ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Phân tích cú pháp xác suất

Viện Công nghệ Thông tin và Truyền thông

Làm cách nào chọn cây đúng?

- Ví dụ:

  I saw a man with a telescope.

- Khi số luật tăng, khả năng nhập nhằng tăng
- Tập luật NYU: bộ PTCP Apple pie : 20,000-30,000 luật cho tiếng Anh
- Lựa chọn luật AD: V DT NN PP

  (1) VP $\rightarrow$ V NP PP

  NP $\rightarrow$ DT NN

  (2) VP $\rightarrow$ V NP

  NP $\rightarrow$ DT NN PP

Kết hợp từ (bigrams pr)

Ví dụ:

Eat ice-cream (high freq)

Eat John (low, except on Survivor)

Nhược điểm:

- P(John decided to bake a) có xác suất cao
- Xét:

$P(w_3) = P(w_3|w_2w_1)=P(w_3|w_2)P(w_2|w_1)P(w_1)$

Giả thiết này quá mạnh: chủ ngữ có thể quyết định bổ ngữ trong câu

Clinton admires honesty

➢ sử dụng cấu trúc ngữ pháp để dừng việc lan truyền

- Xét Fred watered his mother's small garden. Từ *garden* có ảnh hưởng như thế nào?
  - Pr(*garden*|*mother's small*) thấp ⇒ mô hình trigram không tốt
  - Pr(*garden* | X là thành phần chính của bổ ngữ cho động từ *to water*) cao hơn

➢ sử dụng bigram + quan hệ ngữ pháp

Kết hợp từ (bigrams pr)

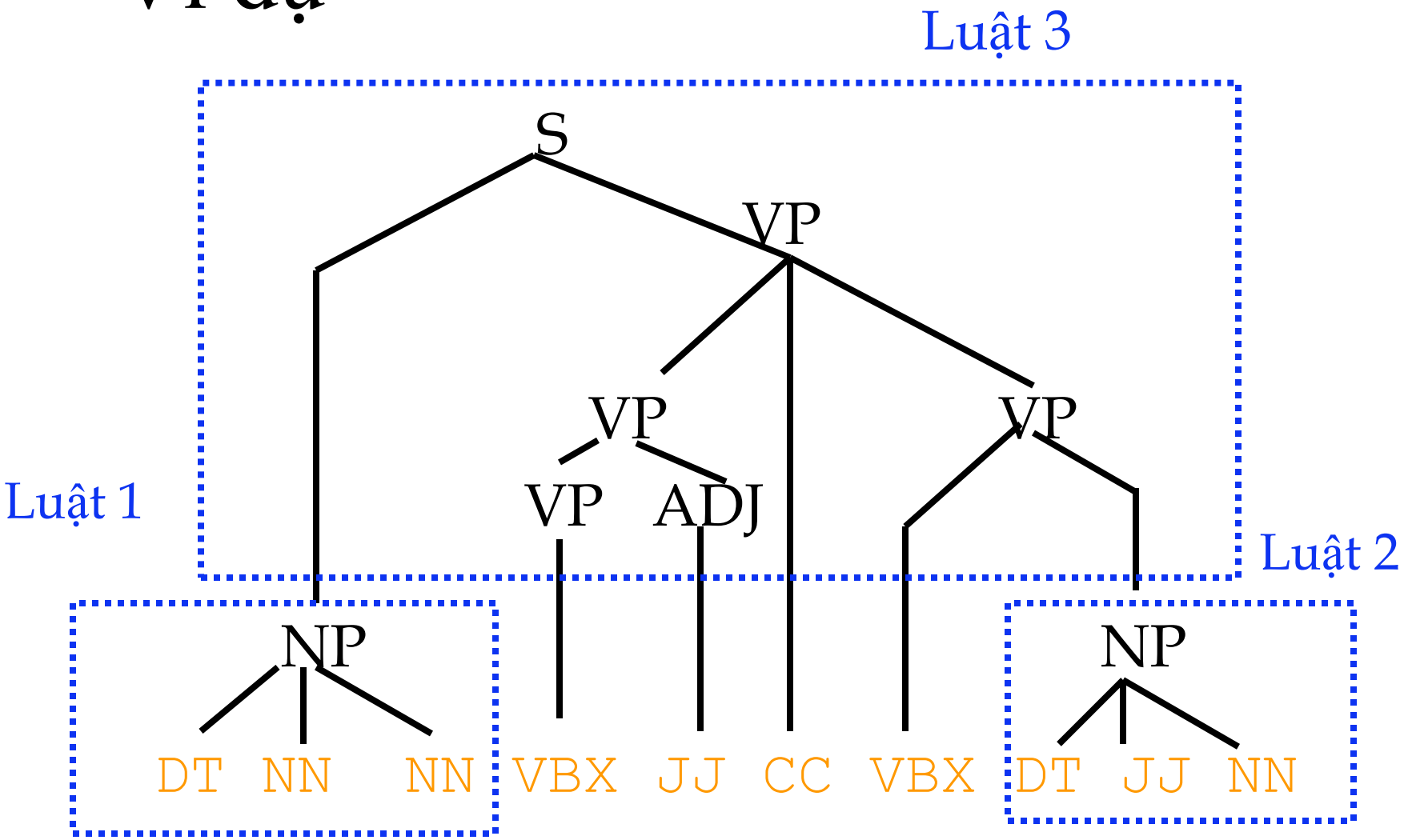
- V có một số loại bổ ngữ nhất định
  $\Rightarrow$ Verb-with-obj, verb-without-obj
- Sự tương thích giữa chủ ngữ và bổ ngữ:
  John admires honesty
  Honesty admires John ???

Nhược điểm:

- Kích thước tập ngữ pháp tăng
- Các bài báo của tạp chí Wall Street Journal trong 1 năm: 47,219 câu, độ dài trung bình 23 từ, gán nhãn bằng tay: chỉ có 4.7% hay 2,232 câu có cùng cấu trúc ngữ pháp

➢ Không thể dựa trên việc tìm các cấu trúc cú pháp đúng cho cả câu. Phải xây dựng tập các mẫu ngữ pháp nhỏ

Ví dụ


Luật 3
S
VP
VP
VP
VP
ADJ
Luật 1
Luật 2
NP
NP
DT NN NN
VBX JJ CC VBX
DT JJ NN
This apple pie looks good and is a real treat

Luật

1. NP→DT NN NN
2. NP→DT JJ NN
3. S→NP VBX JJ CC VBX NP

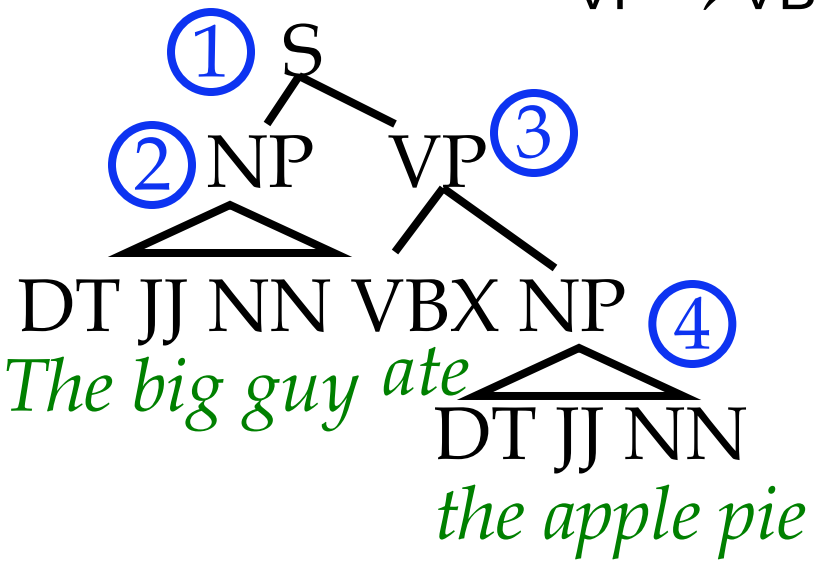
- Nhóm (NNS, NN) thành NX; (NNP, NNPs)=NPX; (VBP, VBZ, VBD)=VBX;
- Chọn các luật theo tần suất của nó

Tính xác suất

$$\Pr(X \rightarrow Y) = \frac{X \Rightarrow NP}{Y \quad NP} = \frac{DT\ JJ\ NN}{NP} = \frac{1470}{9711} = 0.1532$$

Tính Pr

S → NP VP; 0.35
NP → DT JJ NN; 0.1532
VP → VBX NP; 0.302



Luật áp dụng	*Chuỗi Pr*
1 S →NP VP	0.35
2 NP → DT JJ NN	0.1532 x 0.35 = 0.0536
3 VP → VBX NP	0.302 x 0.0536= 0.0162
4 NP → DT JJ NN	0.1532 x 0.0162=0.0025

Pr = 0.0025

Văn phạm phi ngữ cảnh xác suất

- 1 văn phạm phi ngữ cảnh xác suất (Probabilistic Context Free Grammar) gồm các phần thông thường của CFG
- Tập ký hiệu kết thúc $\{w^k\}$, $k = 1, \ldots, V$
- Tập ký hiệu không kết thúc $\{N^i\}$, $i = 1, \ldots, n$
- Ký hiệu khởi đầu N^1
- Tập luật $\{N^i \rightarrow \zeta^j\}$, ζ^j là chuỗi các ký hiệu kết thúc và không kết thúc
- Tập các xác suất của 1 luật là:

$$\forall i \sum_j P(N^i \rightarrow \zeta^j) = 1$$

- Xác suất của 1 cây cú pháp:

$$P(T) = \Pi_{i=1..n} p(r(i))$$

Các giả thiết

- ***Độc lập vị trí:*** Xác suất 1 cây con không phụ thuộc vào vị trí của các từ của cây con đó ở trong câu

  $\forall k, P(N_{jk}(k+c) \rightarrow \zeta)$ là giống nhau

- ***Độc lập ngữ cảnh:*** Xác suất 1 cây con không phụ thuộc vào các từ ngoài cây con đó

  $P(N_{jkl} \rightarrow \zeta |$ các từ ngoài khoảng k đến l) = $P(N_{jkl} \rightarrow \zeta)$

- ***Độc lập tổ tiên:*** Xác suất 1 cây con không phụ thuộc vào các nút ngoài cay con đó

  $P(N_{jkl} \rightarrow \zeta |$ các nút ngoài cây con N_{jkl}) = $P(N_{jkl} \rightarrow \zeta)$

Các thuật toán

- CKY
- Beam search
- Agenda/chart-based search
- …

CKY kết hợp xác suất

- Cấu trúc dữ liệu:
  - Mảng lập trình động π[i,*j*,*a*] lưu xác suất lớn nhất của ký hiệu không kết thúc a triển khai thành chuỗi i…*j*.
  - Backptrs lưu liên kết đến các thành phần trên cây
- Ra: Xác suất lớn nhất của cây

Tính Pr dựa trên suy diễn

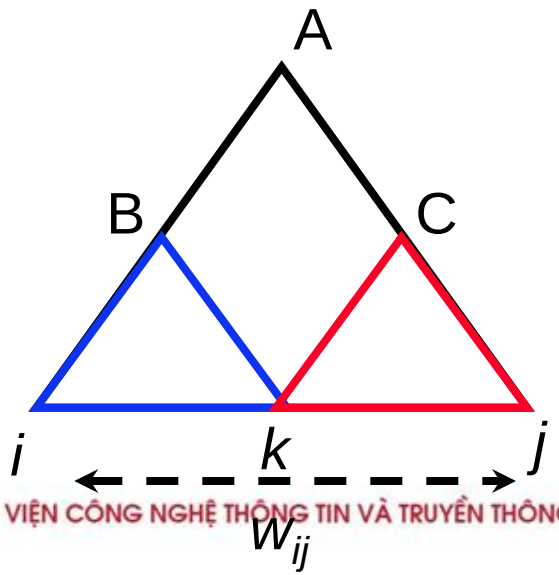
- Trường hợp cơ bản: chỉ có 1 từ đầu vào

  $Pr(tree) = pr(A \rightarrow w_i)$

- Trường hợp đệ qui: Đầu vào là xâu các từ

  $A \Rightarrow^{*} w_{ij}$ if $\exists k: A \rightarrow BC, B \Rightarrow^{*} w_{ik}, C \Rightarrow^{*} w_{kj}, i \leq k \leq j.$

  $p[i,j] = \max(p(A \rightarrow BC) \times p[i,k] \times p[k,j])$.



function CYK(*words*,*grammar*) **returns** *best_parse*

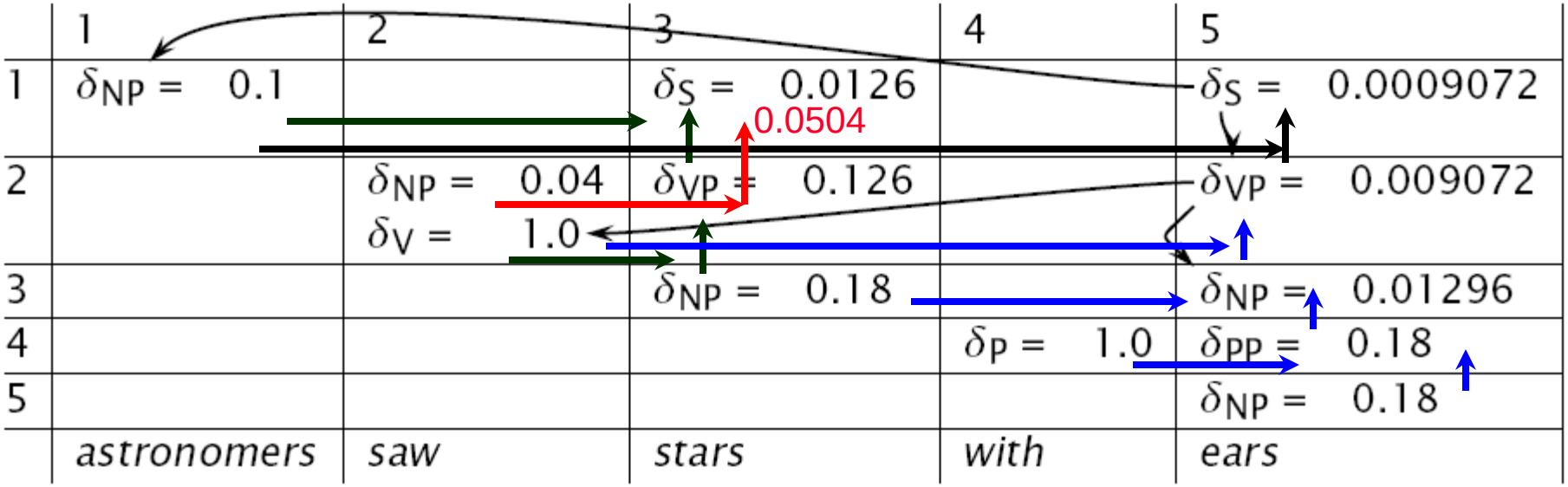
Create and clear p[*num_words*,*num_words*,*num_nonterminals*]

base case
for $i = 1$ **to** *num_words*
  for $A = 1$ **to** *num_nonterminals*
    if $A \rightarrow w_i$ is in grammar **then**
      $\pi[i, i, A] = P(A \rightarrow w_i)$

recursive case
for $j = 2$ **to** *num_words*
  for $i = 1$ **to** *num_words*-j+1
    for $k = 1$ **to** j-1
      for $A = 1$ **to** *num_nonterminals*
      for $B = 1$ **to** *num_nonterminals*
      for $C = 1$ **to** *num_nonterminals*
        $prob = \pi[i, k, B] \times p[i+k, j-k, C] \times P(A \rightarrow BC)$
        if $(prob > \pi[i, j, A])$ **then**
          $\pi[i, j, A] = \text{prob}$
          $B[i, j, A] = \{k, A, B\}$

Tính xác suất Viterbi (thuật toán CKY)

S → NP VP	1.0	NP → NP PP	0.4
PP → P NP	1.0	NP → *astronomers*	0.1
VP → V NP	0.7	NP → *ears*	0.18
VP → VP PP	0.3	NP → *saw*	0.04
P → *with*	1.0	NP → *stars*	0.18
V → *saw*	1.0	NP → *telescopes*	0.1



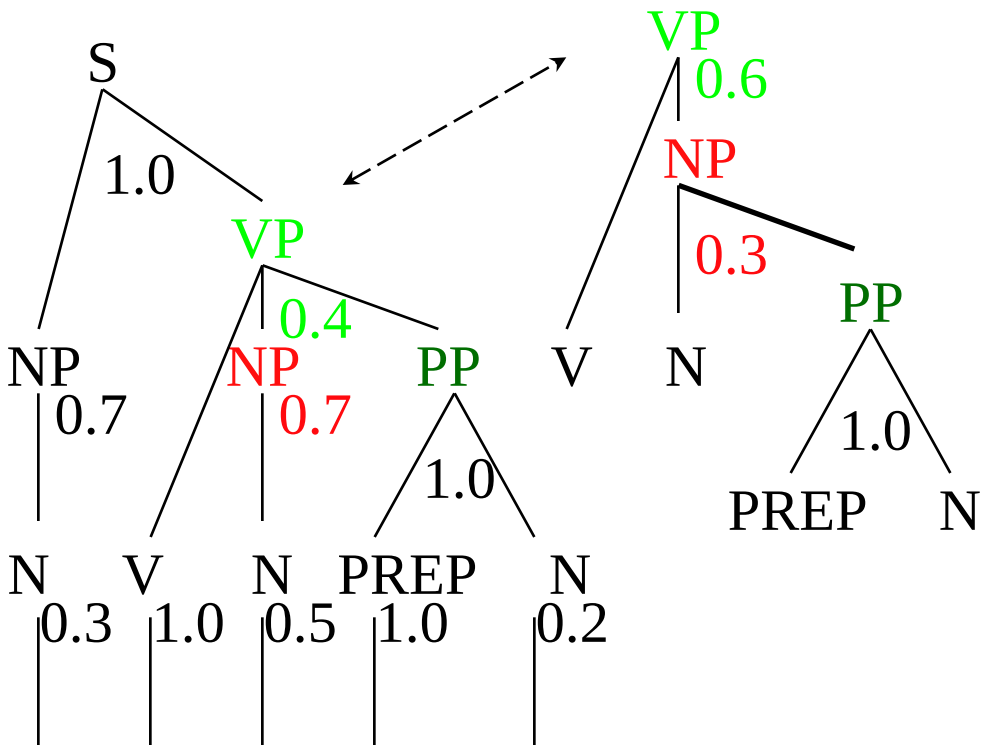
Ví dụ

- S → NP VP 0.80
- NP → Det N 0.30
- VP → V NP 0.20
- V → includes 0.05
- Det → the 0.50
- Det → a 0.40
- N → meal 0.01
- N → flight 0.02

Dùng thuật toán CYK phân tích câu vào:
"The flight includes a meal"

Tính Pr

1. S → NP VP 1.0
2. VP → V NP PP 0.4
3. VP → V NP 0.6
4. NP → N 0.7
5. NP → N PP 0.3
6. PP → PREP N 1.0
7. N → a_dog 0.3
8. N → a_cat 0.5
9. N → a_telescop 0.2
10. V → saw 1.0
11. PREP → with 1.0



a_dog saw a_cat with a_telescope

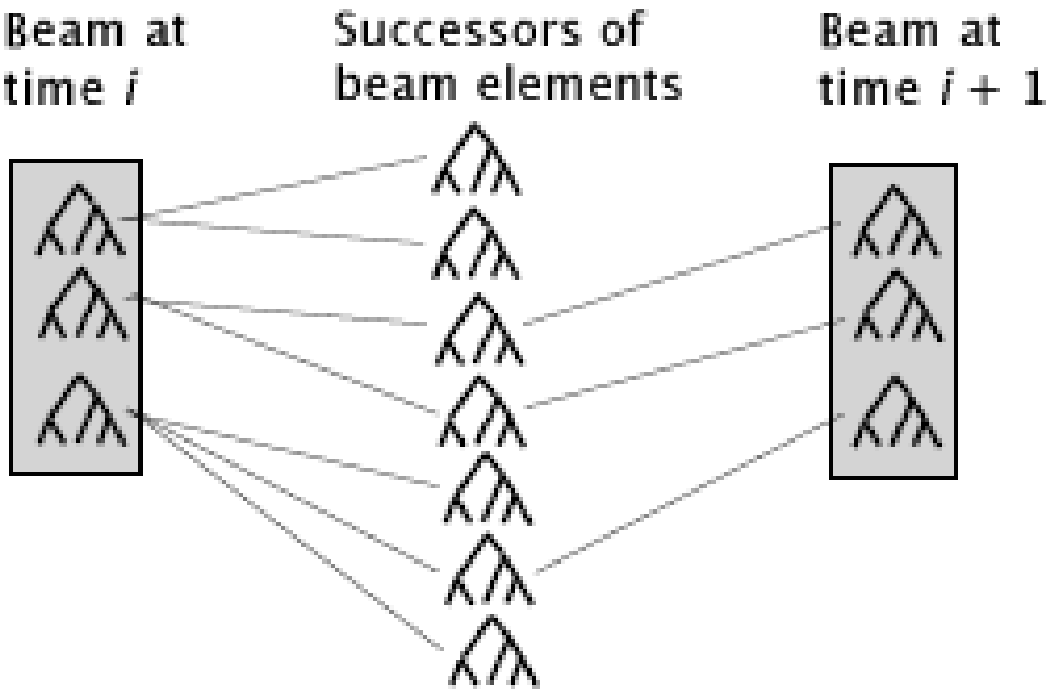
$P_l = 1'.7'.4'.3'.7'1'.5'1'1'.2 = .00588$

$P_r = 1'.7'.6'.3'.3'1'.5'1'1'.2 = .00378$

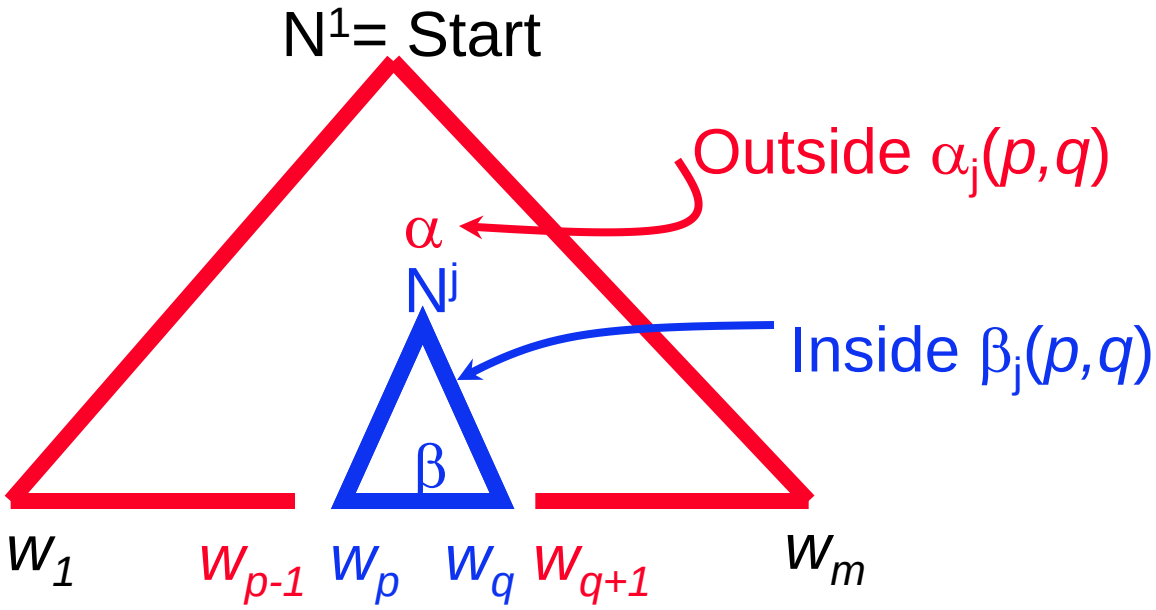
➢ P_l is chosen

Tìm kiếm kiểu chùm

- Tìm kiếm trong không gian trạng thái
- Mỗi trạng thái là một cây cú pháp con với 1 xác suất nhất định
  - Tại mỗi thời điểm, chỉ giữ các thành phần có điểm cao nhất

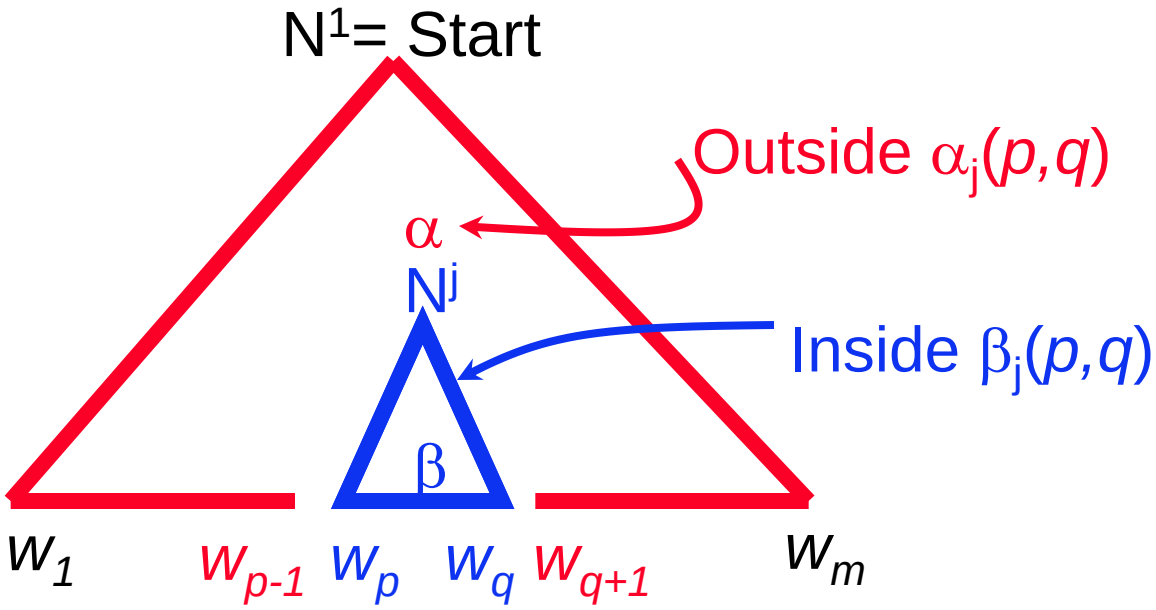


Xác suất trong và ngoài



- N_{pq} = ký hiệu không kết thúc N^j trải từ vị trí p đến q trong xâu
- α_j = xác suất ngoài (outside)
- β_j = xác suất trong (inside)
- N^j *phủ* các từ $w_p \ldots w_q$, *nếu* $N^j \Rightarrow^* w_p \ldots w_q$

Xác suất trong và ngoài



$\alpha_j(p,q)=P(w_{1(p-1)}\ ,\ N_{pq}^{j},w_{(q+1)m}|G)$

$\beta_j(p,q)=P(w_{pq}|N_{pq}^{j},\ G)$

$\alpha_j(p,q)\ \beta_j(p,q) = P(N^1 \Rightarrow^* w_{1m}\ ,\ N^j \Rightarrow^* w_{pq}\ |\ G)$

$= P(N^1 \Rightarrow^* w_{1m}\ |G) \bullet P(N^j \Rightarrow^* w_{pq}\ |\ N^1 \Rightarrow^* w_{1m,}\ G)$

Tính xác suất của xâu

- Sử dụng thuật toán ***Inside***, 1 thuật toán lập trình động dựa trên xác suất inside

$P(w_{1m}|G) = P(N^1 \Rightarrow^* w_{1m}|G) = P(w_{1m}|N_{1m}^1, G) = \beta_1(1,m)$

- Trường hợp cơ bản:

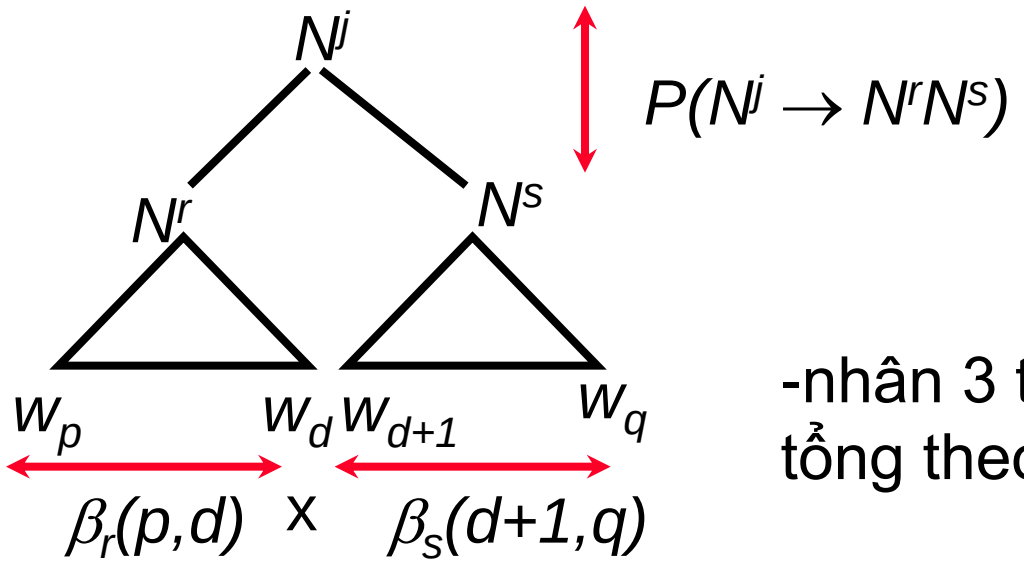
$$\beta_j(k,k) = P(w_k|N_{kk}^j, G) = P(N^j \rightarrow w_k|G)$$

- Suy diễn:

$$\beta_j(p,q) = \Sigma_{r,s}\Sigma_{d\in(p,q-1)} P(N^j \rightarrow N^r N^s)\ \beta_r(p,d)\ \beta_s(d+1,q)$$

Suy diễn

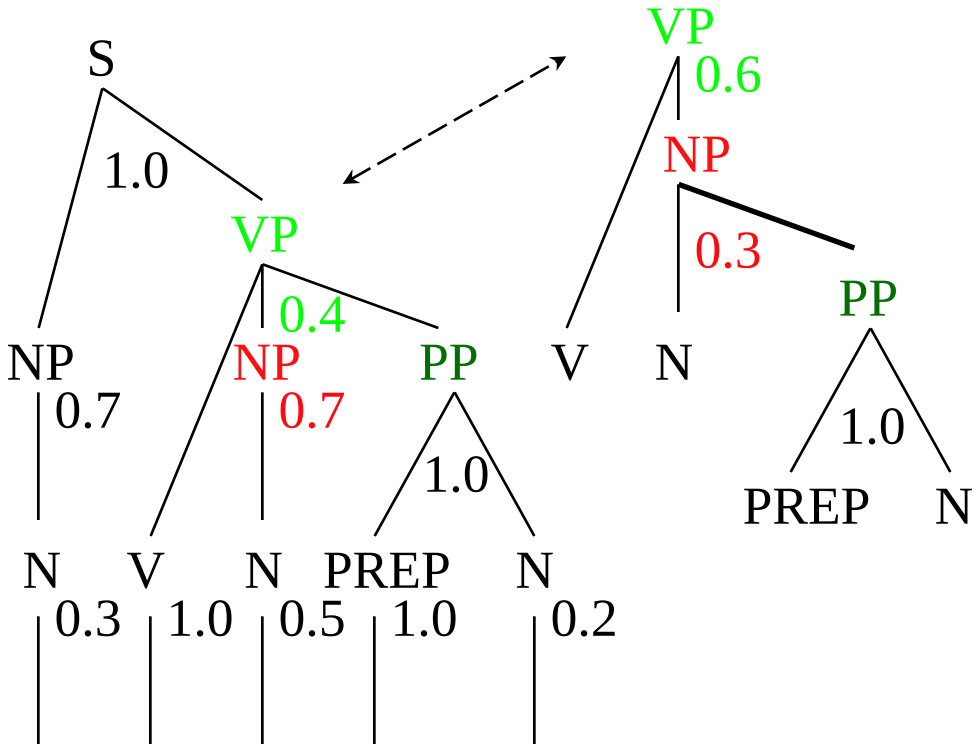
Tính $\beta_j(p,q)$ với p < q – tính trên tất cả các điểm j – thực hiện từ dưới lên



-nhân 3 thành phần, tính tổng theo *j, r,s.*

Ví dụ

1.	S → NP VP	1.0
2.	VP → V NP PP	0.4
3.	VP → V NP	0.6
4.	NP → N	0.7
5.	NP → N PP	0.3
6.	PP → PREP N	1.0
7.	N → a_dog	0.3
8.	N → a_cat	0.5
9.	N → a_telescope	0.2
10.	V → saw	1.0
11.	PREP → with	1.0



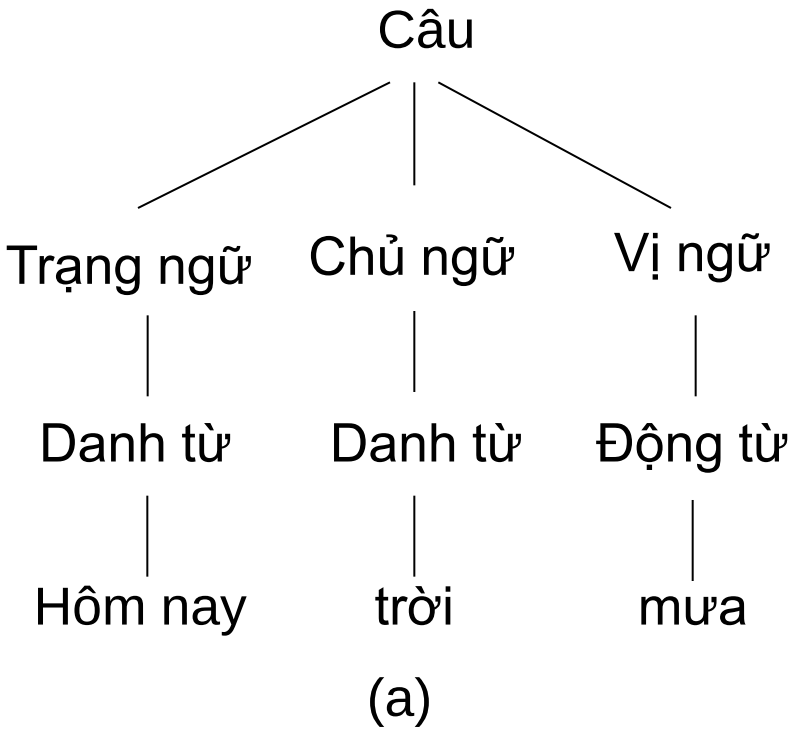
P(a_dog saw a_cat with a_telescope) =

1´.7´.4´.3´.7´1´.5´1´1´.2 + ... ´.6... ´.3... = .00588 + .00378 = .00966

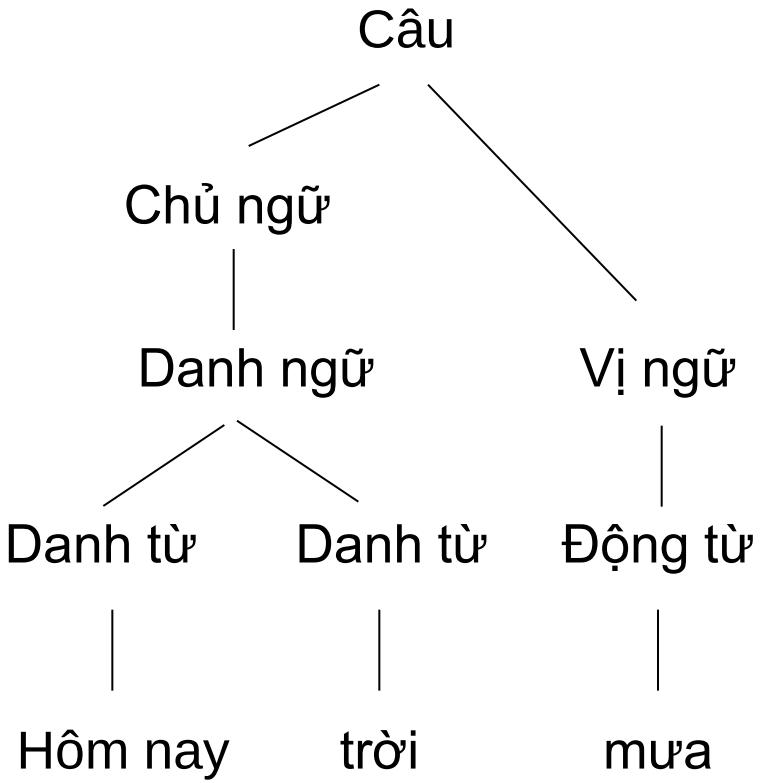
Nhập nhằng trong PTCP tiếng Việt

- 2 loại nhập nhằng cú pháp:
  - Câu có thể hiểu theo nhiều nghĩa khác nhau dẫn đến các cây cú pháp khác nhau.
    - Ví dụ, câu "*Tôi nhìn thấy anh Hải ở tầng hai*"
  - Câu chỉ có một nghĩa nhưng bộ PTCP vẫn tạo ra nhiều cây cú pháp, trong đó chỉ có một cây đúng.
    - Ví dụ, câu "*Hôm nay trời mưa*"

Nhập nhằng trong PTCP tiếng Việt



(a)



(b)

Nhập nhằng trong PTCP tiếng Việt

Hướng giải quyết:

Cách 1: Phân loại chi tiết hơn các nhãn từ loại/ngữ loại:

Thay vì luật

<Danh ngữ> → <Danh từ><Danh từ>

ta đưa ra luật

<Danh ngữ> → <Danh từ loại A><Danh từ loại B>.

Nhược điểm:

- Chưa thống nhất trong việc đặt tên các nhãn từ loại/ngữ loại
- Kích thước tập luật cú pháp tăng lên đáng kể.
- Phải xây dựng một cách thủ công tập luật cú pháp ứng với tập nhãn từ loại mới → khó thực hiện

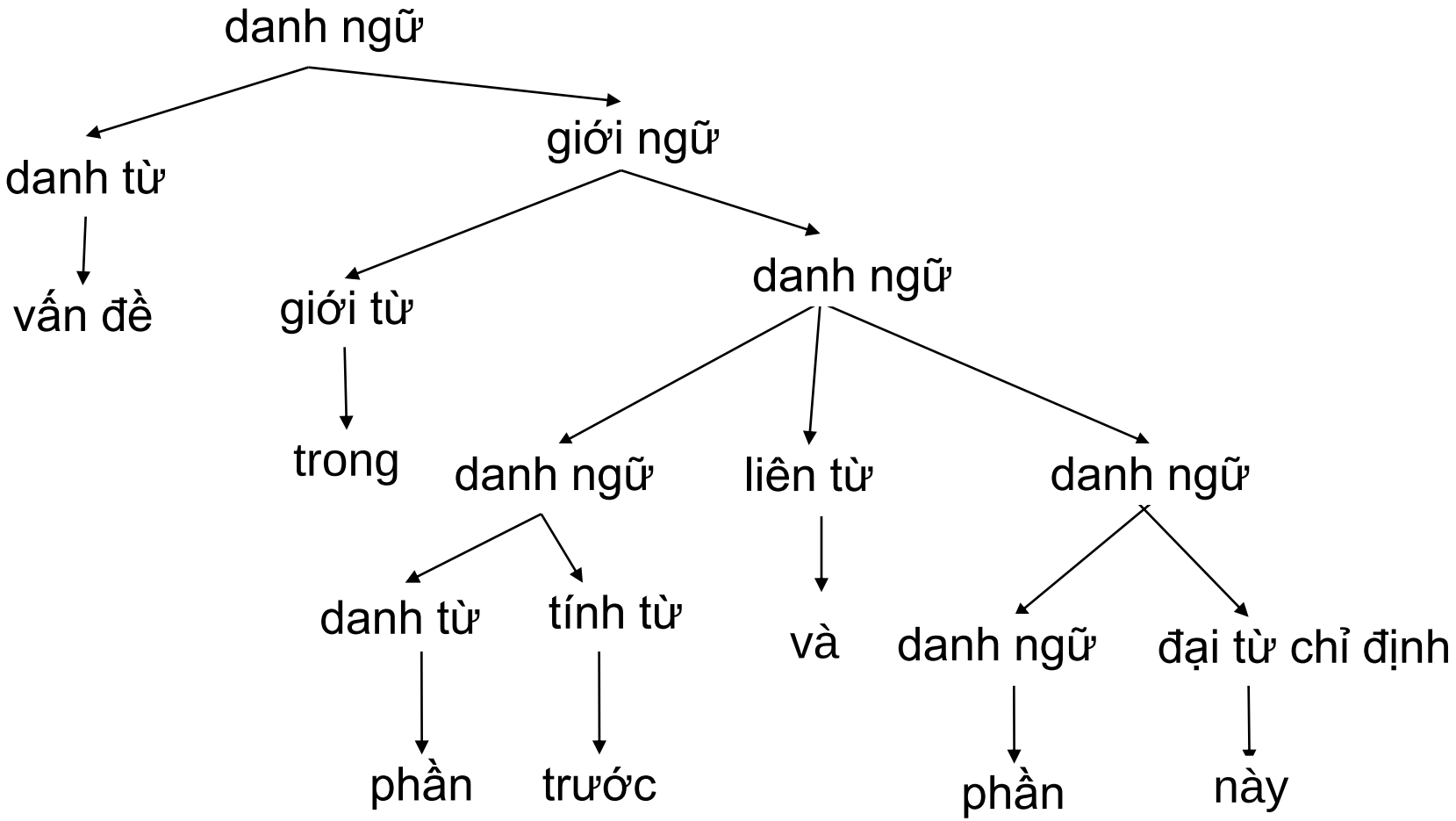
Nhập nhằng trong PTCP tiếng Việt

Hướng giải quyết:

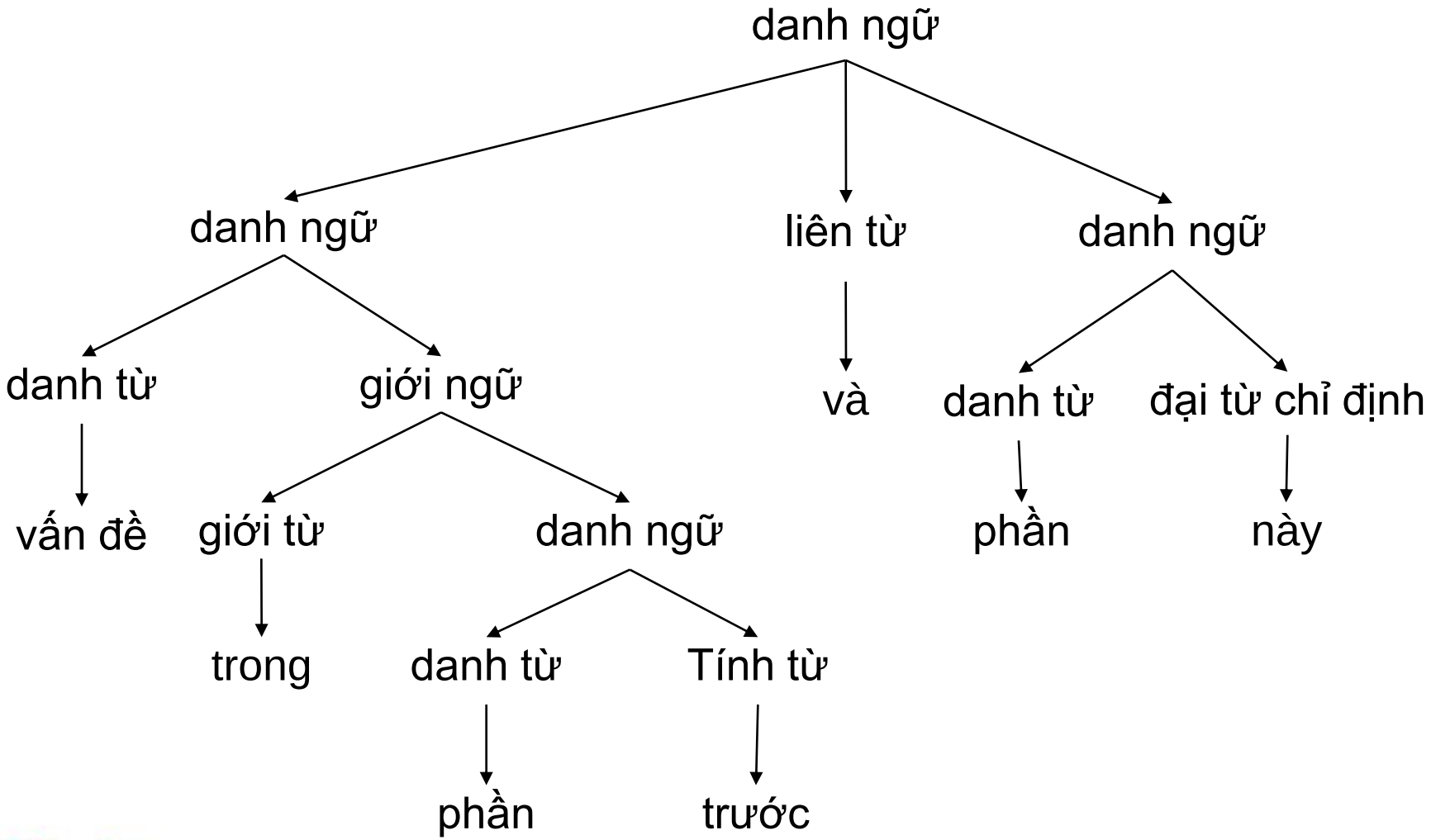
Cách 2: đưa xác suất vào tập luật cú pháp CFG

- Xử lý được câu "*Tôi nhìn thấy anh Hải ở tầng hai*"
- Chưa giải quyết nhập nhằng liên quan đến tính chất của các từ cụ thể.
- Ví dụ, danh ngữ "*vấn đề trong phần trước và phần này*"

Nhập nhằng trong PTCP tiếng Việt



Nhập nhằng trong PTCP tiếng Việt



Các từ cụ thể đôi khi ảnh hưởng đến việc PTCP

1. Để giải quyết nhập nhằng trong PTCP, đôi khi cần thông tin về từ cụ thể. Ví dụ
   - "*Tôi ăn*" ít khi được chấp nhận là một câu hoàn chỉnh do mang lượng thông tin nhỏ.
   - "*Tôi đang ăn*" dễ được chấp nhận là câu hoàn chỉnh hơn.

➢ Phải dựa trên tính chất cụ thể của từ giữ vai trò chính trong câu

2. Nhập nhằng do lược bỏ quan hệ từ. Ví dụ
   - có thể nói *bạn tôi, con tôi;*
   - không nói *con chó tôi, con mèo tôi*.

➢ Từ cũng có vai trò quan trọng trong việc PTCP

➢ đưa thông tin từ vựng vào văn phạm (làm giàu PCFG)

Làm giàu PCFG

- PCFG từ vựng hóa : PLCFG (Probabilistic Lexicalized CFG, Collins 1997; Charniak 1997)
- Gán từ vựng với các nút của luật
- Cấu trúc Head
  - Mỗi phần tử của parsed tree được gắn liền với một *lexical he*ad
  - Để xác định *head* của một nút trong ta phải xác định trong các nút con, nút nào là *head* (xác định *head* trong vế phải của một luật).

Làm giàu PLCFG

VP(dumped) → VBD(dumped) NP(sacks) PP(into) $3*10^{-10}$

VP(dumped) → VBD(dumped) NP(cats) PP(into) $8*10^{-11}$

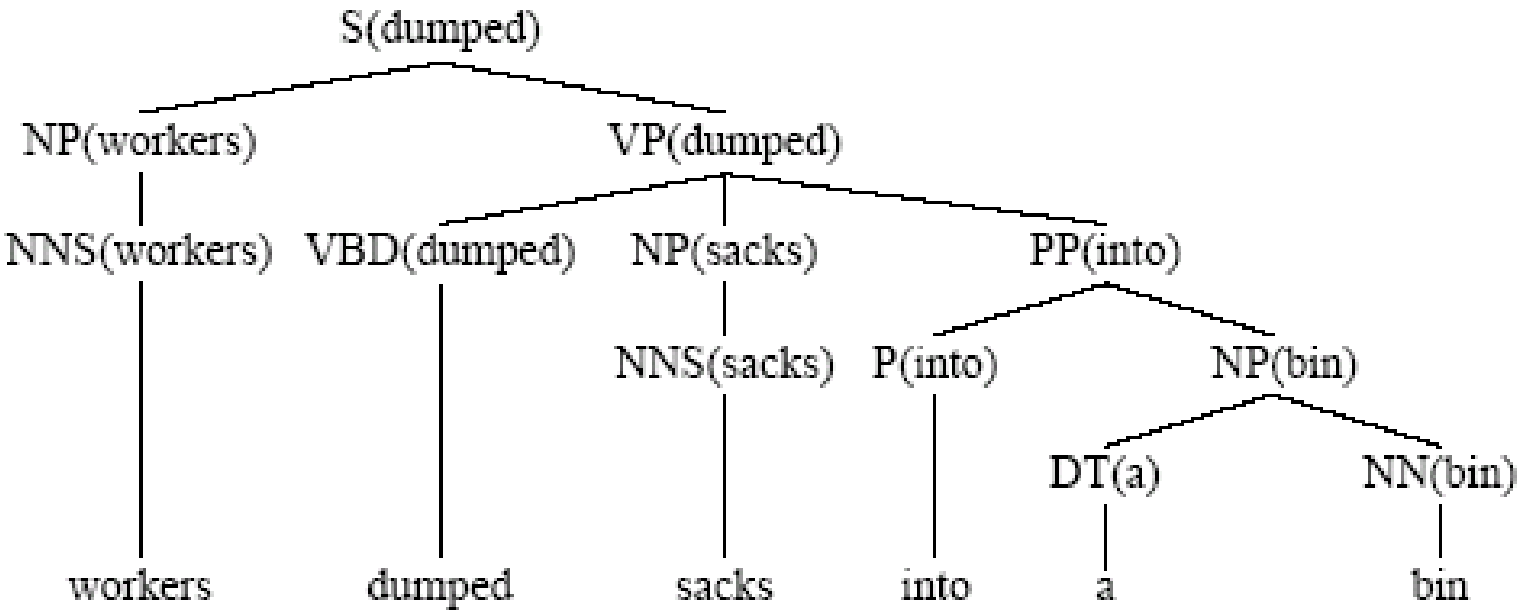


Figure 12.12 A lexicalized tree from Collins (1999).

Hạn chế của PLCFG

VP -> VBD NP PP

VP(*dumped*) -> VBD(*dumped*) NP(*sacks*) PP(*into*)

- Không có một corpus đủ lớn!
  - Thể hiện hết các trường hợp cú pháp, hết các trường hợp đối với từng từ.

Penn Treebank

- Penn Treebank: tập ngữ liệu có chú giải ngữ pháp, có 1 triệu từ, là nguồn ngữ liệu quan trọng
- Tính thưa:
  - có 965,000 mẫu, nhưng chỉ có 66 mẫu WHADJP, trong đó chỉ có 6 mẫu không là *how much* hoặc *how many*
- Phần lớn các phép xử lý thông minh phụ thuộc vào các thống kê mối quan hệ từ vựng giữa 2 từ liền nhau:

A Penn Treebank tree

```
( (S (NP-SBJ The move)
     (VP followed
         (NP (NP a round)
             (PP of
                 (NP (NP similar increases)
                     (PP by
                         (NP other lenders))
                     (PP against
                         (NP Arizona real estate loans)))))
         ,
         (S-ADV (NP-SBJ *)
                (VP reflecting
                    (NP (NP a continuing decline)
                        (PP-LOC in
                                (NP that market))))))
     .))
```

Đánh giá độ chính xác của PTCP

- Độ chính xác của parser được đo qua việc tính xem có bao nhiêu thành phần ngữ pháp trong cây giống với cây chuẩn, gọi là gold-standard reference parses.
- Độ chính xác (Precision) =

$$\% \quad \frac{\text{trường hợp hệ gán đúng}}{\text{tổng số trường hợp hệ gán}}$$

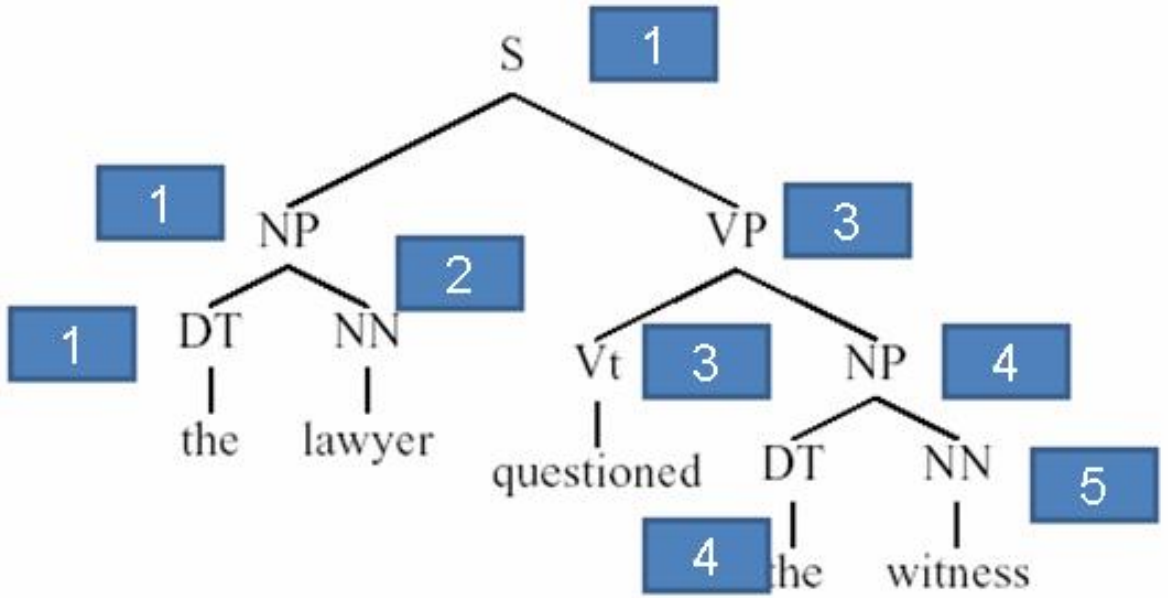
(%THợp hệ tính đúng).

- Độ phủ (Recall) =

$$\% \text{ số } \frac{\text{trường hợp hệ gán đúng}}{\text{tổng số trường hợp đúng}}$$

(%THợp hệ tính đúng so với con người).

Biểu diễn cây theo các thành phần ngữ pháp



Label	Start Point	End Point
NP	1	2
NP	4	5
VP	3	5
S	1	5

Đánh giá

Precision and Recall

Label	Start Point	End Point
NP	1	2
NP	4	5
NP	4	8
PP	6	8
NP	7	8
VP	3	8
S	1	8

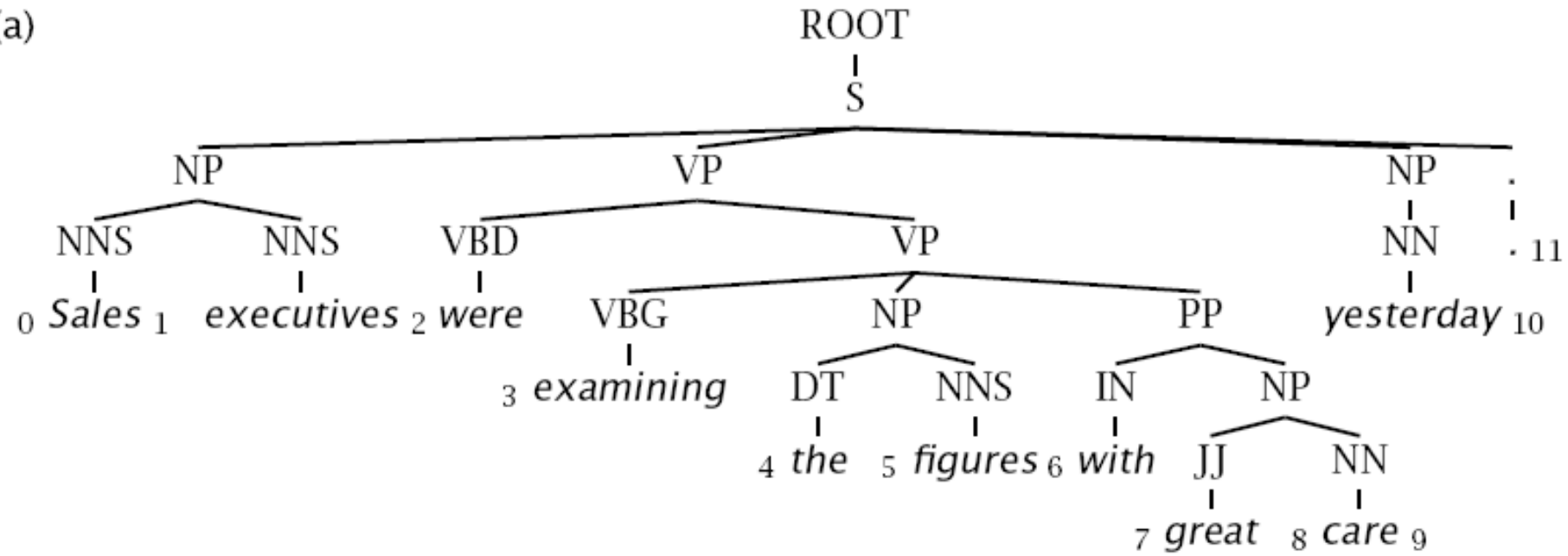
Label	Start Point	End Point
NP	1	2
NP	4	5
PP	6	8
NP	7	8
VP	3	8
S	1	8

- G = number of constituents in gold standard = 7
- P = number in parse output = 6
- C = number correct = 6

$$\text{Recall} = 100\% \times \frac{C}{G} = 100\% \times \frac{6}{7} \qquad \text{Precision} = 100\% \times \frac{C}{P} = 100\% \times \frac{6}{6}$$

Ví dụ 2

(a)



(b) Brackets in gold standard tree (a.):
S-(0:11), **NP-(0:2)**, VP-(2:9), VP-(3:9), **NP-(4:6)**, PP-(6-9), NP-(7,9), *NP-(9:10)

(c) Brackets in candidate parse:
S-(0:11), **NP-(0:2)**, VP-(2:10), VP-(3:10), NP-(4:10), **NP-(4:6)**, PP-(6-10), NP-(7,10)

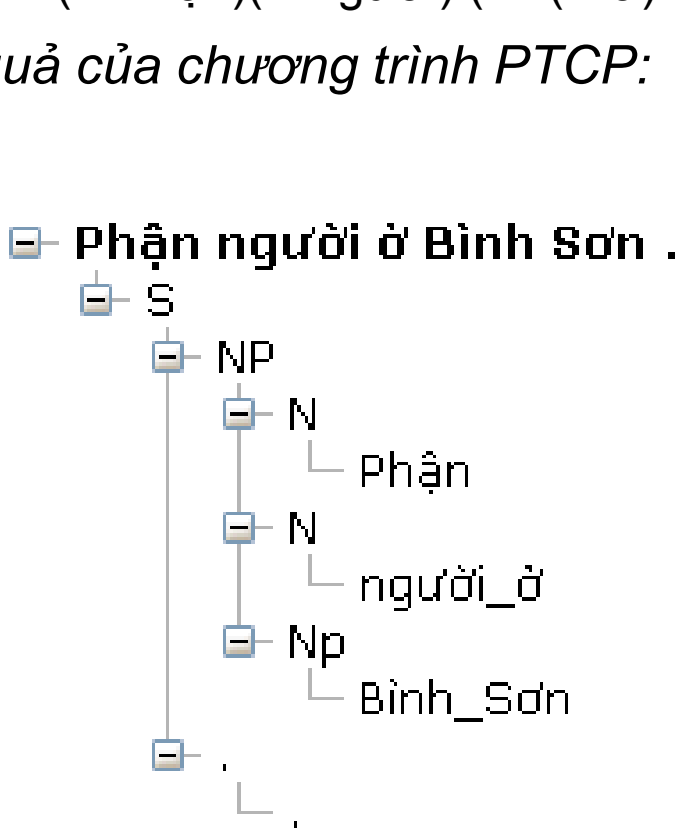
(d) Precision:	3/8 = 37.5%	Crossing Brackets:	0
Recall:	3/8 = 37.5%	Crossing Accuracy:	100%
Labeled Precision:	3/8 = 37.5%	Tagging Accuracy:	10/11 = 90.9%
Labeled Recall:	3/8 = 37.5%		

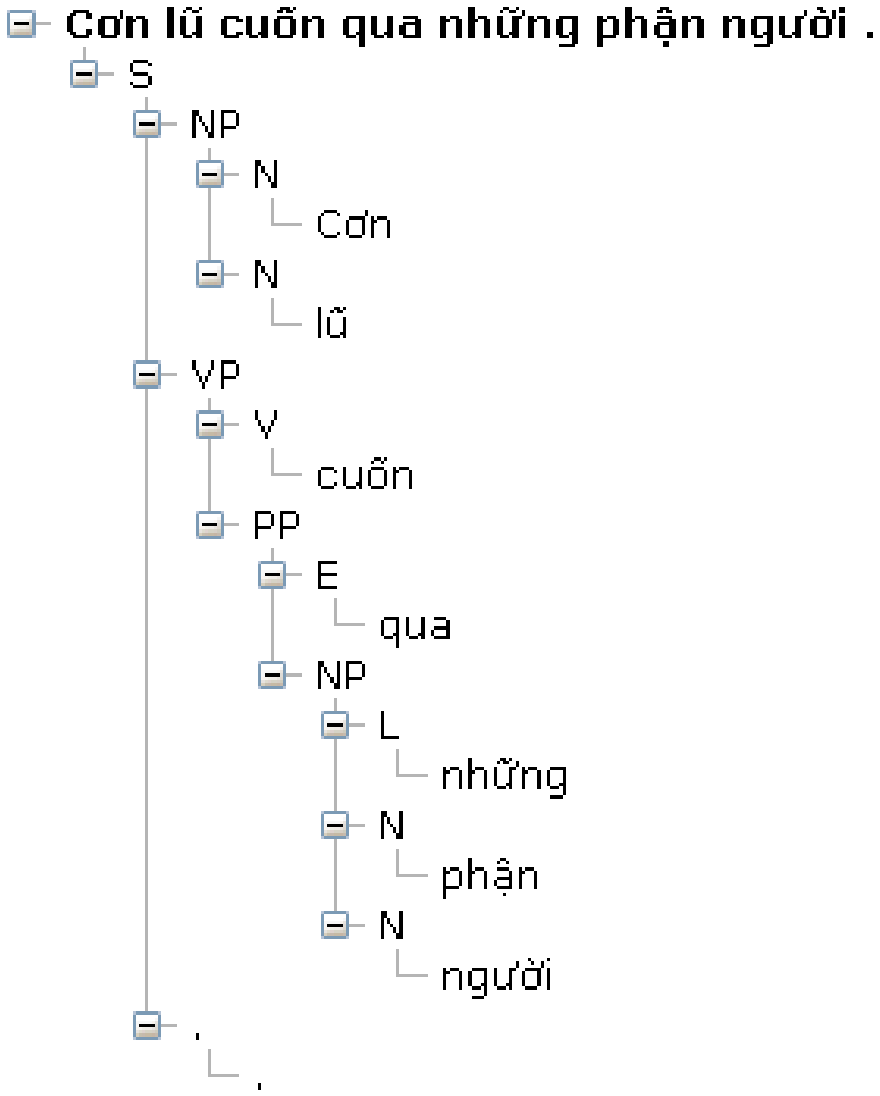
Bài tập - tính P, R

Cho kết quả PTCP chuẩn:

- (S (NP (N Cơn)(N lũ)) (VP(V cuốn)(V qua) (NP (L những)(N phận)(N người))) (. .))
- (S(NP(N Phận)(N người) (PP(E ở) (NP(Np Bình Sơn))))( ))

Kết quả của chương trình PTCP:





Các hệ thống PTCP tốt nhất

- CFG (context free grammar):
  - Berkeley : http://nlp.cs.berkeley.edu/software.shtml
  - Charniak: http://bllip.cs.brown.edu/resources.shtml
- HPSG (Head-driven Phrase Structure Grammar)
  - Enju, deepNLP: https://mynlp.github.io/enju/
- Depedency grammar
  - ClearNLP : http://clearnlp.wikispaces.com/depParser
  - Google SyntaxNet: open-source, sử dụng NN, cho câu đúng ngữ pháp,
  - https://research.googleblog.com/2016/05/announcing-syntaxnet-worlds-most.html
  - Netbase, cho cả câu twitter
  - https://www.codeproject.com/Articles/43372/NetBase-A-Minimal-NET-Database-with-a-Small-SQL
  - Stanford **:** https://nlp.stanford.edu/software/lex-parser.shtml