CS626: Speech, NLP and the Web

Dependency Parsing, Technique of Probabilistic Parsing, Difficult Parsing Phenomena

Pushpak Bhattacharyya

Computer Science and Engineering Department

IIT Bombay

Week of 28th September, 2020

Agenda for the week

- Dependency Parsing (DP), Need for DP
- DP algorithms
- Probabilistic DP

Difference between "Discriminative" and "Generative" Models

- Historical reason
- Binary classification problem
- Want to decide if a patient has cancer based on different "features" from the reports
- $Argmax_D(P(D|S))$
- *D* takes values 'Y' and 'N'
- Decide 'Y' if $P(D=Y|S) > P(=N|S)$, else 'N'

Discriminative Model

- Compute $P(D|S)$ directly
- "Features" from reports, $S = \{F_1, F_2, F_3, \ldots, F_K\}$ (like, fever, weight loss, hair loss, haemoglobin level etc.)
- P(D=Y|<fever, weight loss, hair loss, haemoglobin level,…>)
- We are discriminating, i.e., differentiating wrt the features input

Generative Model

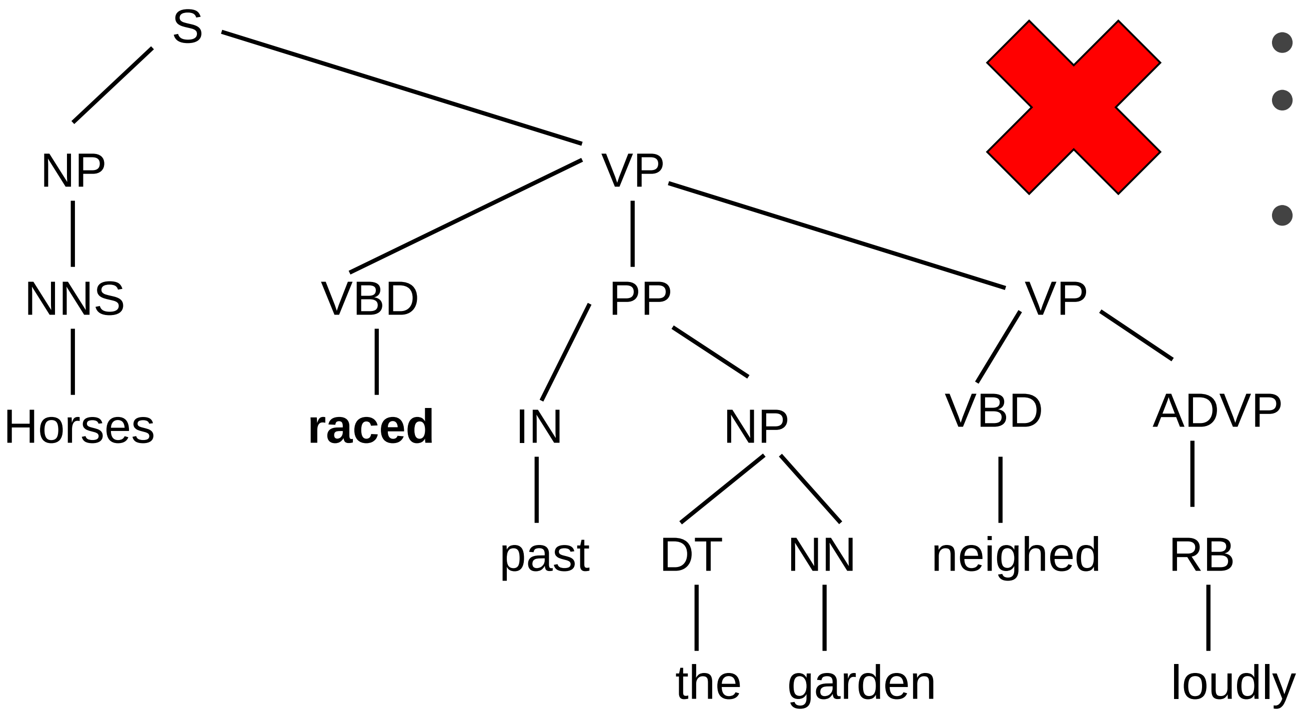
- Compute $P(D)$ and $P(S|D)$ and take product
- For P(D) we will need the proportion of cancer patients in the population (obtained via sampling)
- For the likelihood, we will make use of naïve Bayes assumption and require values of $P(F_i|D)$, e.g., what is the probability of a cancer patient having fever
- Hence the “discrimination” is not direct!!

Garden path phenomenon

Ellipsis

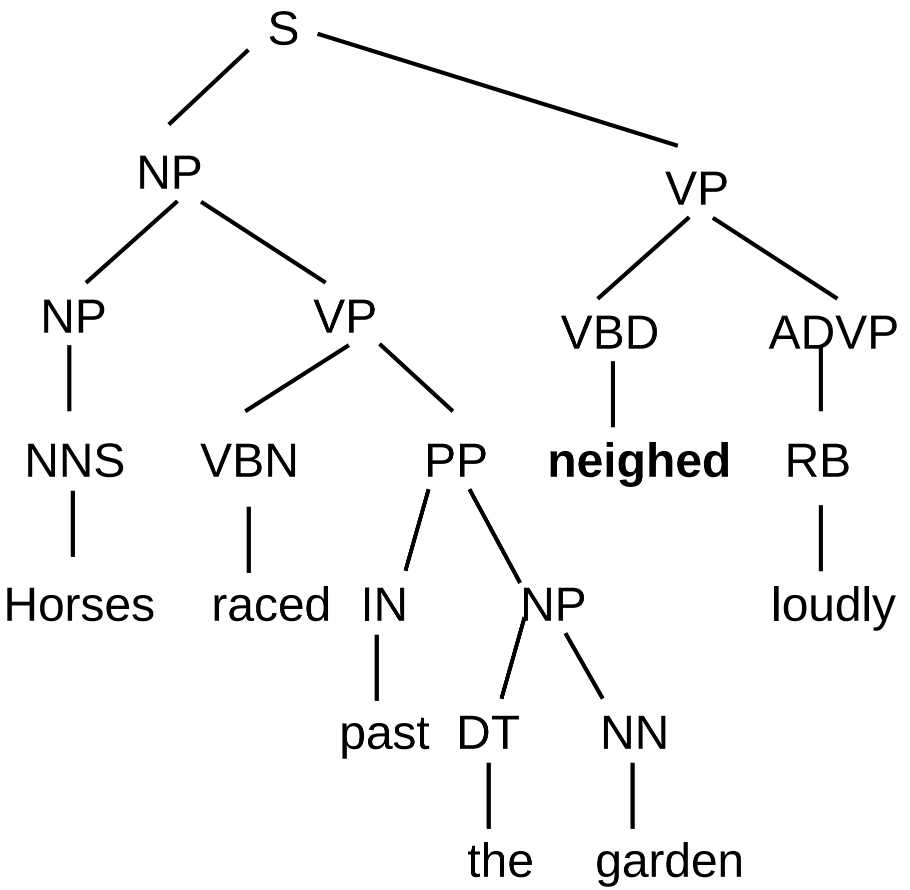
- Text is dropped (filled in the mind of the recipient by context)
- Example
  - *Horses raced past the garden neighed loudly.*
    - *Horses, **which were** raced past the garden, neighed loudly*
    - *Subject NP: Horses raced past the garden*
  - *Ram reads*
    - *Ram reads **a book.***
    - “a book” or anything which can be read is implicit

Garden path sentences: Horses raced past the garden neighed loudly



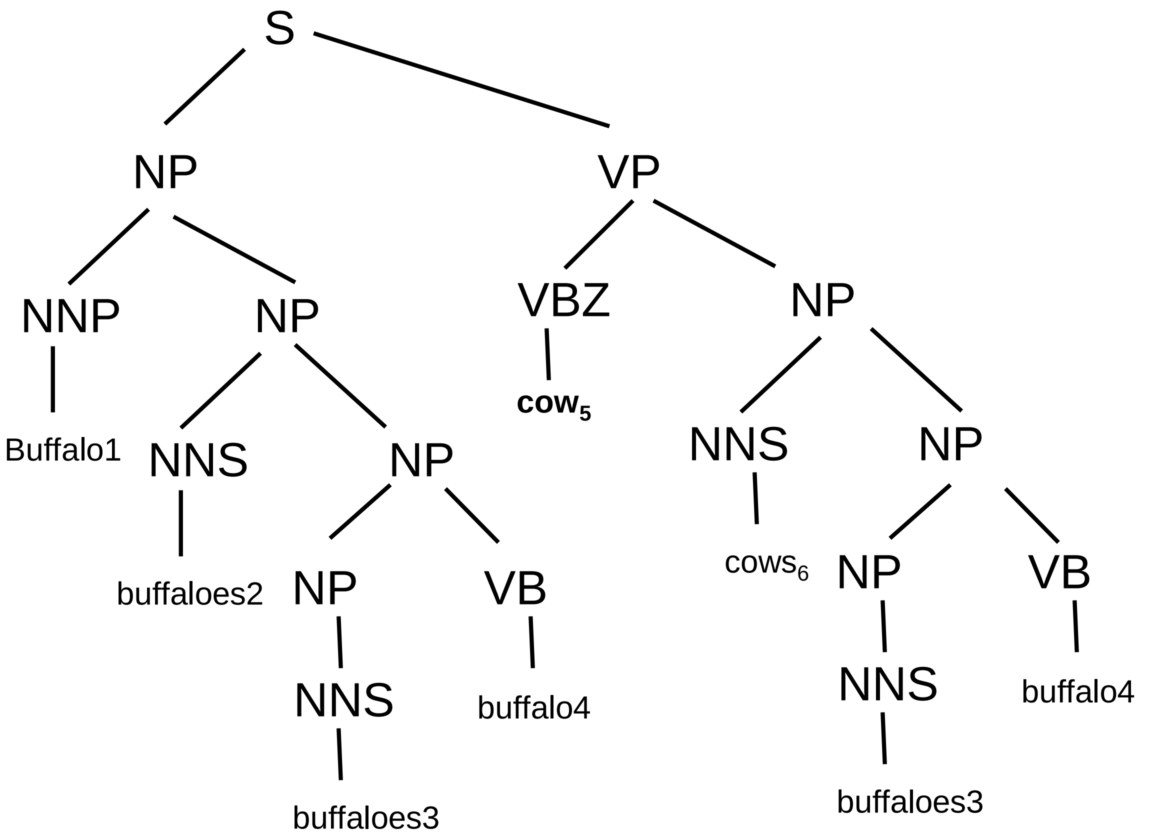
- Wrong parse
- “raced” can not be main verb
- Ungrammatical

Horses raced past the garden neighed loudly: Correct parse



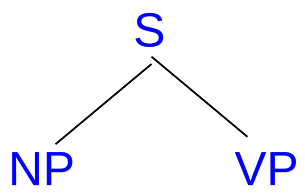
- Correct parse
- "raced" can not be main verb
- "neighed" is the main verb

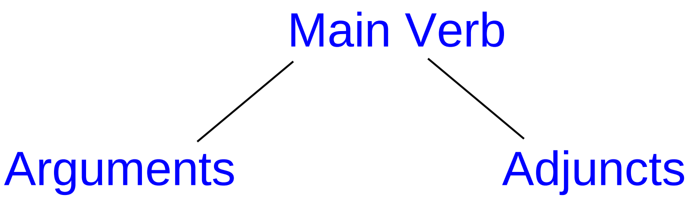
Buffalo$_1$ buffaloes$_2$ buffaloes$_3$ buffalo$_4$ cow$_5$ cows$_6$ buffaloes$_7$ buffalo$_8$



Need for dependency parsing

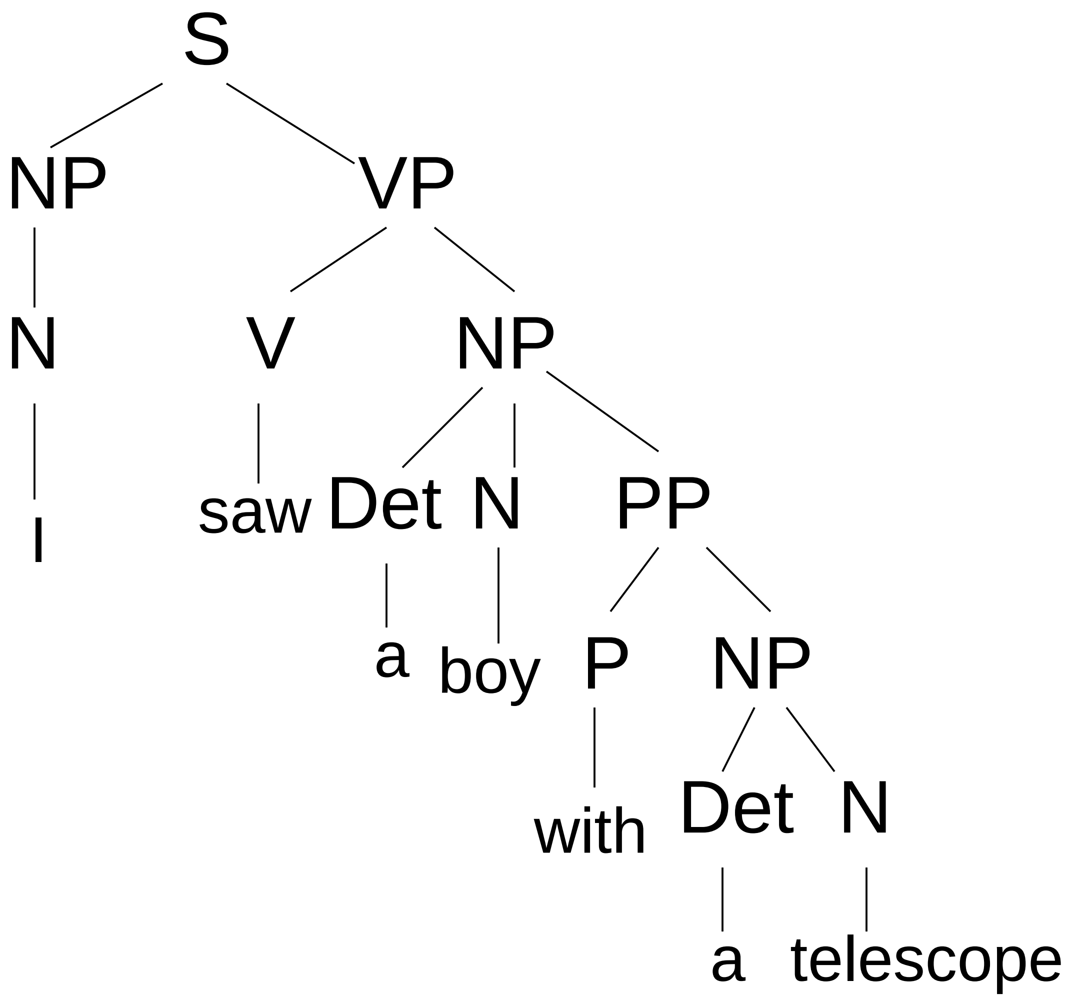
Two kinds of parse representations: Constituency Vs. Dependency



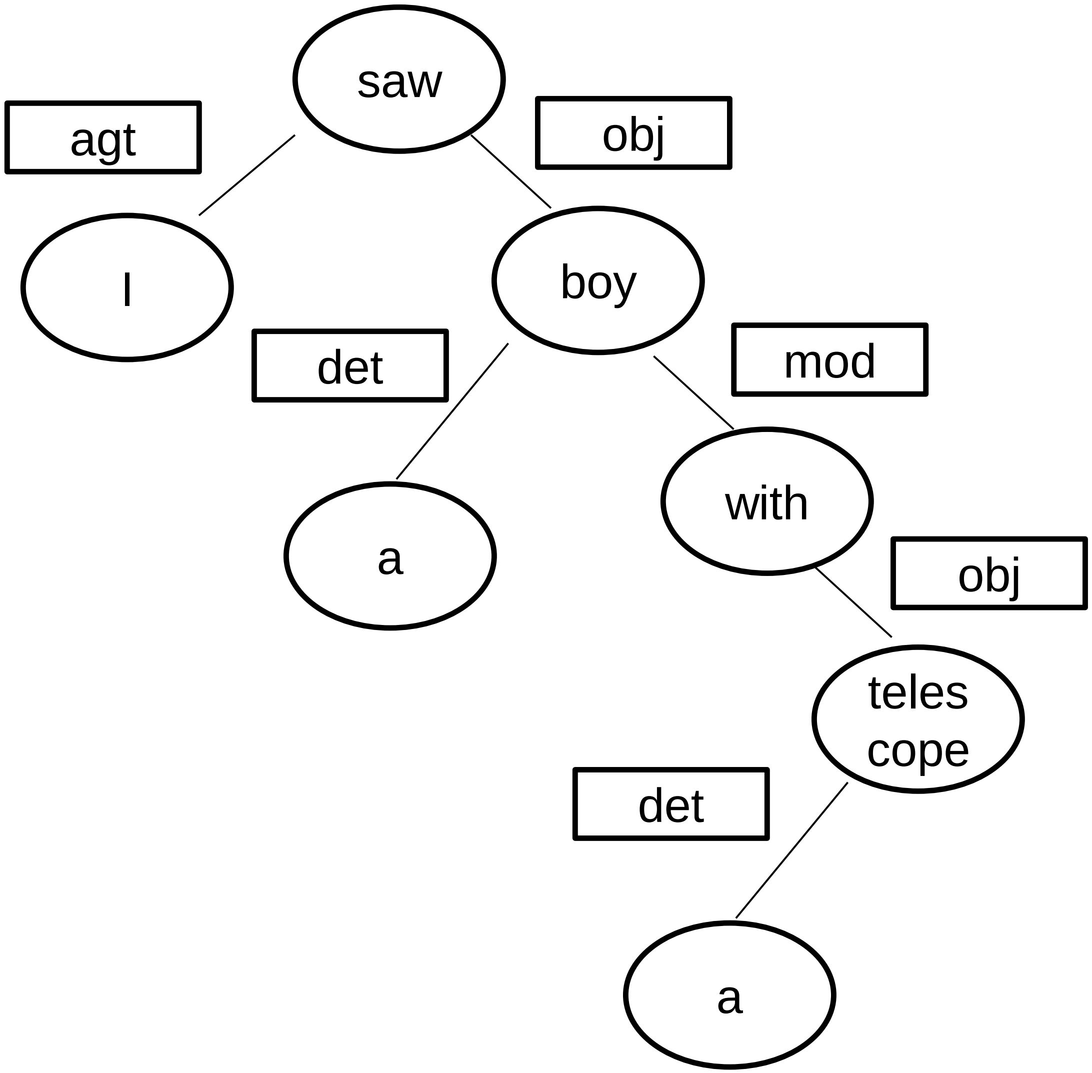


- Penn Constituency Treebank
  - http://www.cis.upenn.edu/~treebank/
- Prague Dependency Treebank
  - http://ufal.mff.cuni.cz/pdt2.0/

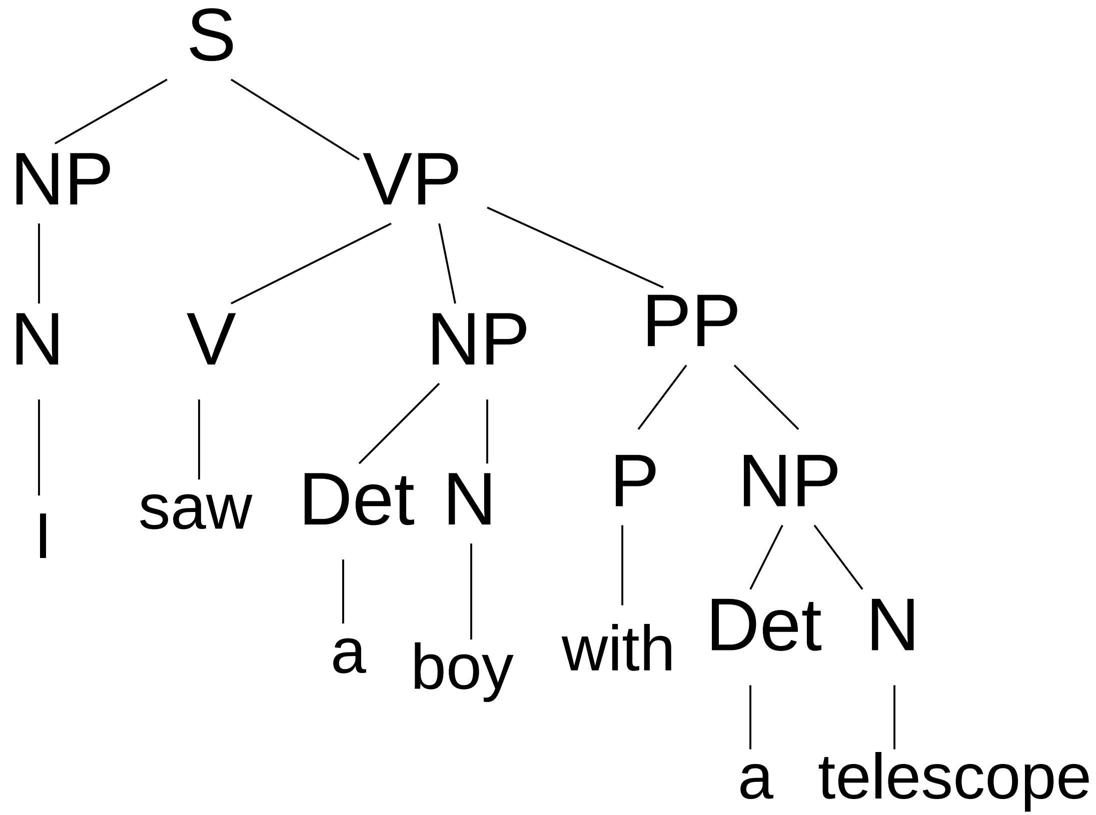
"I saw the boy with a telescope": Constituency parse-1: *telescope with boy*



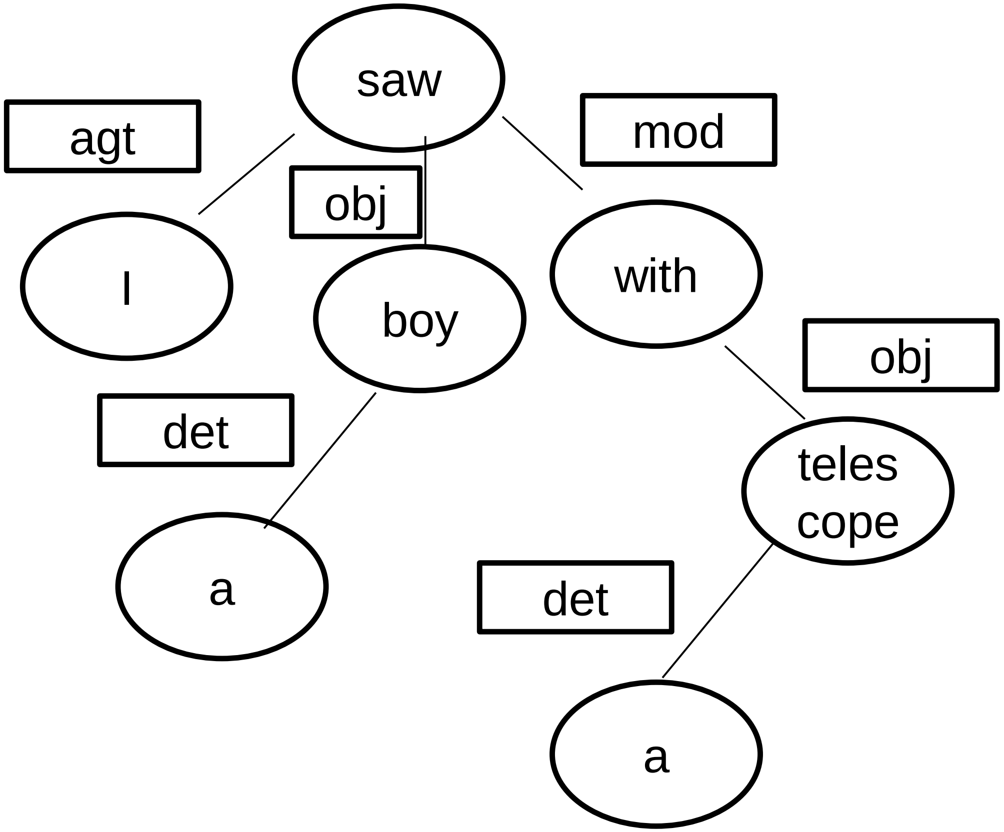
“I saw the boy with a telescope”: Dependency Parse Tree-1



Constituency Parse Tree-2: *telescope with me*



Dependency Parse Tree-2



Advantage of DP over CP

- Related entities are closer in DP than in CP: in terms of path length

- Free word order does not affect DP; CP needs additional rules

- Additional rules may overgeneralize!!

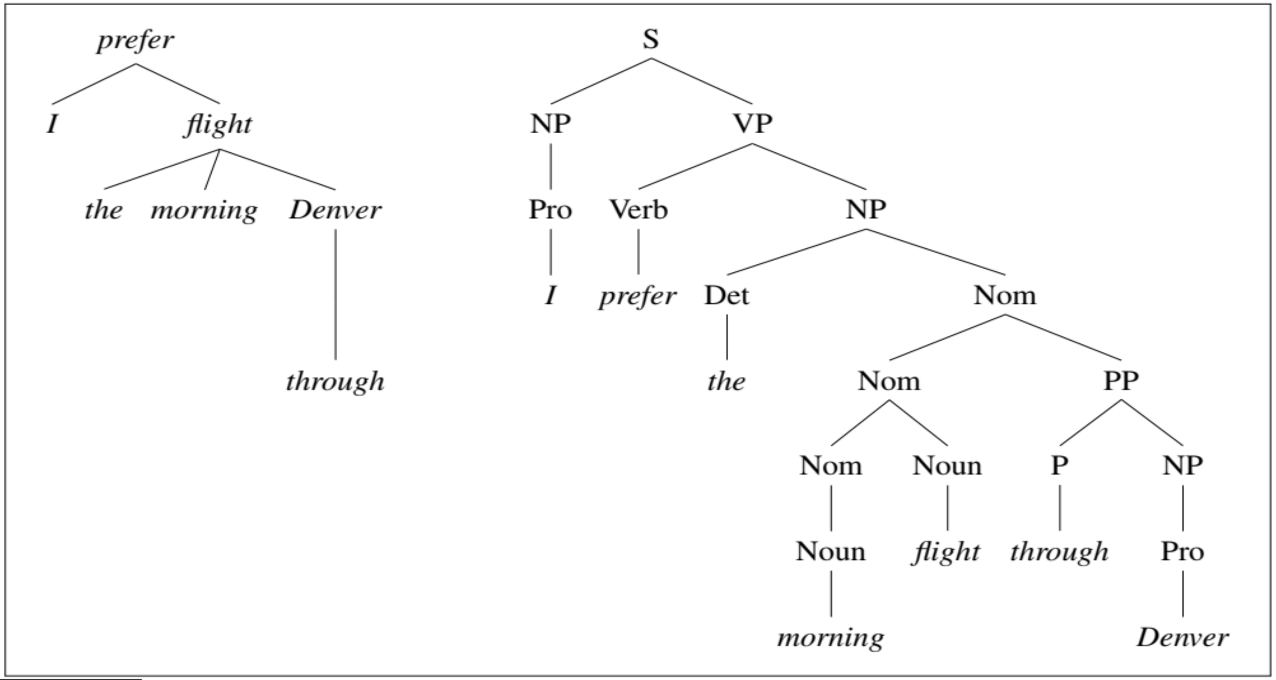
…CP needs additional rules

- *I saw the boy with a telescope*
  - *S → NP VP*
  - *VP → VBD NP PP*
- *With a telescope I saw the boy*
  - *S → NP VP*
  - *S → PP NP VP ???*

Impact of free order on constituency parsing

- Constituency parse fundamentally use adjacency information.
- Word order disturbs the adjacency
- Chomsky normal form demands that
  - The deduction should happen by linking together two adjacent entities.
- Example:
  - राम ने श्याम को देखा | (Ram ne Shyam ko dekha)
    - श्याम को देखा =VP
  - श्याम को राम ने देखा | (Shyam ko Ram ne dekha)
    - VP is discontinuous
    - Constituency parsing failed here
  - The agent and object is reversed in the above

Arguments are immediately linked



J & M, Chapter 15, 3rd Edition

Prefer: who prefers? “I”; what is preferred?: “flight”.

On the other hand, phrases are like *suitcases* that put all related things **at one place**: “The morning flight through Denver”

Subset of Dependency Relations: from Universal Dependency Project (Nivre et all 2016)

Clausal Argument Relations	Description
NSUBJ	Nominal subject
DOBJ	Direct object
IOBJ	Indirect object
CCOMP	Clausal complement
XCOMP	Open clausal complement
Nominal Modifier Relations	**Description**
NMOD	Nominal modifier
AMOD	Adjectival modifier
NUMMOD	Numeric modifier
APPOS	Appositional modifier
DET	Determiner
CASE	Prepositions, postpositions and other case markers
Other Notable Relations	**Description**
CONJ	Conjunct
CC	Coordinating conjunction

Examples to illustrate Dependency Relations

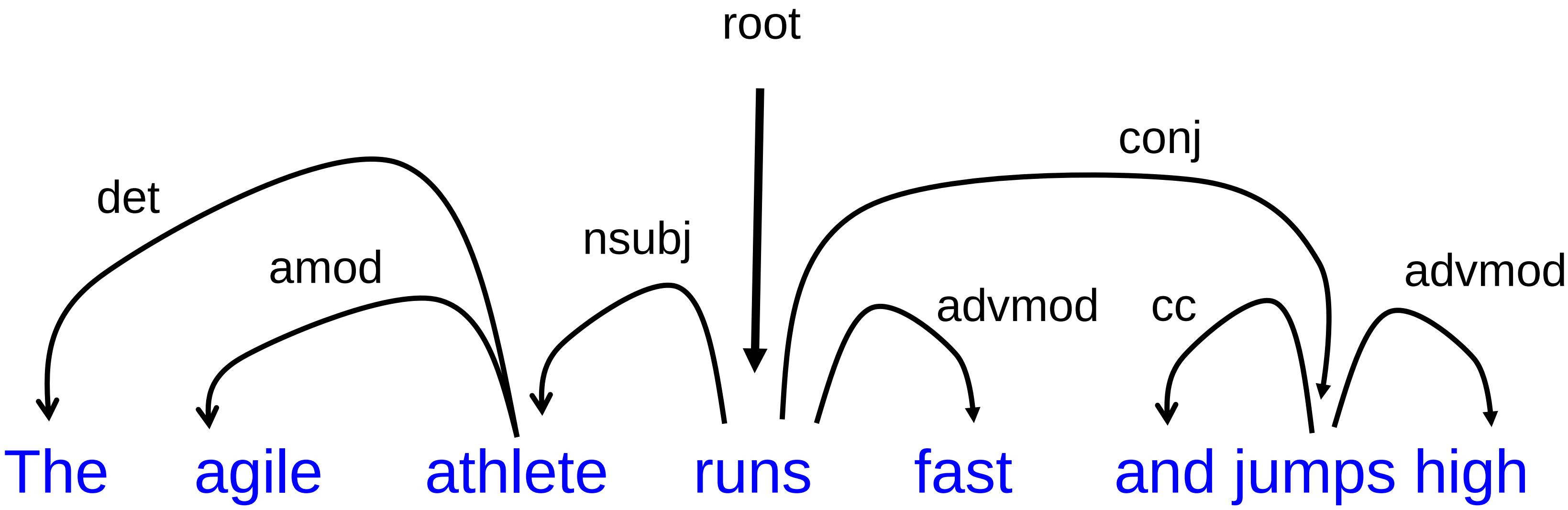
- NSUBJ, DOBJ, IOBJ- "*Ram gave a book to Shyam*"
  - Main Verb (MV): *gave*
  - NSUBJ: *Ram;* DOBJ: *book;* IOBJ: *Shyam*
- CCOMP, XCOMP: "I said that he should go", "I told him to go"
  - CCOMP: *said→go*
  - XCOMP: *told→go*

Illustration of DRs cntd.

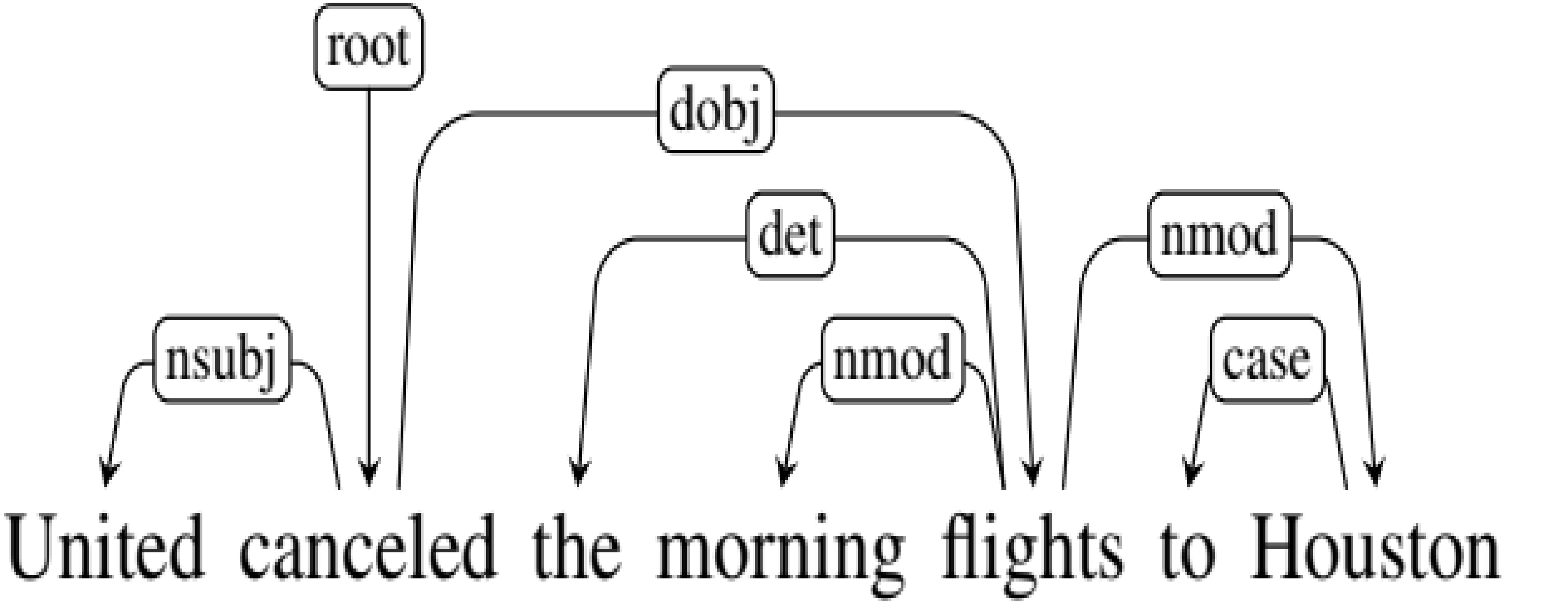
- NMOD (nominal modifier), AMOD (adjective modifier), NUMMOD (numerical modifier), APPOS (appositional modifier)
  - NMOD: *The bungalow of the Director: bungalow→Director*
  - AMOD: *The large bungalow: large→bungalow*
  - NUMMOD: *Three cups: three→cups*
  - APPOS: *covid19, the pandemic: covid19→pandemic*

Illustration of DRs cntd.

- DET (determiner), CASE (preposition, postposition and other case markers), CONJ (conjunct), CC (coordinating conjuct)
  - DET: *The bungalow: The→bungalow*
  - CASE: *The bungalow of Director: of→Director*
  - CONJ: *He is sincere and honest: sincere→honest*
  - CC: *He is sincere and honest: honest→and*


root
det
conj
nsubj
amod
advmod
advmod
cc
The
agile
athlete
runs
fast
and jumps high

Head → Modifier, e.g., *morning → flight*



Dependency Tree

- (1) There is a single designated root node that has no incoming arcs.
- (2) With the exception of the root node, each vertex has exactly one incoming arc.
- (3). There is a unique path from the root node to each vertex in V.

Statement of Assignment on Parsing (1/3)

- This assignment is on parsing. Its goal is to build a 2-way bridge between constituency parsing and dependency parsing.
- You are supposed to create a transformer from constituency parse (CP) to dependency parse (DP) and vice versa.

Statement of Assignment on Parsing (2/3)

Create a tool that will:

- (1) Input an English sentence
- (2) Obtain the CP output for the input sentence from any standard parser: Stanford, AllenNLP, NLTK, Spacy etc.
- (3) Convert the CP output to DP
- (4) Do steps 1-3 in the reverse direction: i.e., from DP to CP.

Statement of Assignment on Parsing (3/3)

- IMP: start with the simplest situation: *single subject- single verb- single object*, e.g., "students played football".
- Then gradually increase complexity:
  - "senior students", "senior students who had finished their exams",
  - "played energetically", "played energetically all day",
  - "street football", "street football with crowds watching" and so on.

Example: raw sentence

The strongest rain shut down the financial hub of Mumbai

(from: Stanford parser https://nlp.stanford.edu/software/lex-parser.shtml)

Example: POS Tagged sentence

The/DT strongest/JJS rain/NN shut/VBD down/RP the/DT financial/JJ hub/NN of/IN Mumbai/NNP

Constituency parse

```
(S
   (NP
         (DT The)
          (JJS strongest)
             (NN rain))
    )

 (VP
      …
```

```
(VP
   (VP
        (VBD shut)
        (PRT (RP down))
   (NP
       (NP
          (DT the) (JJ financial)
          (NN hub))
       (PP (IN of)
              (NP (NNP Mumbai)))))
```

Dependency Parse

root(ROOT-0, shut-4)
nsubj(shut-4, rain-3)
prt(shut-4, down-5)
det(rain-3, the-1)
amod(rain-3, strongest-2)

dobj(shut-4, hub-8)
det(hub-8, the-6)
amod(hub-8, financial-7)
prep(hub-8, of-9)
pobj(of-9, Mumbai-10)

Getting back to Probabilistic parsing

Data for ML based Parsing

[$_{S1}$[$_{S}$[$_{S}$[$_{VP}$[$_{VB}$Come]][$_{NP}$[$_{NNP}$July]]]]

[$_{,}$,]

[$_{CC}$ and]

[$_{S}$ [$_{NP}$ [$_{DT}$ the] [$_{JJ}$ IIT] [$_{NN}$ campus]]

[$_{VP}$ [$_{AUX}$ is]

[$_{ADJP}$ [$_{JJ}$ abuzz]

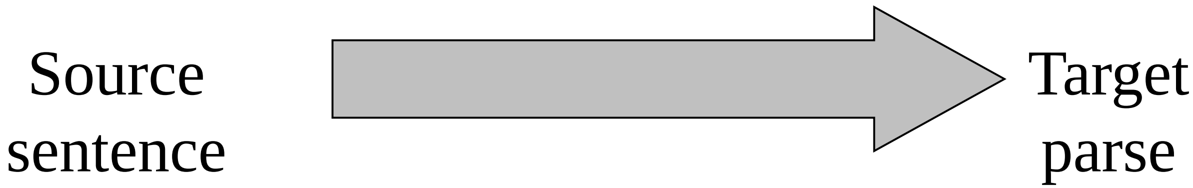
[$_{PP}$[$_{IN}$ with]

[$_{NP}$[$_{ADJP}$ [$_{JJ}$ new] [$_{CC}$ and] [$_{VBG}$ returning]]

[$_{NNS}$ students]]]]]]

[$_{.}$.]]]

Noisy Channel Modeling



$$T^* = \underset{T}{\operatorname{argmax}}\ [P(T|S)]$$

$$= \underset{T}{\operatorname{argmax}}\ [P(T).P(S|T)]$$

$$= \underset{T}{\operatorname{argmax}}\ [P(T)]$$, ***since given the parse the sentence is completely determined and P(S|T)=1***

Formal Definition of PCFG

- A PCFG consists of
  - A set of terminals $\{w_k\}$, k = 1,….,V

    $\{w_k\}$ = { child, teddy, bear, played…}
  - A set of non-terminals $\{N^i\}$, i = 1,…,n

    $\{N_i\}$ = { NP, VP, DT…}
  - A designated start symbol N^1
  - A set of rules $\{N^i \rightarrow \zeta^j\}$, where ζ^j is a sequence of terminals & non-terminals

    NP $\rightarrow$ DT NN
  - A corresponding set of rule probabilities

Rule Probabilities

- Rule probabilities are such that

$$\forall i \sum_{i} P(N^{i} \rightarrow \zeta^{j}) = 1$$

E.g., P(NP → DT NN) = 0.2
P(NP → NN) = 0.5
P(NP → NP PP) = 0.3

- P(NP → DT NN) = 0.2
  - Means 20 % of the training data parses use the rule NP → DT NN

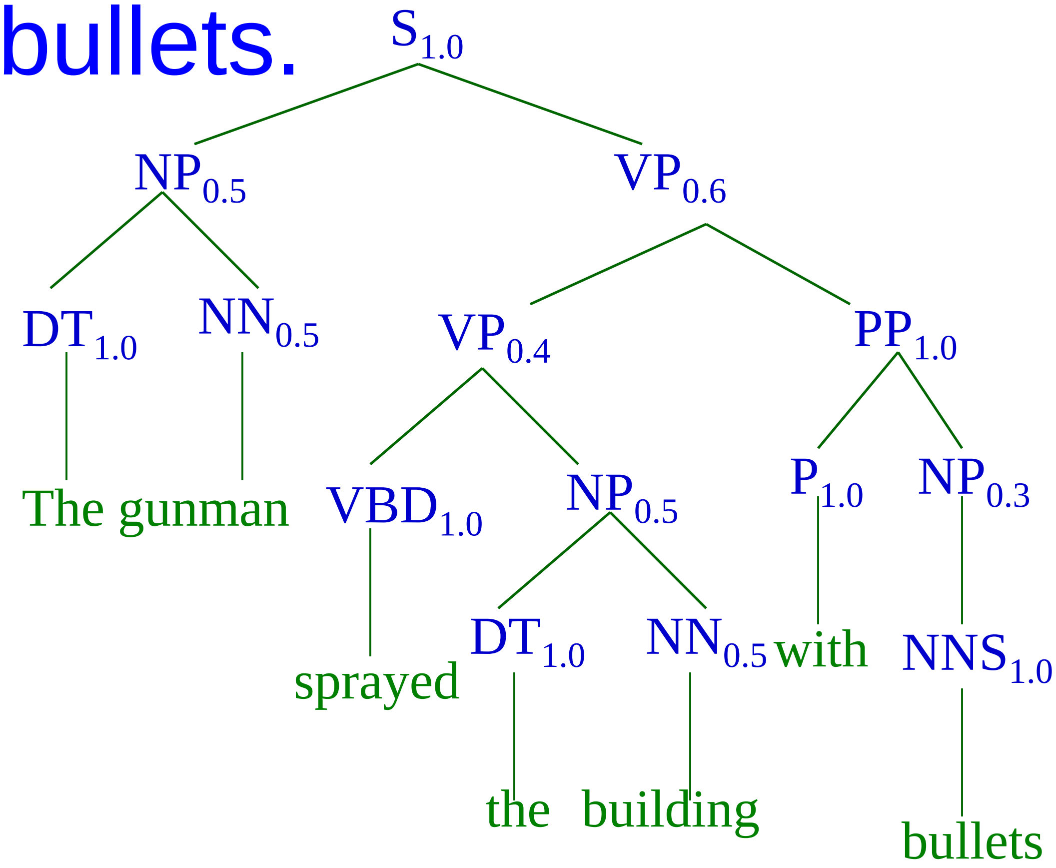
Probabilistic Context Free Grammars

- S → NP VP 1.0
- NP → DT NN 0.5
- NP → NNS 0.3
- NP → NP PP 0.2
- PP → P NP 1.0
- VP → VP PP 0.6
- VP → VBD NP 0.4

- DT → the 1.0
- NN → gunman 0.5
- NN → building 0.5
- VBD → sprayed 1.0
- NNS → bullets 1.0

Example Parse t_1`

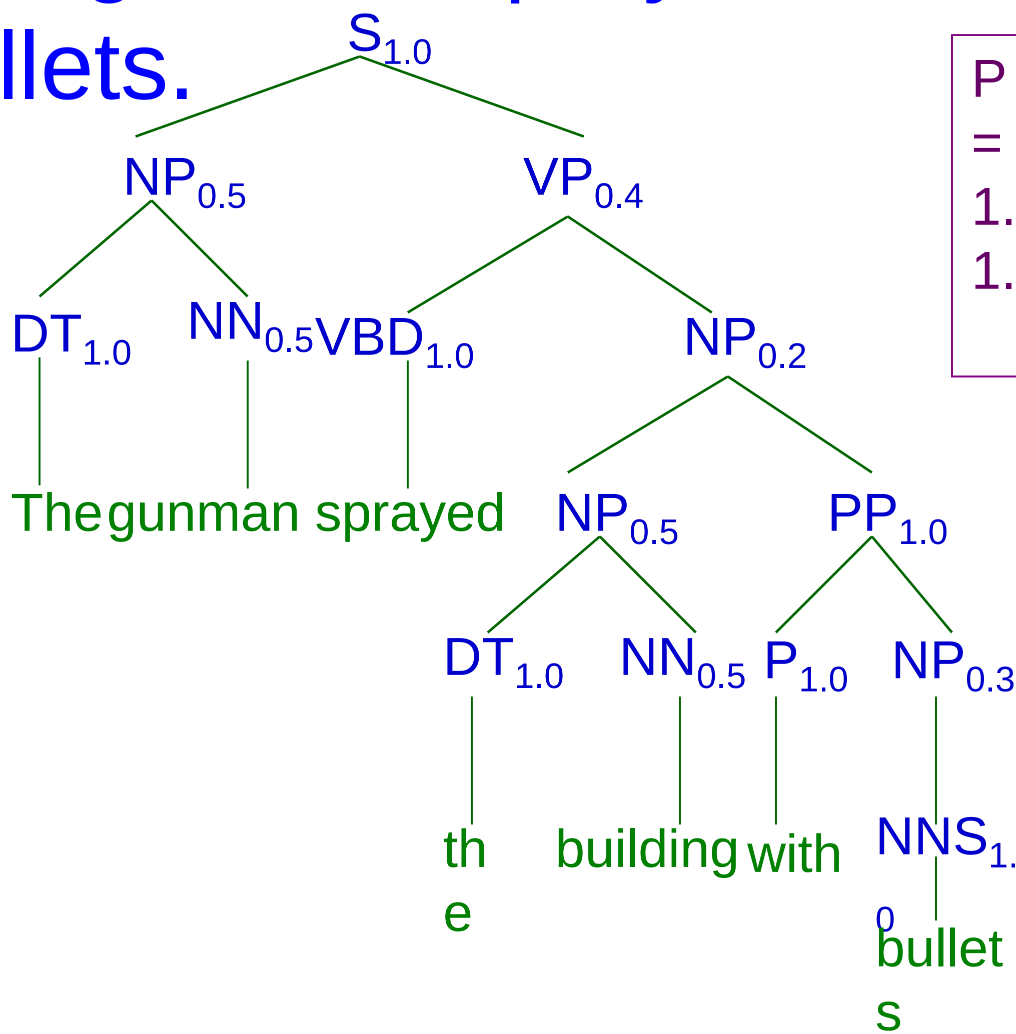
- The gunman sprayed the building with bullets.



P (t_1) = 1.0 * 0.5 * 1.0 * 0.5 * 0.6 * 0.4 * 1.0 * 0.5 * 1.0 * 0.5 * 1.0 * 1.0 * 0.3 * 1.0 = 0.00225

Another Parse t_2

- The gunman sprayed the building with bullets.



P (t_2)
= 1.0 * 0.5 * 1.0 * 0.5 * 0.4 * 1.0 * 0.2 * 0.5 * 1.0 * 0.5 * 1.0 * 1.0 * 0.3 * 1.0
= 0.0015

Assumptions of the PCFG model

- Place invariance :

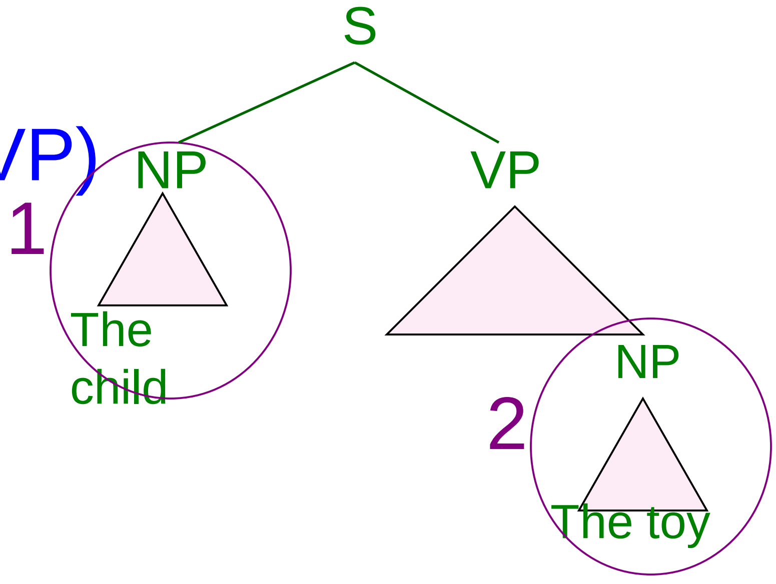
  P(NP $\rightarrow$ DT NN) is same in locations 1 and 2

- Context-free :

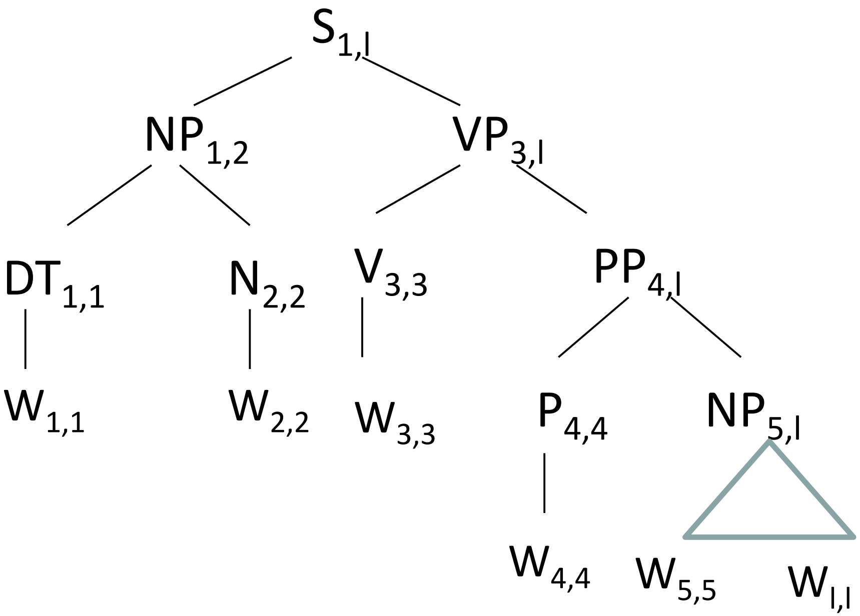
  P(NP $\rightarrow$ DT NN | anything outside “The child”)
  = P(NP $\rightarrow$ DT NN)

- Ancestor free : At 2,

  P(NP $\rightarrow$ DT NN|its ancestor is VP)
  = P(NP $\rightarrow$DT NN)



Probability of a parse tree (cont.)



$P(t|s) = P(t \mid S_{1,l})$

$= P(NP_{1,2}, DT_{1,1}, w_{1,1},$
$N_{2,2}, w_{2,2}$
$VP_{3,l}, V_{3,3}, w_{3,3}$
$PP_{4,l}, P_{4,4}, w_{4,4}\ NP_{5,l}, w_{5\ldots l} \mid S_{1,l})$

$= P(NP_{1,2}, VP_{3,l} \mid S_{1,l}) * P(DT_{1,1}, N_{2,2} \mid NP_{1,2}) *$
$P(w_{1,1} \mid DT_{1,1}) * P(w_{2,2} \mid N_{2,2}) * P(V_{3,3}, PP_{4,l} \mid VP_{3,l}) *$
$P(w_{3,3} \mid V_{3,3}) * P(P_{4,4}, NP_{5,l} \mid PP_{4,l}) * P(w_{4,4} \mid P_{4,4}) *$
$P(w_{5\ldots l} \mid NP_{5,l})$

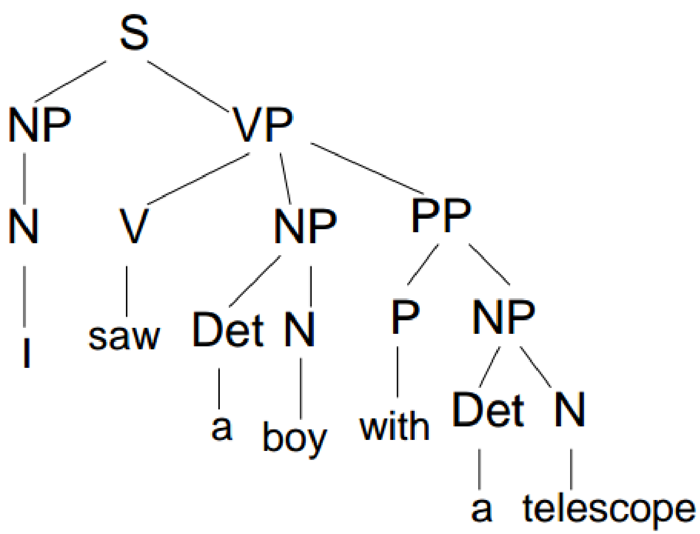
(Using Chain Rule, Context Freeness and Ancestor Freeness)

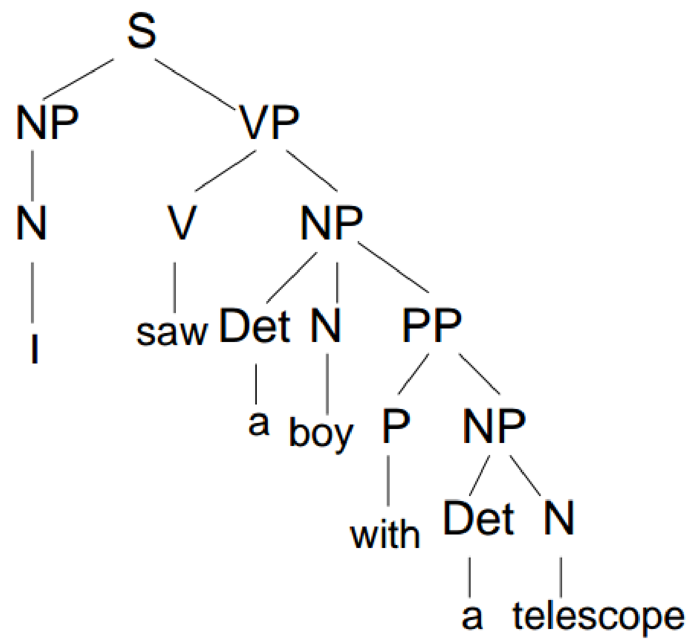
Domination

- A sentence is dominated by the symbol S through domination of segments by phrases
- Examples
  - The capital of a country dominates the whole country.
  - The capital of a state dominates the whole state.
  - The district headquarter dominates the district.
  - IIT Bombay is dominated by the administration of IIT Bombay.
  - Administration dominates Heads of Depts
  - The department is dominated by head of the department.

Ambiguity in determining domination

I saw a boy with a telescope.





- “saw” dominated by VP
- “a boy” dominated by NP
- “with a telescope” dominated by PP
- Yield of first NP is “a telescope”

- “saw” dominated by VP
- “with a telescope” dominated by PP
- “a boy with a telescope” dominated by NP
- Yield of NP is a “a boy with a telescope”

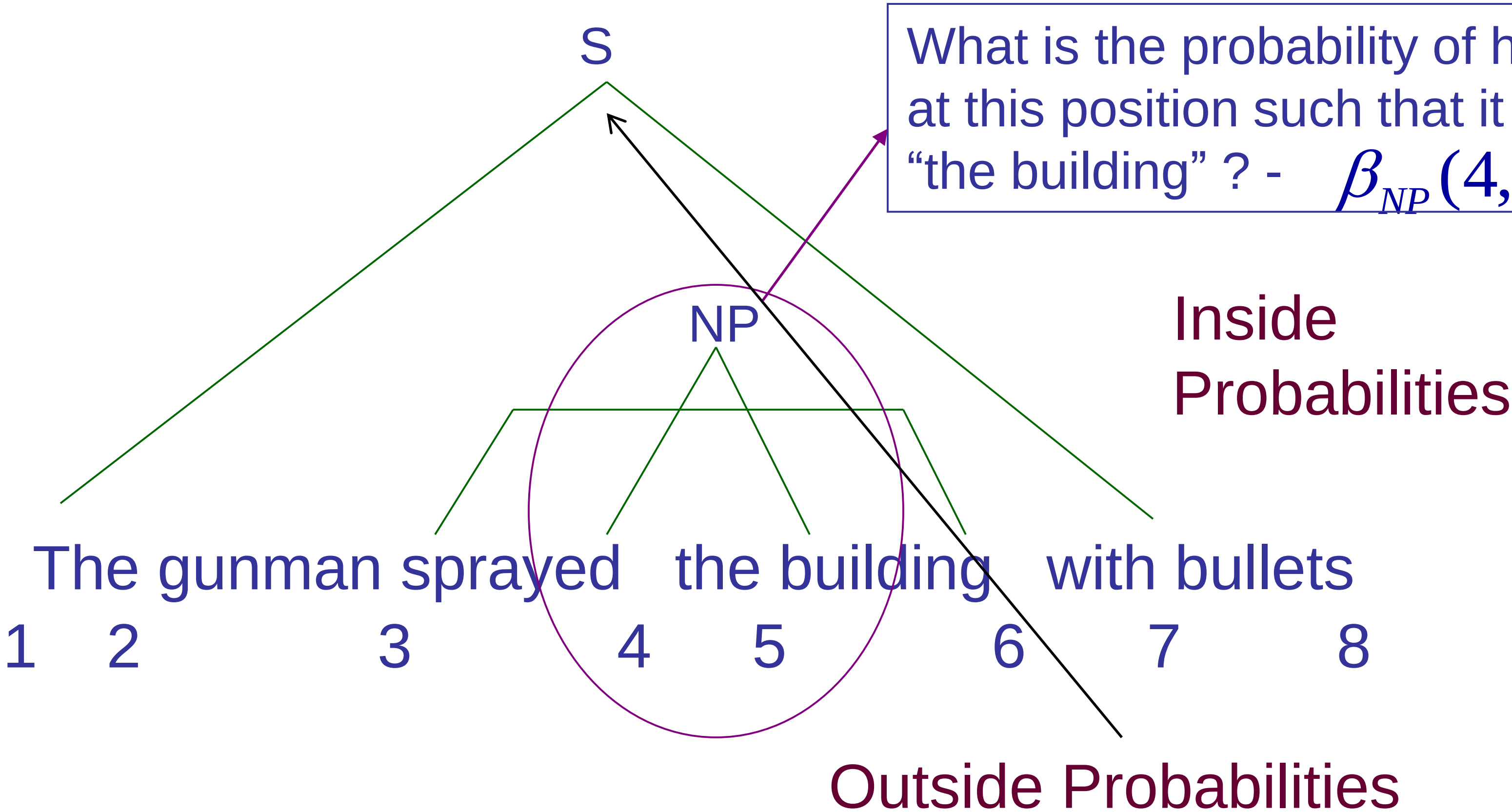
Main task in probabilistic parsing

- Main Intuition
  - Resolving the uncertainty
    - which non-terminal dominates how much territory in the sentence.
- The ambiguity in determining
  - The yield of NP
  - Will the NP dominate "a boy" or "a boy with a telescope"

Crucial Probabilities

Interesting Probabilities

What is the probability of having a NP at this position such that it will derive "the building" ? - $\beta_{NP}(4,5)$



What is the probability of starting from S and deriving "The gunman sprayed", a NP and "with bullets" ? - $\alpha_{NP}(4,5)$

Parse tree for the given sentence using probabilistic CYK parsing

$_0$ **The** $_1$ **gunman** $_2$ **sprayed** $_3$ **the** $_4$ **building** $_5$ **with** $_6$ **bullets** $_7$

- Two parse trees are possible because the sentence has attachment ambiguity .

- Total 16 multiplications are required to make both the parse trees using probabilistic CYK.

- Number of multiplications is less in comparison to a probabilistic parsing which prepares the two parse trees independently with 28 multiplication.

	The 1	gunman 2	Sprayed 3	the 4	Building 5	with 6	Bullets 7
0	β_{DT} (0-1) =1.0	β_{NP} (0-2) =0.25					β_{S}(0-7) =0.006
1		β_{NN} (1-2) =0.5					
2			β_{VBD}(2-3) =1.0		β_{VP} (2-5) =0.1		β_{VP}(2-7) =0.024
3				β_{DT}(3-4) =1.0	β_{NP} *(3-5) =0.25*		β_{NP}(3-7) =0.015
4					β_{NN} (4-5) =0.5		
5						β_{P}(5-6) =1.0	β_{PP}(5-7) =0.3
6							$\beta_{NP/NNS}$(6-7) =1.0

Calculation of values for each non terminal occuring in the CYK table

β_{DT} (0-1) =1.0   (From Grammar rules)

β_{NN} (1-2) =0.5   (From Grammar rules)

β_{NP} (0-2) = P(the gunman | NP_{0-2} , G)
= P(NP->DT NN)* β_{DT} (0-1) * β_{NN} (1-2)
= 0.5 * 1.0 * 0.5
=0.25

β_{VBD}(2-3) =1.0   (From Grammar rules)

β_{DT}(3-4) =1.0   (From Grammar rules)

β_{NN} (4-5)   =0.5   (From Grammar rules)

β_{NP} (3-5) = P(the building | NP_{3-5} , G)
= P(NP->DT NN)* β_{DT} (3-4) * β_{NN} (4-5)
= 0.5 * 1.0 * 0.5
=0.25

$\beta_{VP}(2\text{-}5) = P(VP\text{->}VBD\ NP) * \beta_{VBD}(2\text{-}3) * \beta_{NN}(3\text{-}5)$

$= 0.4 * 1 * 0.25$

$= 0.1$

$\beta_{P}(5\text{-}6) = 1.0$ (From Grammar rules)

$\beta_{NP/NNS}(6\text{-}7) = 1.0$ (From Grammar rules)

$\beta_{PP}(5\text{-}7) = P(PP\text{->}P\ NP) * \beta_{P}(5\text{-}6) * \beta_{NP/NNS}(6\text{-}7)$

$= 1.0 * 1.0 * 0.3$

$= 0.3$

$\beta_{NP}(3\text{-}7) = P(NP\text{->}NP\ PP) * \beta_{NP}(3\text{-}5) * \beta_{PP}(5\text{-}7)$

$= 0.2 * 0.25 * 0.3$

$= 0.015$

$\beta_{VP}(2\text{-}7) = (P(VP\text{->}VBD\ NP) * \beta_{VBD}(2\text{-}3) * \beta_{NP}(3\text{-}7) + P(VP\text{->}VP\ PP) * \beta_{VP}(2\text{-}5) * \beta_{PP}(5\text{-}7))$

$= 0.4 * 1 * 0.015 + 0.6 * 0.1 * 0.3$

$= 0.024$

$\boldsymbol{\beta_{S}(0\text{-}7) = P(S\text{->}NP\ VP) * \beta_{NP}(0\text{-}2) * \beta_{VP}(2\text{-}7)}$

$\boldsymbol{= 1 * 0.25 * 0.024}$

$\boldsymbol{= 0.006}$

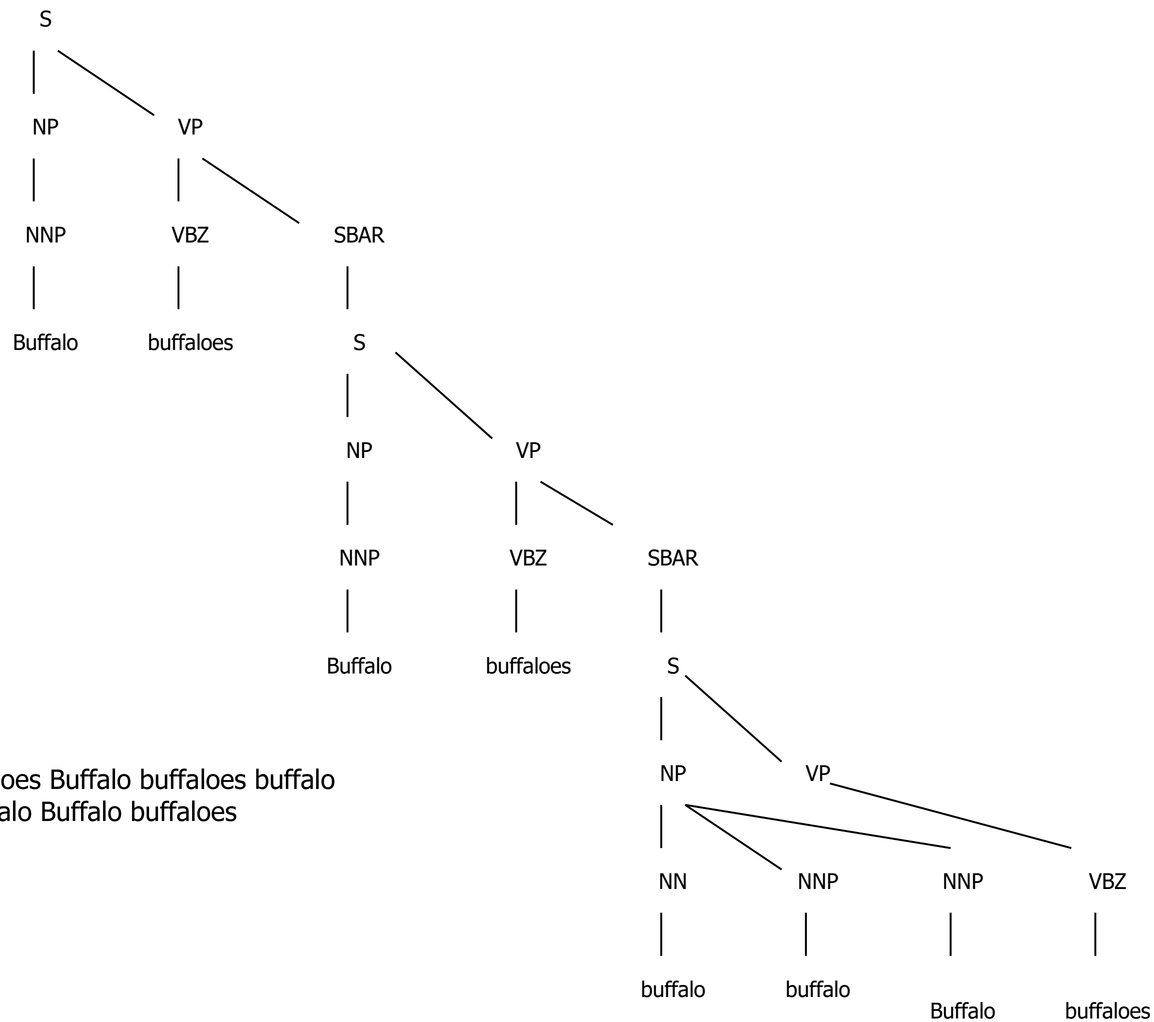
A very difficult parsing situation!

Repeated Word handling

Sentence on Buffaloes!

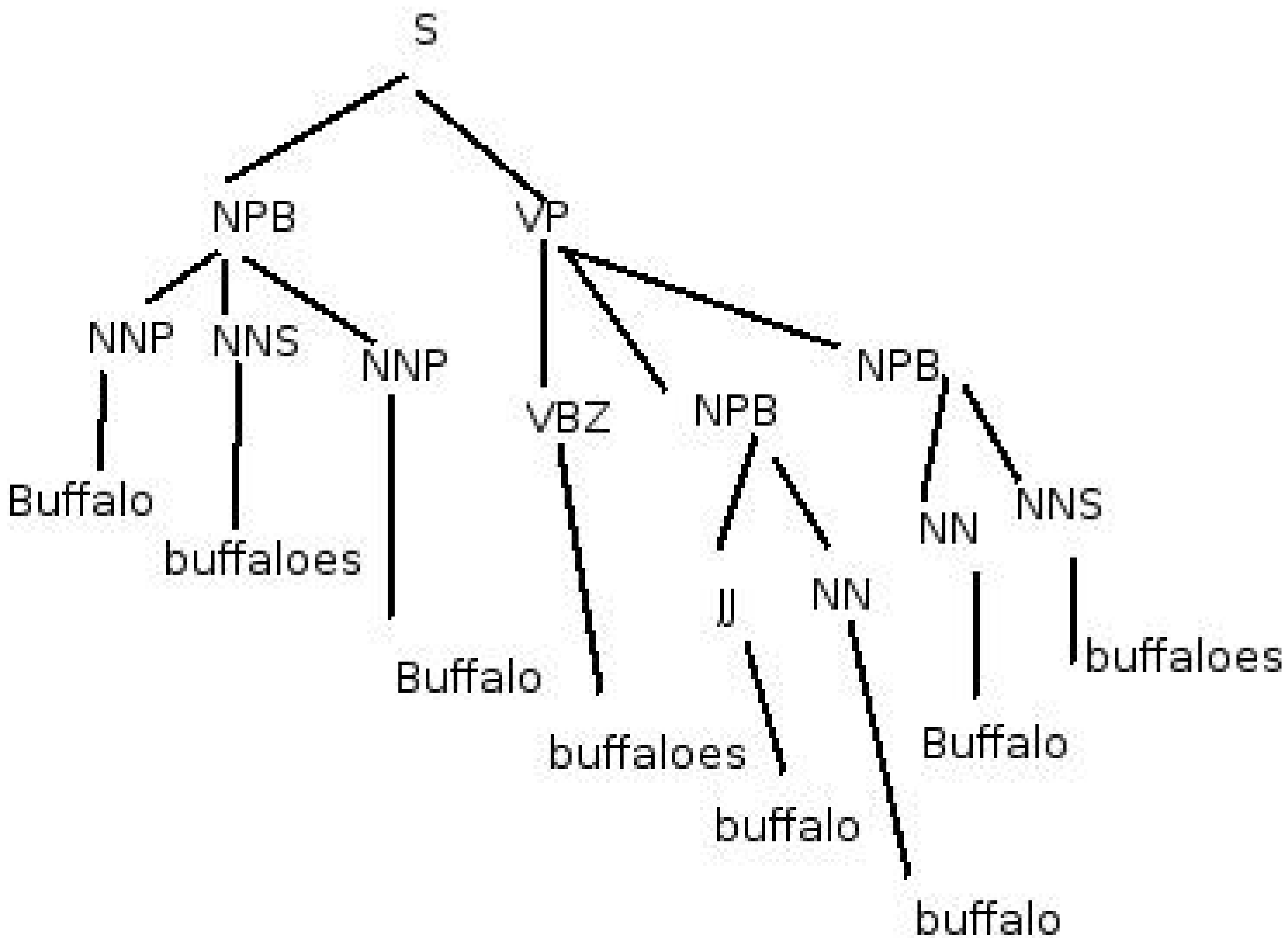
Buffaloe buffaloes Buffaloe buffaloes buffaloe buffaloe Buffaloe buffaloes

Charniak

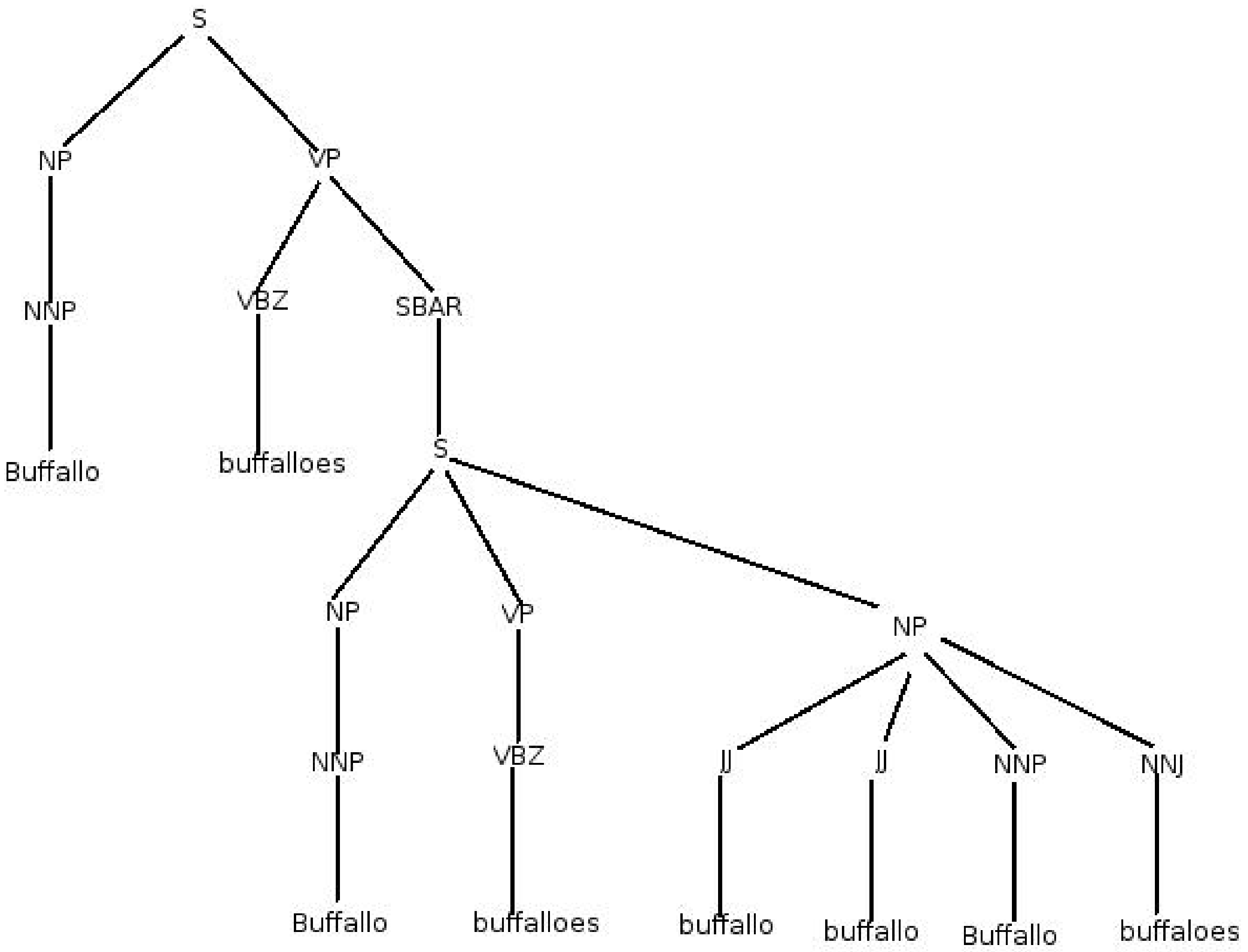


Buffalo buffaloes Buffalo buffaloes buffalo buffalo Buffalo buffaloes

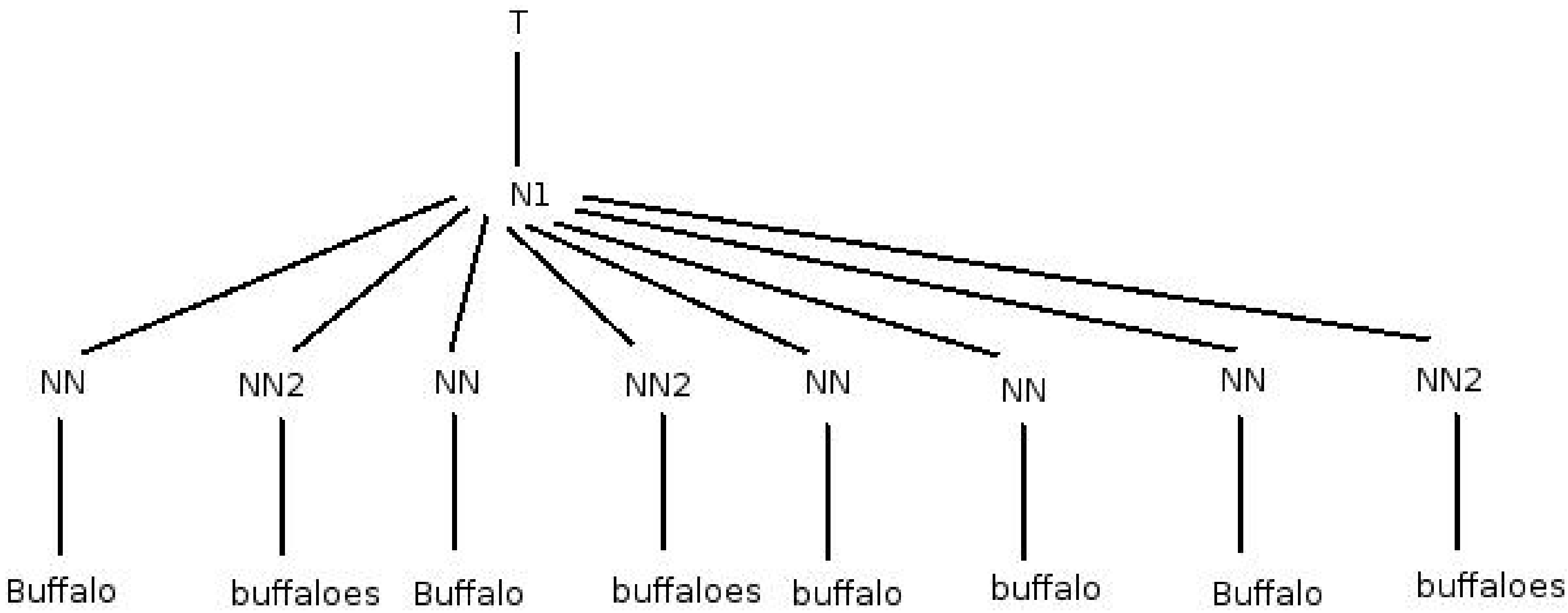
Collins


S
NPB
VP
NNP
NNS
NNP
VBZ
NPB
NPB
Buffalo
buffaloes
JJ
NN
NN
NNS
Buffalo
buffaloes
buffaloes
Buffalo
buffalo
buffalo

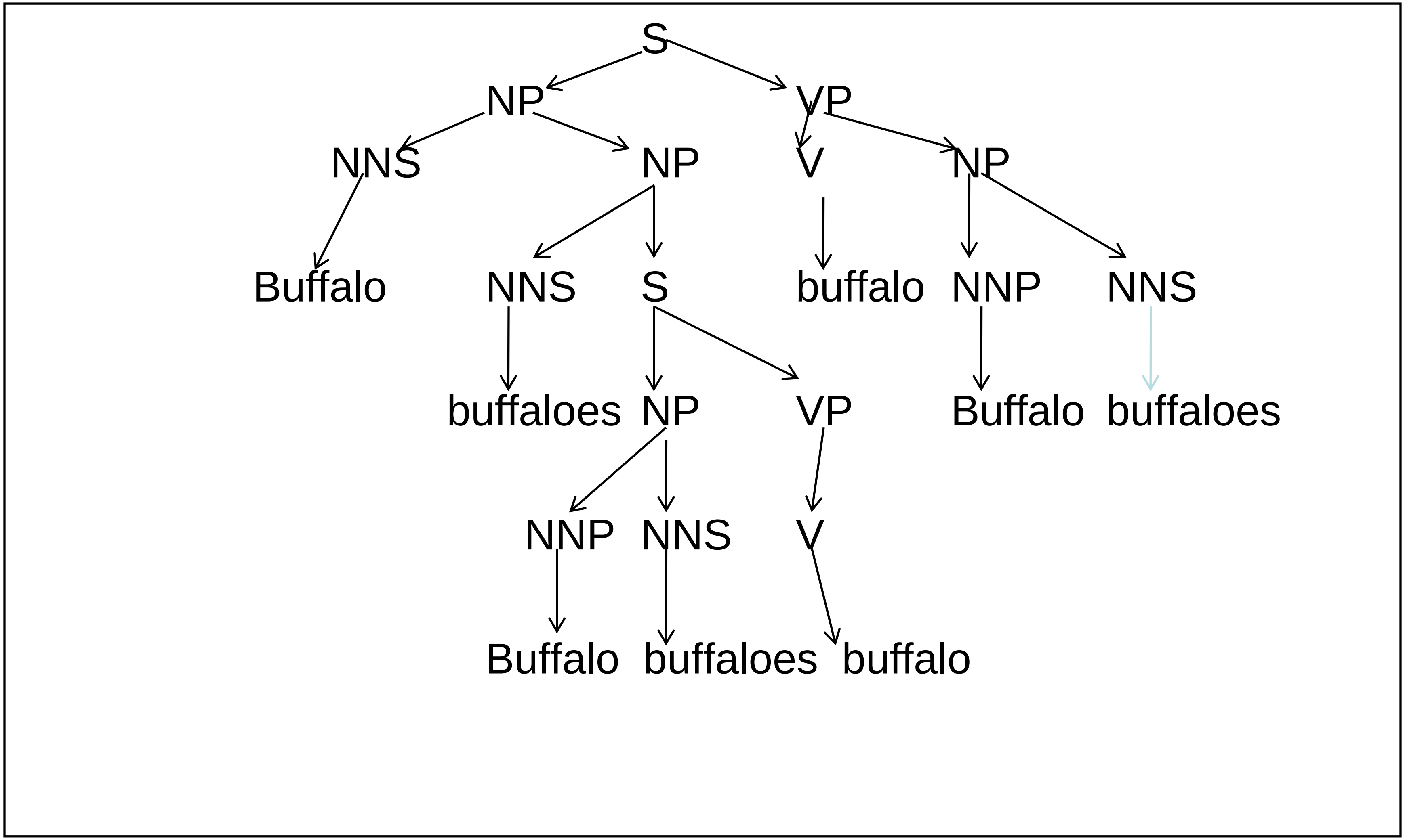
Stanford


S
NP
VP
NNP
VBZ
SBAR
Buffallo
buffalloes
S
NP
VP
NP
NNP
VBZ
JJ
JJ
NNP
NNJ
Buffallo
buffalloes
buffallo
buffallo
Buffallo
buffaloes

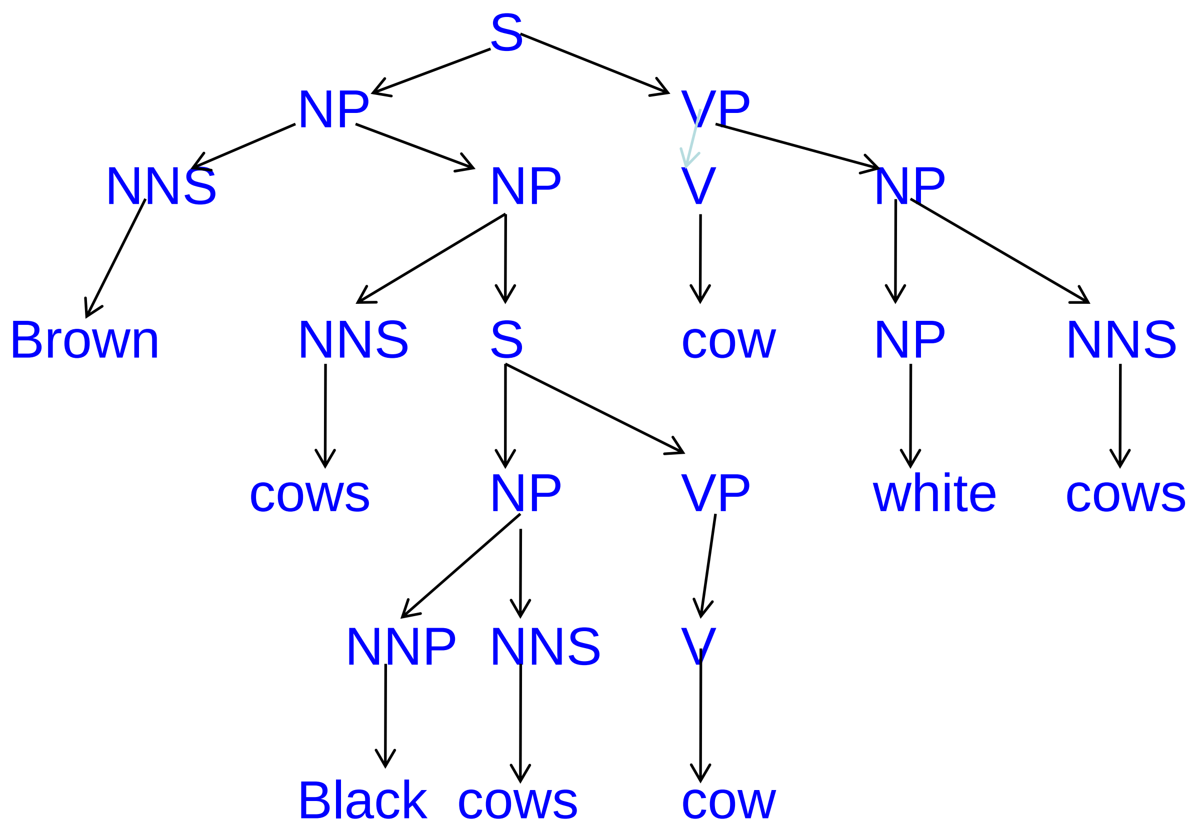
RASP


T
N1
NN
NN2
NN
NN2
NN
NN
NN
NN2
Buffalo
buffaloes
Buffalo
buffaloes
buffalo
buffalo
Buffalo
buffaloes

Correct parse



Another sentence of same structure

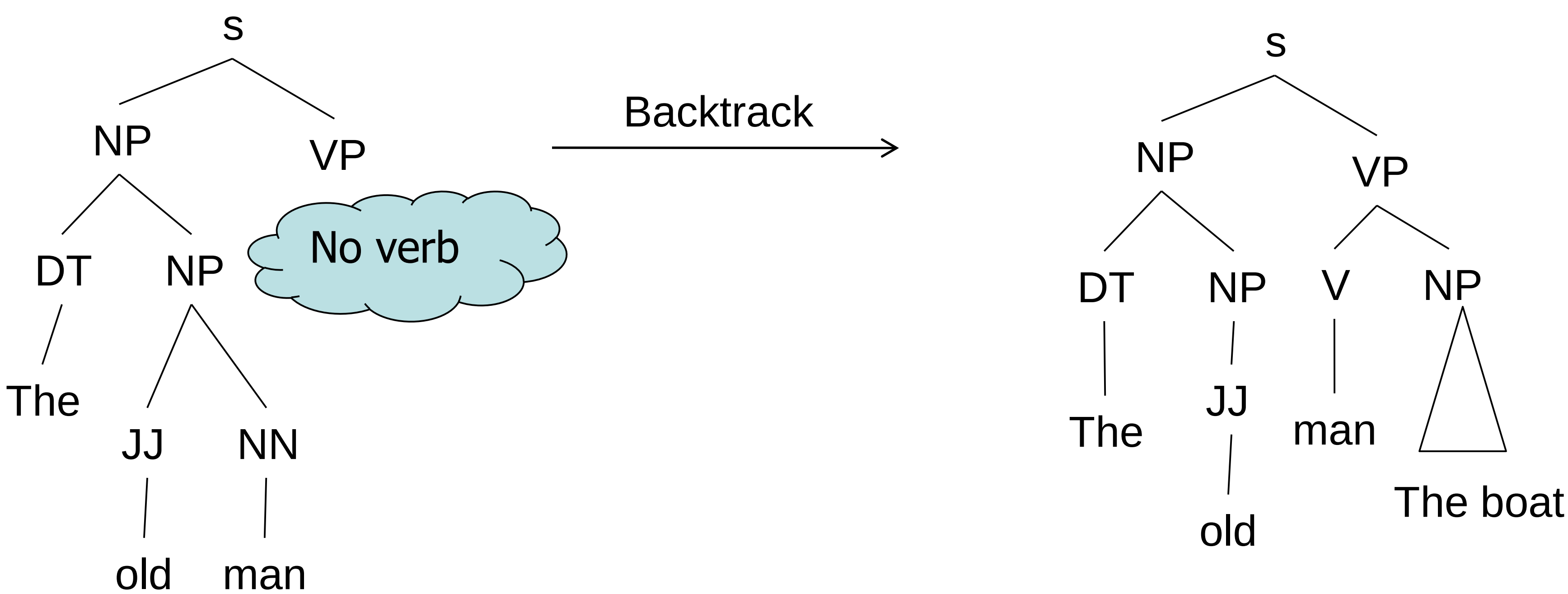


Observation

- Collins and Charniak come close to producing the correct parse.
- RASP tags all the words as nouns.

Another phenomenon: Garden pathing

e.g. The old man the boat.



Another example: The horse raced past the garden fell.