

# LECTURE 10 - NATURAL LANGUAGE PROCESSING

Cues

Based on sequences

Notes

Language models

- Two types

- - Probabilistic, word based, learned

- - Logical, abstraction based, hand coded

Based on syntax trees

Probabilistic Models

- Bag of words (in ML lectures of AI class) is an example of a probabilistic model

- We want

$P = \text{probabilistic model}$   
 $W = \text{word}$

$$P(W_1, W_2, W_3, \dots, W_n) = P(W_{1:n}) \\ = \prod_i P(W_i | W_{1:i-1})$$

- Markov assumption:

$$P(W_i | W_{1:i-1}) \approx P(W_i | W_{i-k:i-1})$$

$$\rightarrow = P(W_i | W_{i-1}) \quad (\text{for } k=1)$$

The effect of previous words on the current word is localized to  $k$  previous words

Summary

Cues

Notes

Stationarity Assumption:

Probability distributions are the same for all words

$$P(w_i | w_{i-1}) = P(w_j | w_{j-1})$$

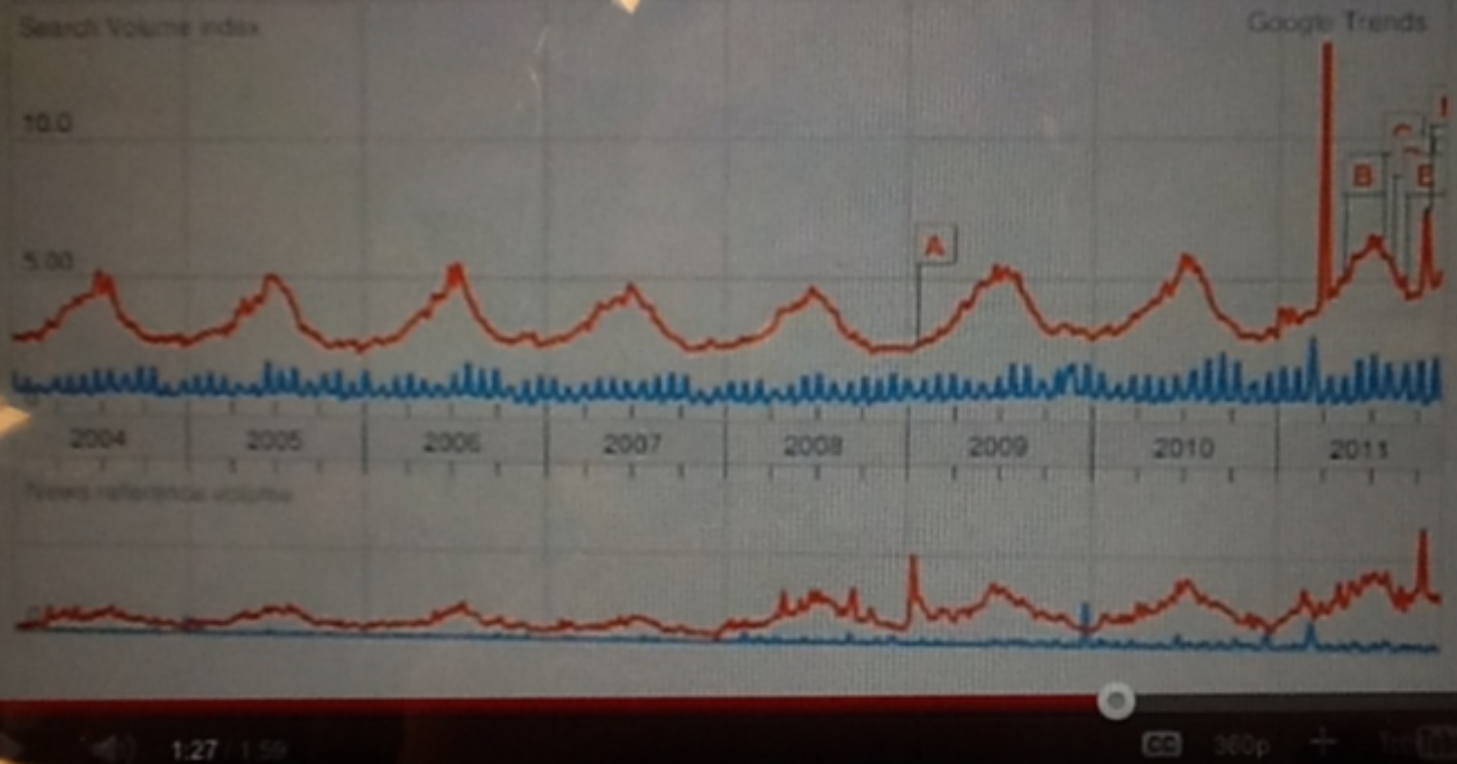
In practice, might want to:

- apply smoothing (Laplace & others)
- augment with alternate data (such as context)
- recover hidden variables like word usage (noun vs. verb)
- consider sequences of words, like "New York City" as single words
- Learning languages can reveal other information about the world.

Summary

## Unit 21 05 Language and Learning

by KnowlbyJecs



Discuss this question on aiqu. When posting use the tag 'unit21-5'

0 2 228 A strange spike in Google Trends

Summary Trends in search volume for the above phrases can show earth orbit period & lunar orbit period.

Cues

Notes

## N-Gram Models

Unigram: N-gram with  $k=1$  - Considering only 1 previous word

Bigram: N-gram with  $k=2$

Trigram: N-gram with  $k=3$

Letter based N-Grams can be used to identify languages

GZip as a classifier:

- Leverages Huffman encoding that count subsequences - add new text to old text and count number of different subsequences.

Summary

• Sentiment analysis

## Cues

## Notes

## Segmentation:

Chinese has no clear word boundaries → In languages with no clear word boundaries, we need a way to determine boundaries.  
Also present in speech recognition

## Probabilistic Segmentation model:

$$S^* = \max P(W_i:n) \\ = \max \prod_i P(W_i | W_{1:i-1})$$

The ideal segmentation model is the one that maximizes the joint probability of all words

Naive bayes  
(wrong simplification) →

Approximated to

$$S^* \approx \max P(W_i) \\ \approx \max P(W_i | W_{k:i-1}) \leftarrow \text{Markov assumption}$$

## Summary

Cues

Notes  
There are  $2^{(n-1)}$  ways to segment  $n$  characters.

This is a large number

Using the  $S^*$  approximation from before:

$$S^* = \operatorname{argmax}_{S \rightarrow r} P(f) \cdot P(S^*(r))$$



Maximize the probability that  $f$  is a word, given the probability the rest of the characters are a word

Example:

"nowisthetime"

Uses a corpus to  
count instances

Summary  
& smooth

<u>f</u>	<u>r</u>	<u>P(f)</u>	<u>P(f) · P(S*(r))</u>
n	owis...	0.00001	$10^{-17}$
no	wist...	0.004	$10^{-13}$
now	isth...	0.003	$10^{-10}$
nowi	sthe...	:	$10^{-18}$

↑  
values depend on  
smoothing used

Cues

Notes

Spelling correction:  
 $c^* = \operatorname{argmax}_c P(c|w)$   
 $= \operatorname{argmax}_c P(w|c) \cdot P(c)$  by Bayes rule  
 $P(c) =$  from data counts from corpus  
 $P(w|c) =$  from spelling correction data  
(or calculated with Levenshtein distance  
(or Hamming?))

Can also consider word on a letter by letter basis

Summary

Cues

Notes

Sentence structure:

where

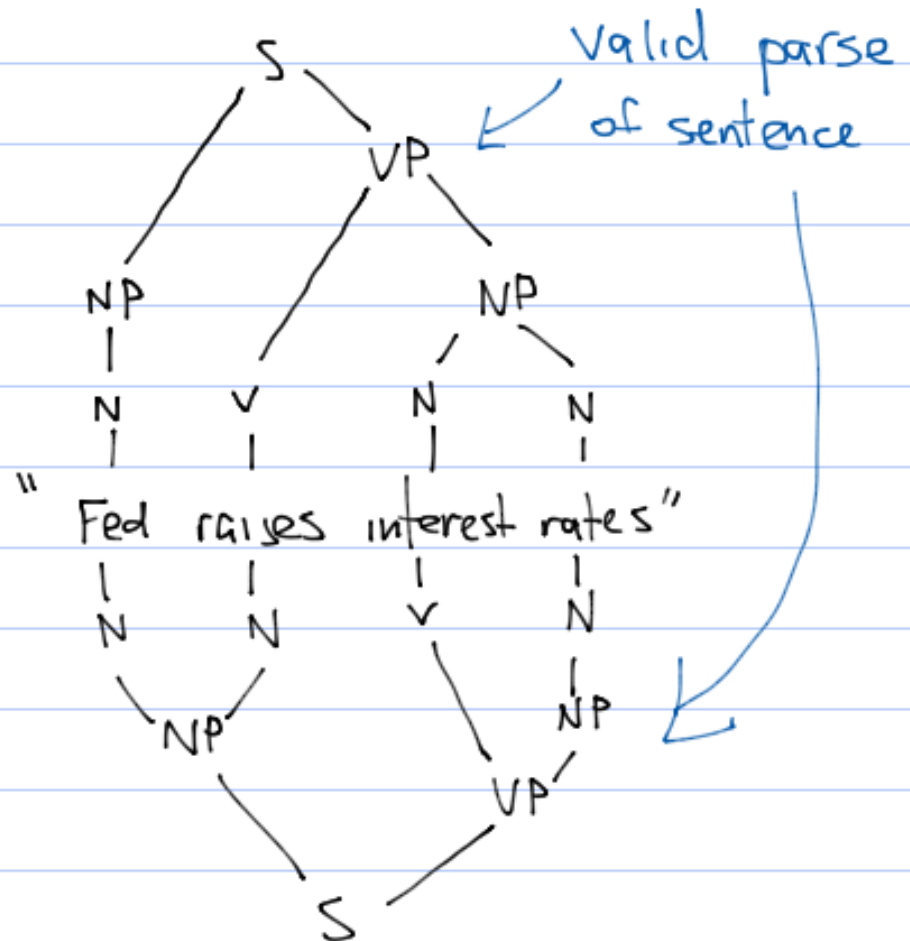
N = noun

NP = noun phrase

V = verb

VP = verb phrase

S = sentence



Summary

Cues

Parse trees come from grammars

Example: D = determiner ("the")

Context free  
grammars

$S \rightarrow NP VP$

$NP \rightarrow N \mid DN \mid NN \mid NNN$

$VP \rightarrow V NP \mid V \mid V NP NP$

$N \rightarrow \text{interest} \mid \text{Fed} \mid \text{rates} \mid \text{raises}$

$V \rightarrow \text{interest} \mid \text{rates} \mid \text{raises}$

$D \rightarrow \text{the} \mid \text{a}$

The above grammar has problems with ambiguity - the same sentence has multiple parses

Summary

Cues

Notes

Writing grammars:

- Goal is to write a grammar that almost matches a language closely, but it's not easy because languages are amorphous and complex

Probabilistic Context Free Grammars (PCFG):

$$S \rightarrow NP VP \quad (1)$$

$$NP \rightarrow N \quad (0.3)$$

$$| DN \quad (0.4)$$

$$| NN \quad (0.2)$$

$$| NNN \quad (0.1)$$

$$N \rightarrow \text{interest} \quad (0.3)$$

$$| \text{Fed} \quad (0.3)$$

$$| \text{rates} \quad (0.3)$$

$$| \text{raises} \quad (0.1)$$

$$VP \rightarrow V NP \quad (0.4)$$

$$| V \quad (0.4)$$

$$| V NP NP \quad (0.2)$$

$$V \rightarrow \text{interest} \quad (0.1)$$

$$| \text{rates} \quad (0.3)$$

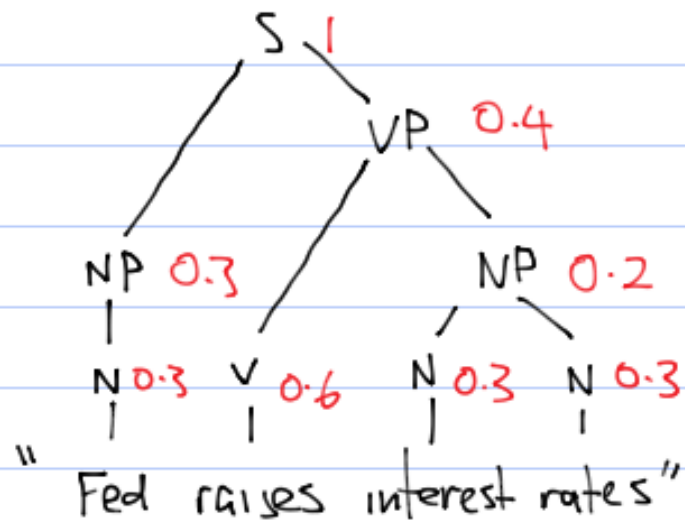
$$| \text{raises} \quad (0.6)$$

Summary

$$D \rightarrow \text{the} \quad (0.5)$$

$$| \text{a} \quad (0.5)$$

Cues

Notes  
G=

$$\begin{aligned}
 P(G) &= 0.3 \times 0.6 \times 0.3 \times 0.3 \\
 &\times 0.3 \times 0.2 \\
 &< 0.4 \\
 &< 1 \\
 &= 0.003888 \\
 &= 0.39\%
 \end{aligned}$$

Going up the tree ↓

Summary

Cues

Parse trees



Notes

Where do the probabilities for PCFGs come from?

Data. Initially was created manually by hand.

Resolving ambiguity:

Example:

"I saw the man with a telescope"

• Can be parsed as

NP V NP - I saw the man (who has a telescope)

or

NP V NP PP - I saw the man (using a telescope)



Prepositional phrase

Resolving this ambiguity depends on answering "Does 'telescope' & 'with' go better with 'man' or 'saw'?"

Summary

Cues

Notes  
Lexicalized Probabilistic Context Free Grammar  
(LPCFG):

Deals with specific words, instead of categories.

PCFG:

What is  $P(VP \rightarrow V \text{ NP NP} \mid \text{lhs} = VP)$

left hand side  
↙

LPCFG

What is  $P(VP \rightarrow V \text{ NP NP} \mid V = \text{gave}) = 0.25$   
vs.  $P(VP \rightarrow V \text{ NP NP} \mid V = \text{said}) = 0.0001$

Summary

Cues

Notes

Parsing into a tree:

- Process of search
  - Either top down or bottom up works
- Attach probabilities is the same process - search
  - Vauquois Pyramid (Machine Translation)

Summary