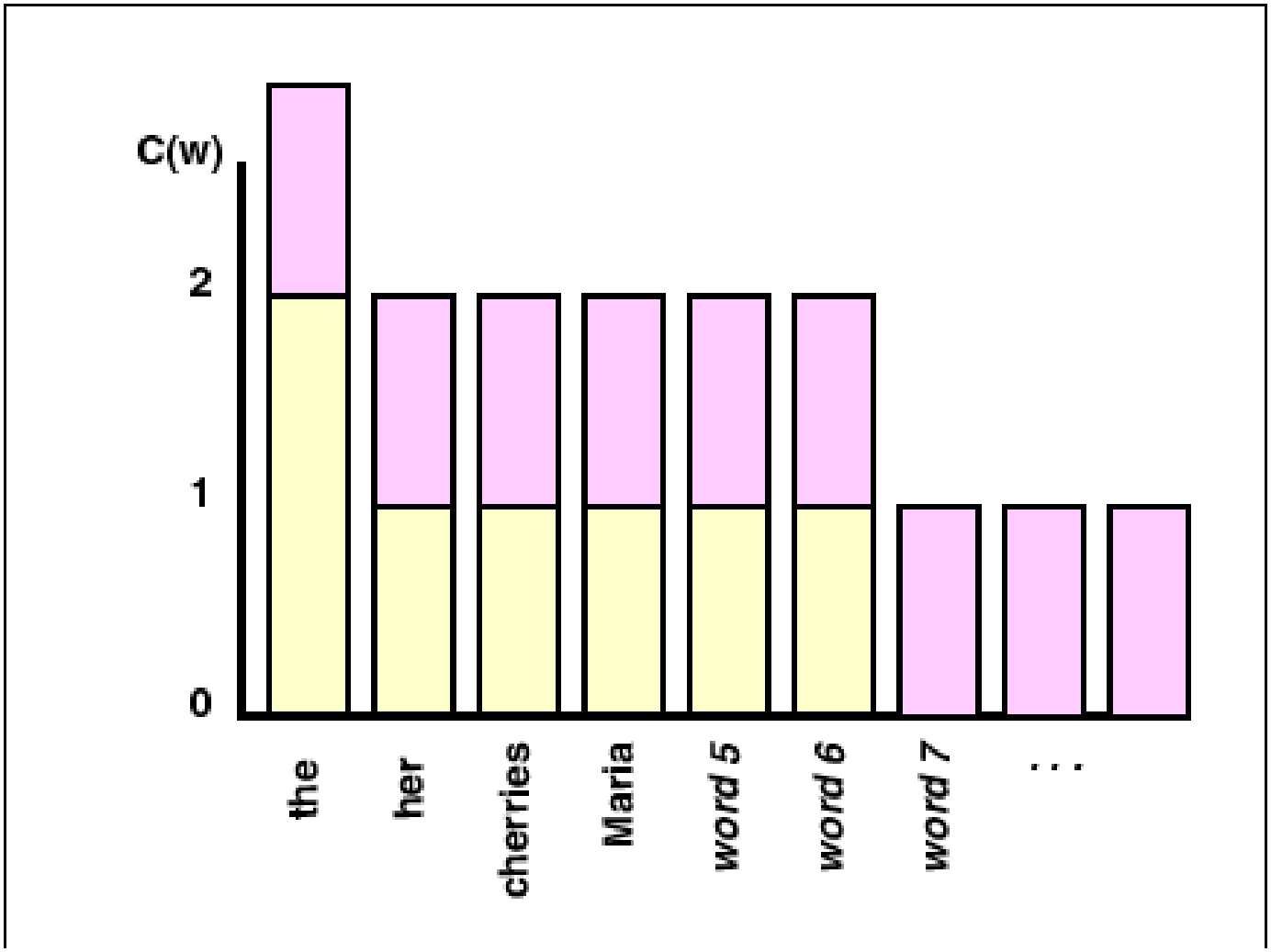
Actual Probability Distribution:



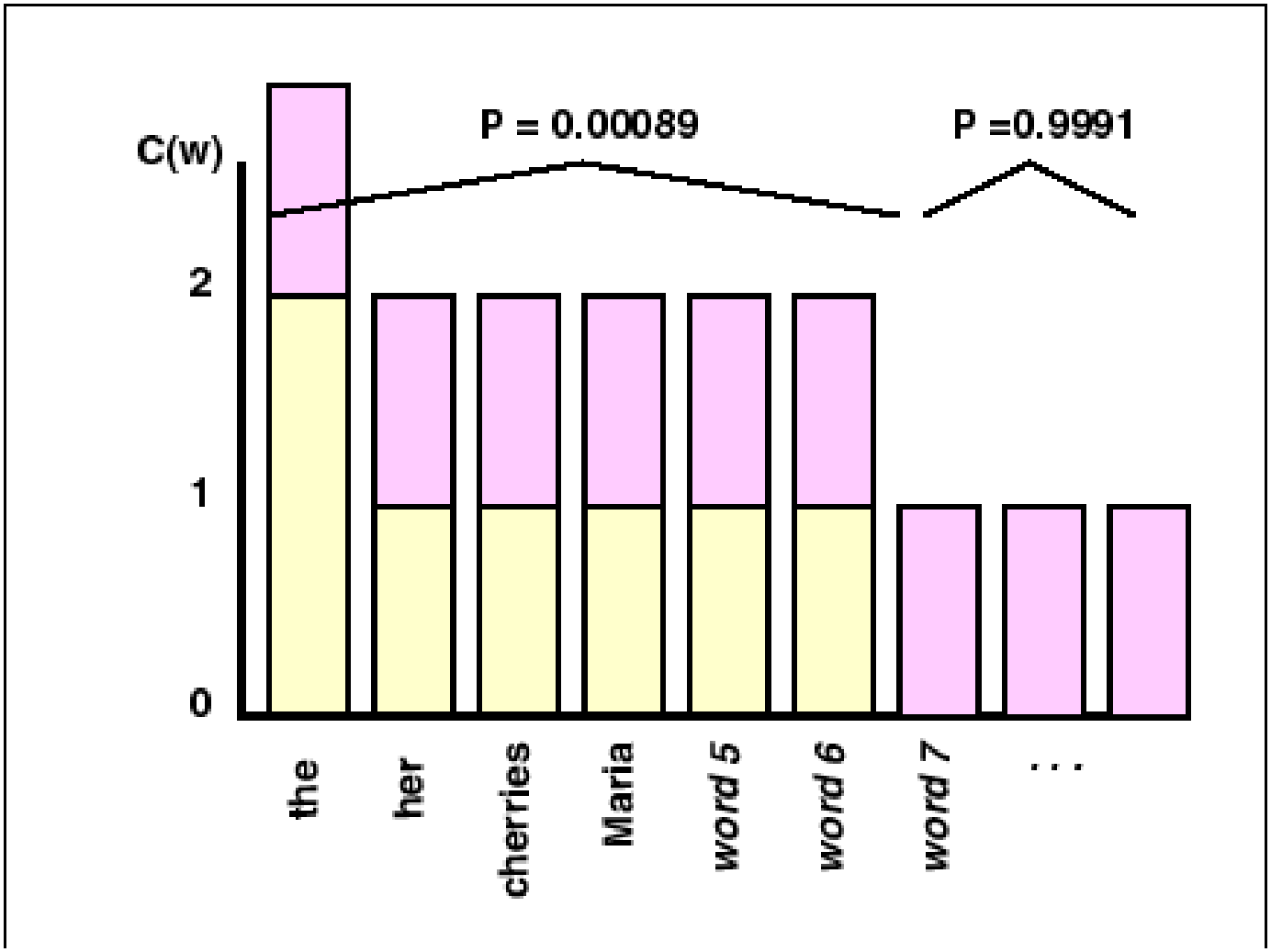
Actual Probability Distribution:



Smoothing: +1



Smoothing: +1


C(w)
P = 0.00089
P =0.9991
2
1
0
the
her
cherries
Maria
word 5
word 6
word 7
. . .

Good-Turing discounting

- How much probability mass to assign to unseen examples? (e.g. unseen bigrams / trigrams),
- Good-Turing estimation : good estimate for the total probability of unseen n-grams = total number of 1-grams *seen* = N_1/N.
- If removing words from corpus, probability of removing a word of frequency i is $\frac{i * N_i}{N}$
- Frequency 1 : N1/N

N-Gram Morphological Analysis

Language Differences

- **Morphemes per word:**

West Greenlandic	3.72	polysynthetic
Sanskrit	2.59	
Swahili	2.55	synthetic
Old English	2.12	
German	1.92	
Modern English	1.68	
Vietnamese	1.06	isolating

[haspelmath & sims 2010] understanding morphology

Language Differences

- **West Greenlandic**

Paasi-nngil-luinnar-para

understand-not-completely-1SG.SBJ.3SG.OBJ.IND

ilaa-juma-sutit.

come-want-2SG.PTCP

'I didn't understand at all that

you wanted to come along.'

(Fortescue 1984: 36)

[haspelmath & sims 2010] understanding morphology

Concatenative Morphology

Concatenative Assumption : phonological material added at

start: prefix

end : suffix

mid : infix

word = prefix[es] + stem + suffix[es]

[hammarstrom borin 11]

Unsupervised Morphology (ULM)

(a) **Border and Frequency**: morphemes = substrings that have varied neighbours

(b) **Group and Abstract**: cluster morphologically related words (e.g. string edit distance, or distributional similarity)

(c) **Features and Classes:** feature = n-grams ; rare features (entropy) --> specific word or stem.

(d) **Phonological Categories and Separation** : vowel / consonant skeletons

[hammarstrom borin 11]

Unsupervised Morphology (ULM)

- Morpheme segmentation
- Paradigm induction
  - *paradigm* = full set of inflections in a language
  - "set" → exponential in #affixes

sg	1-sg	2-sg	3-sg
pres	i sing,	you sing,	[s]he sings,
past	i sang,	you sang,	[s]he sang,

pl	1-pl	2-pl	3-pl
pres	we sing,	you sing,	they sing
past	we sang,	you sang,	they sang

Morpheme Segmentation

(a) **Border and Frequency**: morphemes = substrings that have varied neighbours

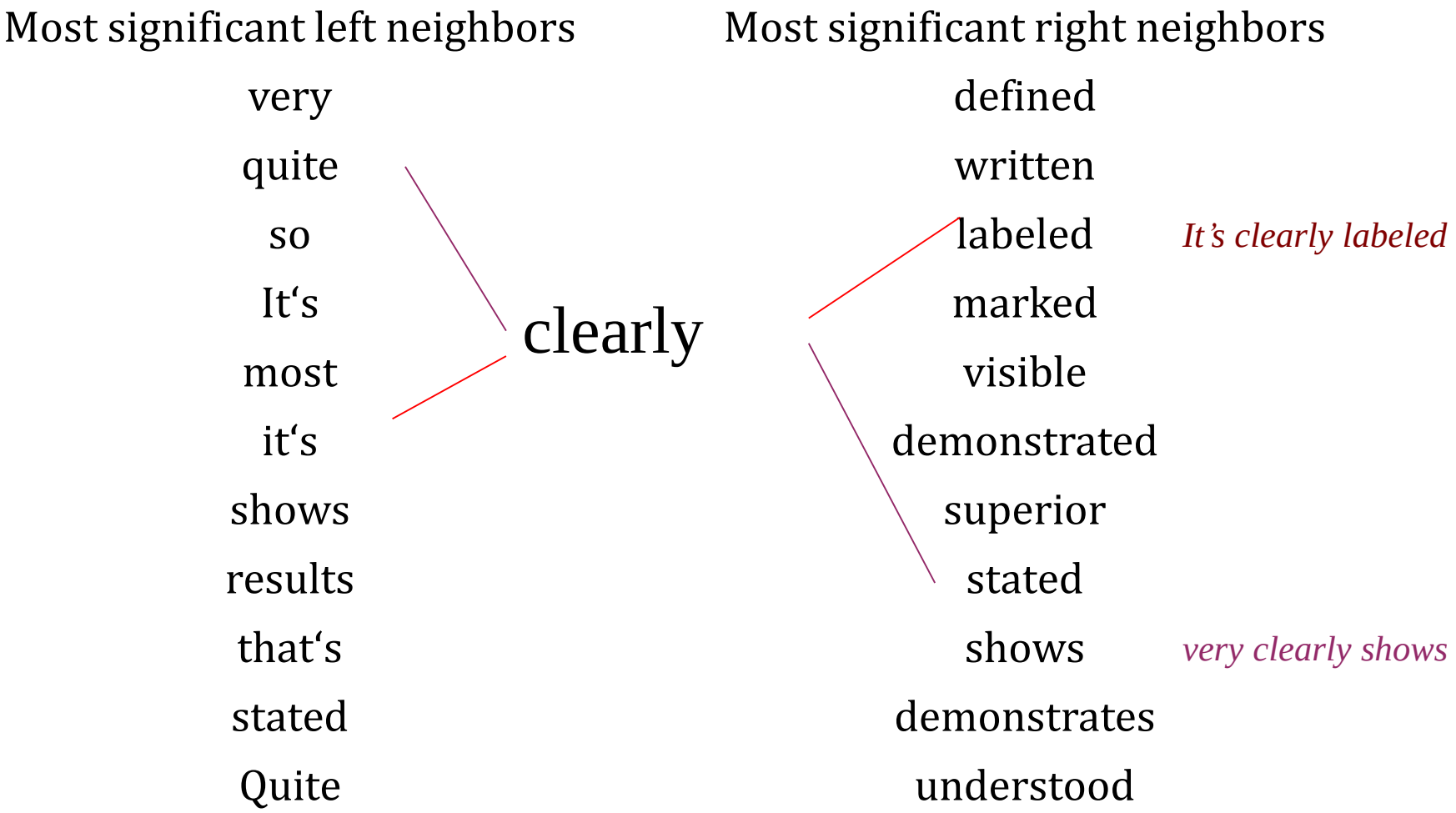
(b) **Group and Abstract**: cluster morphologically related words (e.g. string edit distance, or distributional similarity)

(c) **Features and Classes:** feature = n-grams ; rare features (entropy) --> specific word or stem.

(d) **Phonological Categories and Separation** : vowel / consonant skeletons

[hammarstrom borin 11]

Distributional Similarity



Distributional Similarity

Most significant left neighbors		Most significant right neighbors
very		defined
quite		written
so	…	labeled
It‘s	weakly	marked
most	legally	visible
it‘s	closely	demonstrated
shows	**clearly**	superior
results	greatly	stated
that‘s	linearly	shows
stated	really	demonstrates
Quite	…	understood

Morpheme Segmentation

(a) **Border and Frequency**: morphemes = substrings that have varied neighbours

(b) **Group and Abstract**: cluster morphologically related words (e.g. string edit distance, or distributional similarity)

(c) **Features and Classes:** feature = n-grams ; rare features (entropy) --> specific word or stem.

(d) **Phonological Categories and Separation** : vowel / consonant skeletons

[hammarstrom borin 11]

Zellig Harris 1967

Given the first *m* phonemes of a *n*-phoneme word, we count how many different phonemes follow these first *m* phonemes...

letter successor variety : LSV

The same procedure can be used to count the predecessors of the last *m* phonemes...

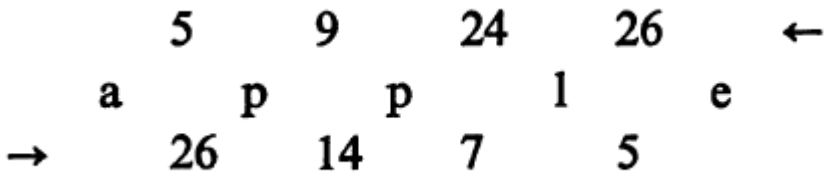
letter predecessor variety : LPV

The points in the given word at which the number of successors (or predecessors) peaks are [approximately], the boundaries between the morphemic segments

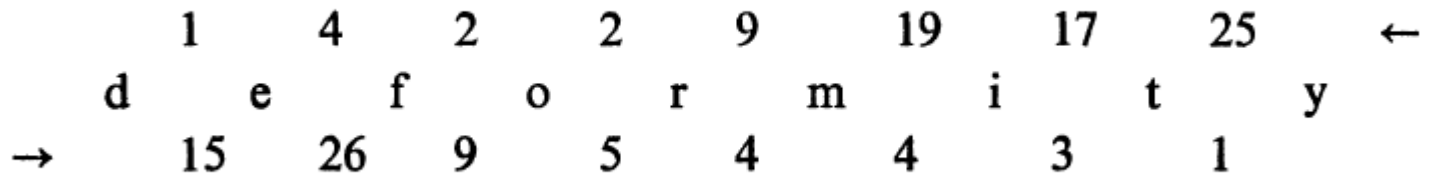
[Harris, 67] Morpheme Boundaries within Words - a Computer Test p.68

Zellig Harris 1967

apple



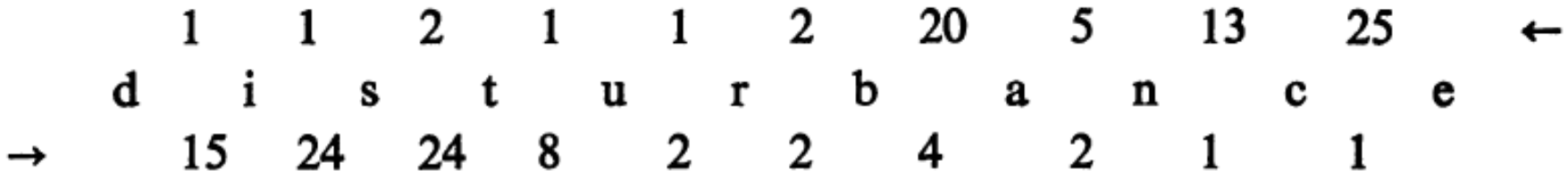
deformity



[Harris, 67] Morpheme Boundaries within Words - a Computer Test p.68

Zellig Harris 1967

disturbance



[Harris, 67] Morpheme Boundaries within Words - a Computer Test p.68

LSV

set of all words = W

LSV (letter successor variety) of a string x of length i

LSV(x) = number of distinct letters that occupy the i + 1st position in words in W that begin with x :

$$LSV(x) = |\{z[|x| + 1] | z = xy \in W\}|$$

LSV

$W = \{\text{abide, able, abode, and, art, at, bat}\}$				
x	a	ab	abe	...
$\{z \mid z = xy \in W\}$	{abide, able, abode, and, art, at}	{abide, able, abode}	$\emptyset$	...
$LSV(x)$	4 (b,n,r,t)	3 (i,l,o)	0	...

Threshold → no theoretical basis

LSV / LPV / LSE ??

Normalized LSV / LPV / LSE

Table 6
Normalized LPV/LPE/LPM-scores for *-e, -ce, -nce, . . . , -disturbance*. All figures are computed on the Brown Corpus of English (Francis and Kucera 1964), using the 27-letter alphabet $[a - z]$ plus the apostrophe. There are $|W| = 42,353$ word types in lowercase.

	d	i	s	t	u	r	b	a	n	c	e
$\overline{LPV}$	0.03	0.03	0.03	0.03	0.03	0.03	0.70	0.22	0.44	0.92	
$\overline{LPE}$	0.0	0.0	0.0	0.0	0.0	0.0	0.74	0.28	0.38	0.81	
$\overline{LPM}$	0.0	0.0	0.0	0.0	0.0	0.0	0.83	0.53	0.37	0.85	

Frequency analysis

Overrepresentation as more-frequent-than-its-length: For a segment x of $|x|$ characters, it is overrepresented to the degree that it is more common than expected from a segment of its length. This applies to a segment in any position.

$$\frac{f(x)}{|\Sigma|^{|x|}}$$

Overrepresentation as more-frequent-than-its-parts: For a segment $x = c_1c_2 \ldots c_n$ of n characters, it is overrepresented to the degree that it is more common than expected from a co-occurrence of its parts. This applies to a segment in any position.

$$\frac{f(c_1c_2 \ldots c_n)}{f(c_1)f(c_2) \ldots (c_n)}$$

Reading

@article{hammarstrom-borin-11_unsupervised-learning-of-morphology,

title={Unsupervised learning of morphology},

author={Hammarstr{\"o}m, Harald and Borin, Lars},

journal={Computational Linguistics},

volume={37}, number={2}, pages={309--350},

year={2011},

publisher={MIT Press},