NLP Lecture 5

Wordnet Lexical Relations

What's special about WordNet?

WordNet gives information about two fundamental, universal properties of human language:

polysemy and **synonymy**

Polysemy = one:many mapping of form and meaning

Synonymy = one:many mapping of meaning and form

Polysemy

One word form expresses multiple meanings

{*table*, tabular_array}

{*table*, piece_of_furniture}

{*table*, mesa}

{*table*, postpone}

Note: the most frequent word forms are the most polysemous!

Synonymy

One concept is expressed by several different word forms:

{beat, hit, strike}

{car, motorcar, auto, automobile}

Polysemy and synonymy

Understanding and generating language (as for translation) means matching a word form with the intended, context-appropriate meaning

People (fluent speakers of a language) do this very efficiently

Polysemy in WordNet

A word form that appears in n synsets

is n-fold polysemous

{table, tabular_array}

{table, piece_of_furniture}

{table, mesa}

{table, postpone}

table is fourfold polysemous/has four senses

four distinct concepts are associated with the word form table

WN as a lexical resource

Have concept, need words”

--depart from synset, travel in WordNet space

“Have word, need concept”

--query word form, find associated synsets

WordNet as a lexical resource

WN has been incorporated into many dictionaries

Google “define” usually brings up WN entry at the top of the list

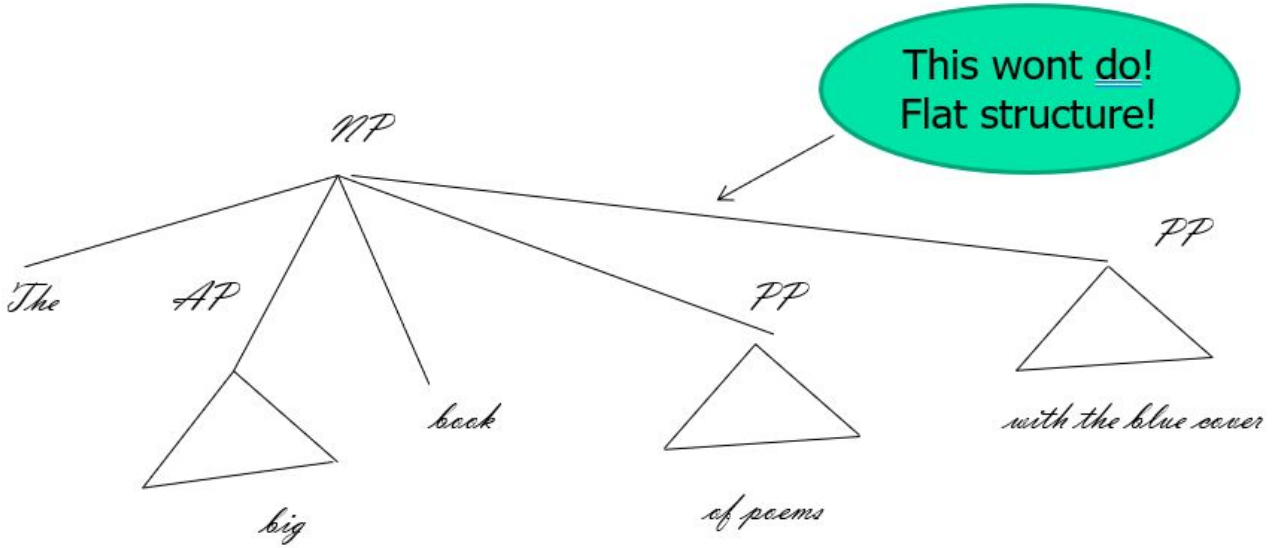
User-created visual interfaces (e.g., visualthesaurus.com)

Keyword Extraction

https://thinkinfi.com/automatic-keyword-extraction-using-rake-in-python/

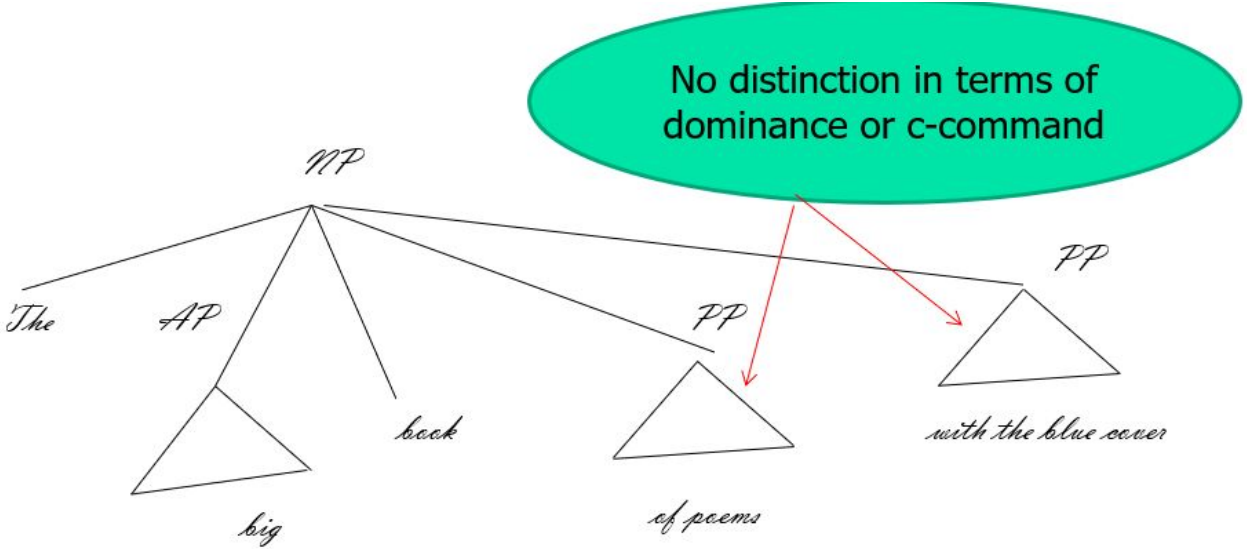
https://www.analyticsvidhya.com/blog/2020/11/words-that-matter-a-simple-guide-to-keyword-extraction-in-python/

Parsing



[The big book of poems with the Blue cover] is on the table.

PPs are at the same level: flat with respect to the head word “book”



[The big book of poems with the Blue cover] is on the table.

Constituency test of Replacement" runs into problems

One-replacement:

I bought the big [book of poems with the blue cover] not the small [one]

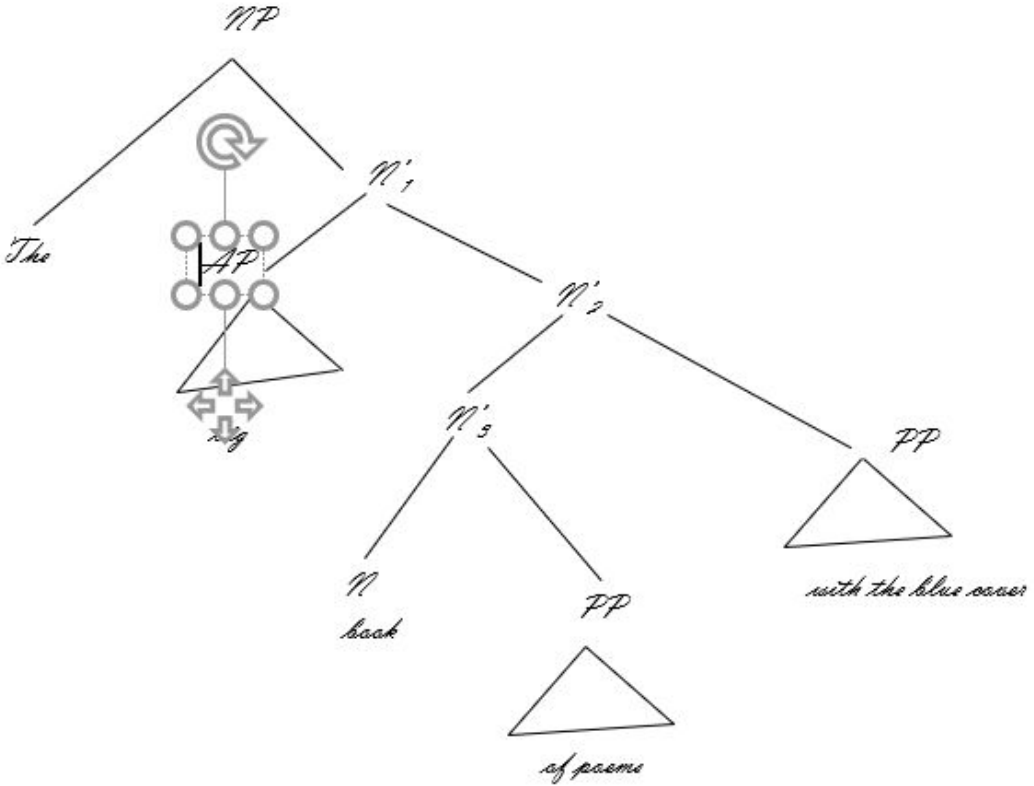
One-replacement targets book of poems with the blue cover

Another one-replacement:

I bought the big [book of poems] with the blue cover not the small [one] with the red cover

One-replacement targets book of poems

Deeply Embedded Structure



To target N1’

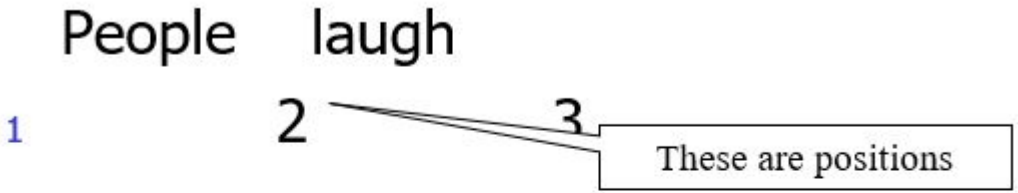
I want [NPthis [N’big book of poems with the red cover] and not [Nthat [None]]

Parsing Algorithms

A simplified grammar

- $S \rightarrow NP\ VP$
- $NP \rightarrow DT\ N \mid N$
- $VP \rightarrow V\ ADV \mid V$

Example Sentence



Lexicon:

People - N, V

Laugh - N, V

This indicate that both Noun and Verb is possible for the word "People"

Top-Down Parsing

	State	Backup State	Action
1.	((S) 1)	-	-
2.	((NP VP)1)	-	-
3a.	((DT N VP)1)	((N VP) 1)	-
3b.	((N VP)1)	-	-
4.	((VP)2)	-	Consume "People"
5a.	((V ADV)2)	((V)2)	-
6.	((ADV)3)	((V)2)	Consume "laugh"
5b.	((V)2)	-	-
6.	((.)3)	-	Consume "laugh"

Termination Condition : All inputs over. No symbols remaining.

Note: Input symbols can be pushed back.

Bottom-Up Parsing

Some conventions:

N_{12}

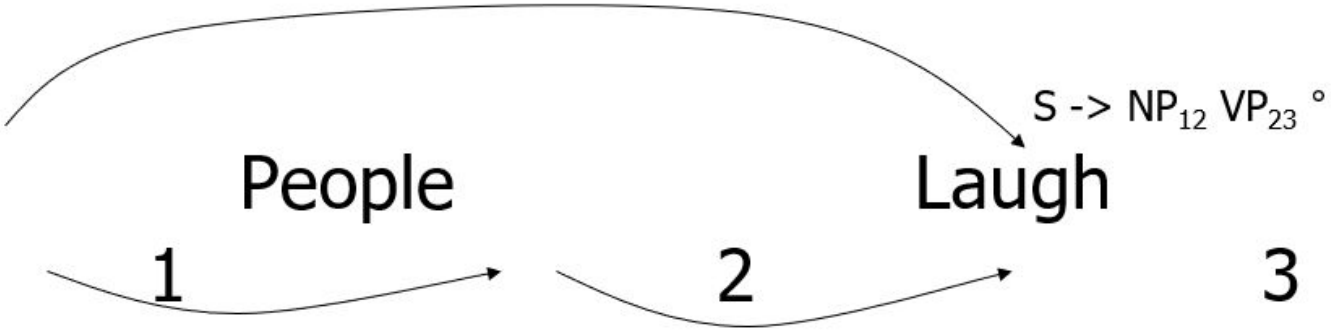
Represents positions

$S_{1?} \rightarrow NP_{12} \circ VP_{2?}$

End position unknown

Work on the LHS done, while the work on RHS remaining

Bottom-Up Parsing (pictorial representation)



N_{12}
V_{12}
NP_{12} -> N_{12} °
VP_{12} -> V_{12} °
$S_{1?}$ -> NP_{12} ° $VP_{2?}$

N_{23}
V_{23}
NP_{23} -> N_{23} °
VP_{23} -> V_{23} °

Problem with Top-Down Parsing

Left Recursion

Suppose you have A-> AB rule.

Then we will have the expansion as follows:

((A)K) -> ((AB)K) -> ((ABB)K) ……..

Top-Down Bottom-Up Chart Parsing

- Combines advantages of top-down & bottom-up parsing.
- Does not work in case of left recursion.
  - *e.g.* – "People laugh"
    - People – noun, verb
    - Laugh – noun, verb
  - Grammar – $S \rightarrow NP\ VP$

    $NP \rightarrow DT\ N \mid N$

    $VP \rightarrow V\ ADV \mid V$

Parse Trees for a Structurally Ambiguous Sentence

Let the grammar be –

$S \rightarrow NP\ VP$

$NP \rightarrow DT\ N \mid DT\ N\ PP$

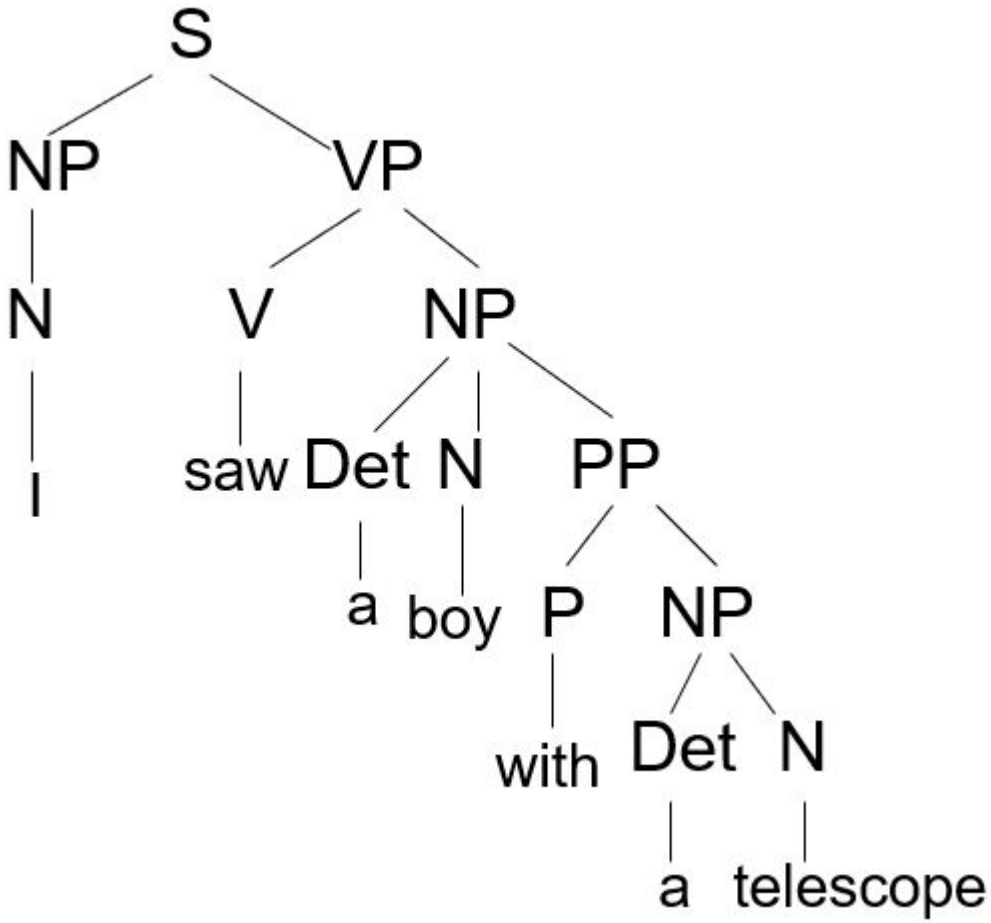
$PP \rightarrow P\ NP$

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

For the sentence,

"I saw a boy with a telescope"

Parse Tree - 1



Parse Tree -2