ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA KHOA HỌC MÁY TÍNH

TRỊNH QUANG TRƯỜNG	**18520393**
PHẠM XUÂN THIÊN	**18520158**
TĂNG NĂNG CHUNG	**18520536**

ĐỒ ÁN MÔN HỌC

TÌM HIỂU VÀ PHÂN TÍCH CÚ PHÁP CẤU TRÚC NGỮ ĐOẠN

MÔN HỌC: XỬ LÝ NGÔN NGỮ TỰ NHIÊN

GIẢNG VIÊN HƯỚNG DẪN

NGUYỄN TRỌNG CHỈNH

TP. HỒ CHÍ MINH, 2021

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG

DANH MỤC TỪ VIẾT TẮT

CKY, CYK : Cocke–Younger–Kasami algorithm

CFG: Context-free grammar

CNF: Chomsky Normal Form

TÓM TẮT KHÓA LUẬN

Đề tài: Tìm hiểu và phân tích cú pháp, cấu trúc ngữ đoạn

Sinh viên thực hiện: Trịnh Quang Trường - 18520393, Phạm Xuân Thiên - 18520158, Tăng Năng Chung - 18520536

Giảng viên hướng dẫn: Nguyễn Trọng Chỉnh

Khoa: Khoa học máy tính

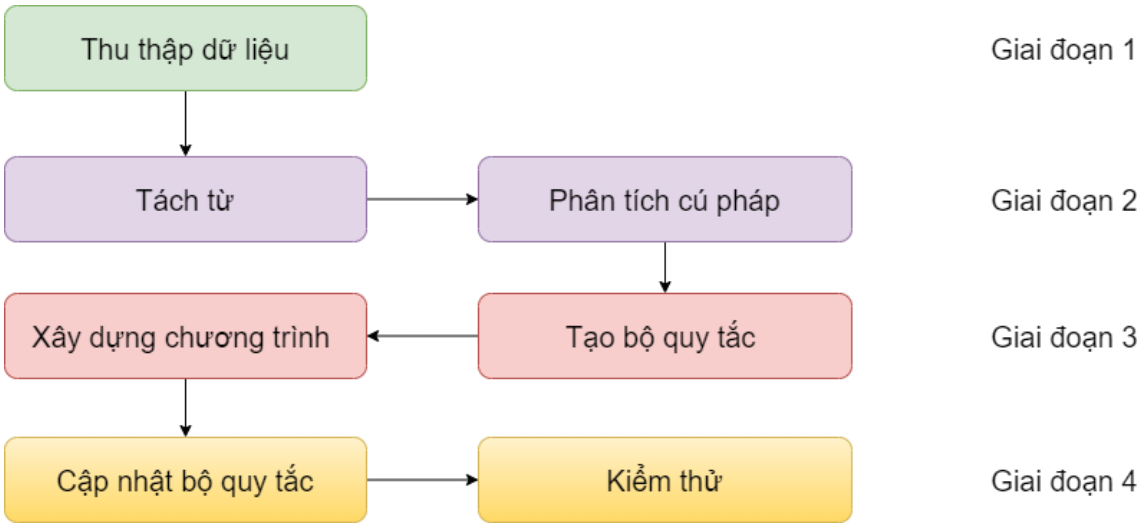
Tóm tắt: Như chúng ta đã biết, máy tính là công cụ do con người đã tạo ra cách đây khá lâu đời. Tuy là công cụ do con người tạo ra nhưng máy tính vẫn chưa thể hiểu được ngôn ngữ tự nhiên của con người. Một trong những hướng để phát triển và giải quyết bài toán làm sao cho máy tính có thể hiểu được ngôn ngữ của con người đó là xử lý ngôn ngữ tự nhiên. Chính vì lý do trên nên nhóm đã chọn đồ án tìm hiểu và phân tích cú pháp cấu trúc ngữ đoạn bởi nó hết sức quan trọng trong xử lý ngôn ngữ tự nhiên. Quá trình thực hiện đồ án, nhóm thao tác trên bộ dữ liệu 78 câu được thu thập từ các tiêu đề của bài báo trên trang báo tuổi trẻ. Sử dụng thuật toán Maximum Matching để tiến hành tách từ, tìm hiểu bộ nhãn VLSP 7.3 với 24 nhãn sẵn và gán nhãn thủ công. Sau đó sử dụng thuật toán đã học là CKY để tiến hành tạo bộ quy tắc. Cuối cùng là cập nhập lại bộ quy tắc và kiểm thử, kết quả trung bình thu được sau quá trình kiểm thử với thuật toán CKY là 0.58(precision) và 0.57(recall). Kết quả sau khi thực hiện là khá tốt vì hạn chế về bộ dữ liệu nên kết quả chỉ đạt được ở mức tầm trung chưa thực sự cao lắm những nhìn chung ở mức khá. Sau quá trình thực hiện đề tài tuy kết quả chưa thực sự được cao nhưng đồ án cũng đã giúp các thành viên rút ra được một số kinh nghiệm riêng cũng như tạo một ít tiền đề để trong tương lai có thể phát triển thêm tiến gần hơn với công nghệ làm máy tính có thể hiểu được ngôn ngữ tự nhiên.

MỞ ĐẦU

Con người đã hình thành và phát triển từ lâu đời, qua quá trình phát triển đó mỗi dân tộc đều hình thành những văn hóa và ngôn ngữ riêng cho mình. Vì cái riêng đó nên chúng ta cũng không thể hiểu hết được các ngôn ngữ của nhau, cũng như thế máy tính được ra đời cách đây khá lâu đời, nhưng từ khi máy tính tồn tại cho đến nay nó vẫn hoàn toàn không thể hiểu được ngôn ngữ tự nhiên của con người, các lập trình viên đã cố gắng viết ra các chương trình để máy tính có thể hiểu được ngôn ngữ tiếng Anh. Tuy nhiên máy tính vẫn không thể nào hiểu được ngôn ngữ tự nhiên và sẽ thực sự hữu ích nếu một máy tính có thể đọc và hiểu được ngôn ngữ tự nhiên của con người. Một giải pháp đề ra giúp máy tính có thể hiểu được ngôn ngữ đó là xử lý ngôn ngữ tự nhiên. Xử lý ngôn ngữ tự nhiên là một nhánh của trí tuệ nhân tạo tập trung vào các ứng dụng trên ngôn ngữ của con người. Trong xử lý ngôn ngữ tự nhiên thì việc tìm hiểu và phân tích cú pháp cấu trúc ngữ đoạn là một bài toán hết sức quan trọng, nó nhằm mục đích giúp máy tính có thể nhận biết cú pháp và cấu trúc của câu. Việc phân tích cú pháp câu có thể chia làm hai mức chính. Mức thứ nhất là tách từ và xác định thông tin từ loại. Mức thứ hai là sinh cấu trúc cú pháp cho câu dựa trên các từ và từ loại do bước trước cung cấp. Với một công cụ phân tích cú pháp tốt, chúng ta có thể tích hợp vào nhiều ứng dụng trong xử lý ngôn ngữ tự nhiên như dịch máy, tóm tắt văn bản, các hệ thống hỏi đáp, … để tăng tính chính xác của các ứng dụng đó. Hiện nay, các công cụ phân tích cú pháp tiếng Việt đã đạt được một số kết quả nhất định. Tuy nhiên, phần lớn các kết quả đạt được mới dừng ở một số trường hợp câu cơ bản như câu đơn và các câu ghép đơn giản. Hiện tượng nhập nhằng và những trường hợp đặc biệt trong phân tích câu vẫn chưa được giải quyết thoả đáng.

Chương 1. QUY TRÌNH THỰC HIỆN

1.1. Tóm tắt quy trình



Ảnh 1.1 Sơ đồ quá trình thực hiện đồ án

1.2. Nội dung:

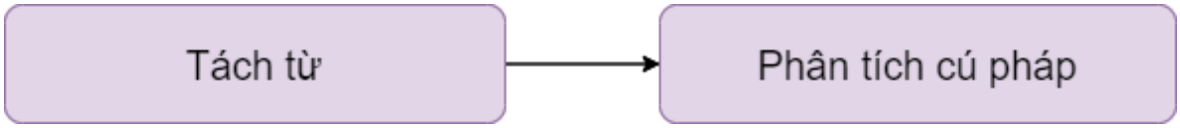
Giai đoạn 1: Thực hiện tìm hiểu bài toán và thu thập dữ liệu.

Giai đoạn 2: Tiến hành tách từ, sau đó gán nhãn từ loại thu được bộ ngữ liệu.

Giai đoạn 3: Tạo tập quy tắc từ bộ ngữ liệu, xây dựng chương trình phân tích cú pháp tự động.

Giai đoạn 4: Đánh giá kết quả đạt được từ bộ quy tắc, cập nhật và kiểm thử lần cuối.

Chương 2. **TẠO BỘ NGỮ LIỆU**



Ảnh 2.1 Giai đoạn 2

2.1. Thu thập ngữ liệu

Vì giới hạn nội dung môn học, nên nhóm chỉ tiến hành thu thập 78 câu là tiêu đề của các bài báo trên báo Tuổi trẻ (*Tuoitre.vn*).

Ví dụ:

Một xã đảo gần 90 công trình xây dựng trái phép , sẽ cưỡng chế 74 công trình .
Anh bị sóng cuốn , em nhảy xuống cứu , chết đuối cả hai .
Bộ GTVT kiến nghị Thủ tướng không triển khai thu phí điện tử tại 7 trạm BOT .
Bắc Giang có tân chủ tịch HĐND và UBND tỉnh .
Nhiều tuyến đường Hà Nội lại hàng núi rác .
Hai tân phó chủ tịch UBND TP. HCM cam kết gì khi được bầu .

Nhận xét bộ dữ liệu sau khi thu thập:

Thu thập được nhiều dạng câu: Câu đầy đủ chủ ngữ-vị ngữ, câu khuyết chủ ngữ, câu cảm thán.

Các cấu lấy về có một số câu khá phức tạp: Câu ghép

2.2. Tách từ

2.2.1. Lý thuyết tách từ

Thuật ngữ “Tách từ” trong tiếng anh là “word segmentation”.

Là liên kết các từ đơn thành một cụm từ có ý nghĩa khác với nghĩa của từng từ đơn. Về mặt biểu hiện thì nhóm em sử dụng dấu ‘_’ để biểu thị liên kết cho 1 từ. Chẳng

hạn như: “Lớp ta phải đoàn kết” sau khi liên kết thì sẽ thành “Lớp ta phải đoàn_kết”.

Lý do phải tách từ:

Vì cấu trúc các từ Tiếng Việt khác với nhiều ngôn ngữ khác. Việc một từ được tạo thành từ 2 hay nhiều từ khác để biểu hiện một ngữ nghĩa nào đó là việc rất bình thường.

Việc tách từ này còn hỗ trợ cho việc xử lý các bài toán liên quan đến ngữ nghĩa.

Không giống như tiếng Anh, tiếng Việt không biến đổi hình thái và không xác định ranh giới từ bằng khoảng trống nên một câu có thể mang nhiều ngữ nghĩa khác nhau dựa trên các cách tách từ.

Ví dụ:

Với câu *“Con ngựa đá con ngựa đá”*

Con_ngựa/ đá / con_ngựa_đá

Con_ngựa đá / con_ngựa_đá

Với câu *“Học sinh học sinh học”* thì nên tách là

Học sinh / học / sinh học

Kết luận: Cho nên việc tách từ trong xử lý ngôn ngữ tự nhiên tiếng Việt đóng một vai trò vô cùng quan trọng và là tiền đề để xử lý các bài toán lớn hơn.

2.2.2. Thuật toán sử dụng

Các phương pháp tách từ phổ biến: Longest Matching, Maximum Matching, Hidden Markov Models – HMM, Transformation–based Learning – TBL.

Nhóm em đã sử dụng phương pháp: Maximum Matching

Ý tưởng của maximum matching:

Sử dụng một bộ từ điển chứa tất cả các từ đang xét và sẽ duyết từ trái qua phải (hoặc ngược lại) và chọn từ dài nhất có thể chọn được (từ xuất hiện trong từ điển) nếu không thì sẽ giảm độ dài của từ đó và tiếp tục kiểm tra cho đến khi hết câu.

Ưu điểm của Maximum matching: Đơn giản, dễ hiểu, tốc độ nhanh. Phù hợp với bộ dữ liệu nhỏ có bộ từ điển ít gặp nhập nhằng.

Nhược điểm: Không xử lý được các trường hợp từ chưa bao giờ xuất hiện và chắc chắn không thể tạch được nếu từ đó không ở trong từ điểm. Và không giải quyết được các trường hợp nhập nhằng.

Với câu *"Công an toàn ở nhà"* thì nên tách là *"Công an_toàn ở nhà"* nhưng maximum matching (từ trái qua phải) sẽ tách là *"Công_an toàn ở nhà"*

2.2.3. Thực hiện đánh giá độ chính xác của thuật toán trên bộ dữ liệu

Trong đồ án này thì chúng em thực hiện tạch thủ công để tạo bộ dữ liệu tách từ gold (dùng để đánh giá kết quả tách từ)

Chúng em cùng nhau thực hiện việc tách từ và có sử dụng từ điển của "Đề tài VLSP" để kiểm tra xem cụm đó có được xem là một từ hay không. Trong quá trình thực hiện tách từ thì nhóm em đạt độ đồng thuận hoàn toàn.

Và nhóm có thực hiện so sách thuật toán Maximum matching với một công cụ tách từ tự động của VnCoreNLP được kết quả như bảng sau:

	Precision	**Recall**
Maximum Matching	99.583	98.974
VnCoreNLP	95.458	95.315

Bảng 2-1 Bảng so sánh Maximum matching và VnCoreNLP

Nhận xét:

- Độ chính xác của Maximum Matching tốt hơn VnCoreNLP với bộ dữ liệu này.
- Thuật toán Maximum Matching rất phù hợp với bộ dữ liệu mà nhóm em thu thập.
- Cho biết rằng các bộ dữ liệu có khá ít nhập nhằng

Ví dụ một số câu sau tách từ của cả 2 phương pháp trong bộ dữ liệu:

2.3. Phân tích cú pháp

2.3.1. Bộ nhãn sử dụng

Để thực hiện gắn nhẫn chúng em có thực hiện tìm hiểu bộ Guildeline VLSP 7.3

Tập các nhãn:

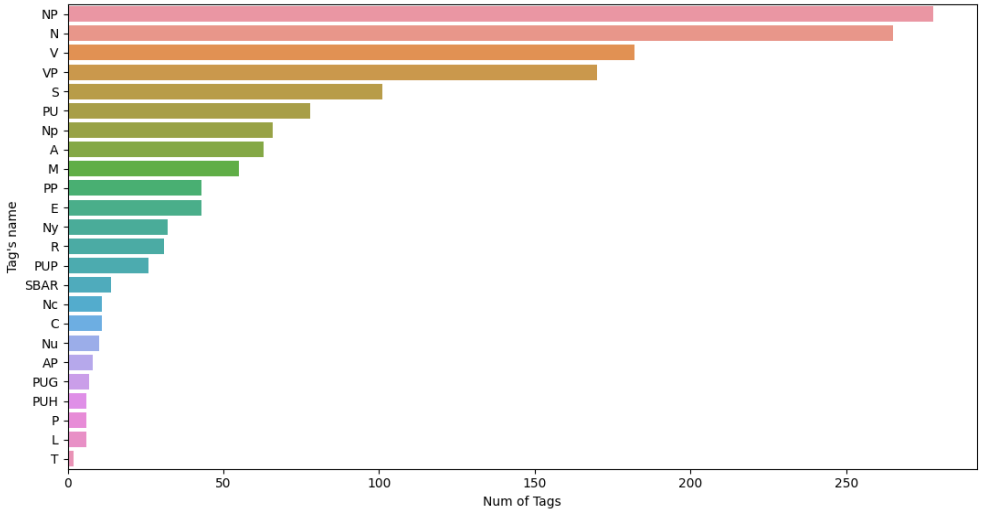
- Các nhãn mệnh đề (5 nhãn): S, SQ, S-EXC…
- Các nhãn từ loại (17 nhãn): N, NP, V, A…
- Các nhãn phân loại phụ ngữ của động từ (8 nhãn): TMP, LOC...
- Các nhãn cụm từ (13 nhãn): NP, VP, AP...
- Các nhãn chức năng cú pháp (9 nhãn): H, SUB, DOB...

Câu ví dụ:

Cô ấy hát không hay.

(S (NP-SUB (N-H cô) (P ấy))
    (VP (V-H hát)
        (AP (R không) (A-H hay))
    (PU .))

Vì giới hạn của đồ án, nhóm chỉ sử dụng 24 nhãn. Số nhãn được thống kê ở ảnh 2.2

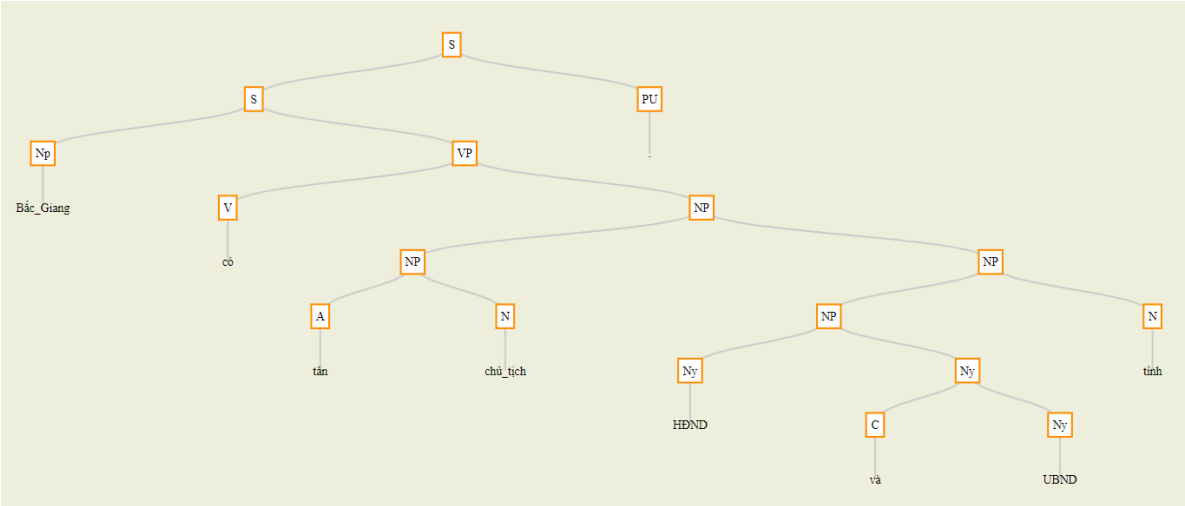


Ảnh 2.2: Bảng thống kê các nhãn sử dụng trong bộ dữ liệu

Tiến hành gán nhãn trên bộ dữ liệu sau khi thực hiện tách từ thủ công, thu được bộ dữ liệu gold.

Ví dụ: *"Bắc_Giang có tân chủ_tịch HĐND và UBND tỉnh ."*

(S (S (Np Bắc_Giang) (VP (V có) (NP (NP (A tân) (N chủ_tịch)) (NP (NP (Ny HĐND) (Ny (C và) (Ny UBND))) (N tỉnh))))) (PU .))



Ảnh 2.3. Ví dụ câu đã phân tích cú pháp trong bộ ngữ liệu

2.3.2. Xử lý nhập nhằng

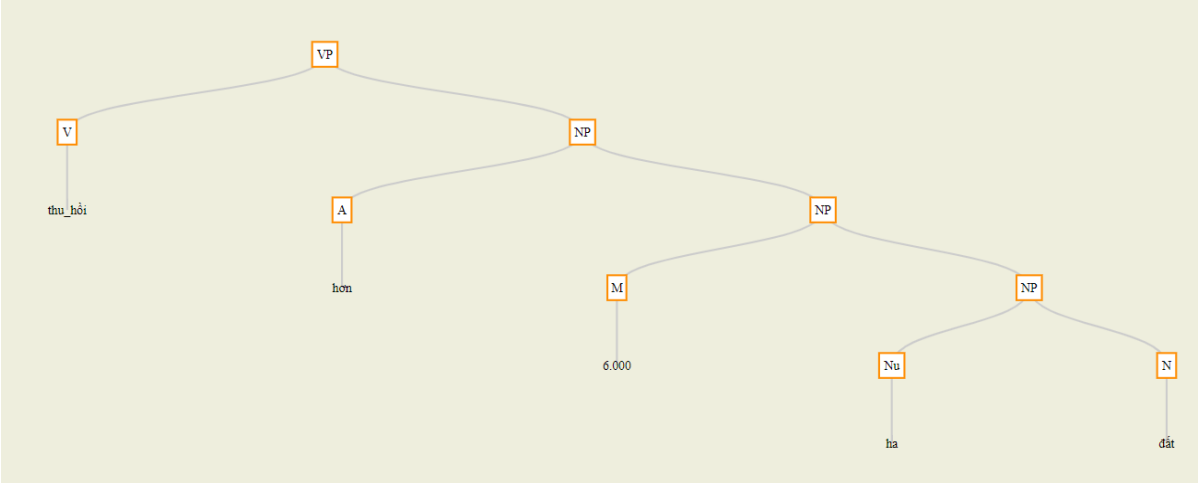
Vì bộ dữ liệu là tiếng Việt và việc gán nhãn được thực hiện thủ công, nên xảy ra một số nhập nhằng trong quá trình gán nhãn. Tiêu biểu trong đó là:

Nhập nhằng trong việc từ bổ nghĩa cho đông từ hay danh từ.

Ví dụ: Trong câu *"thu_hồi hơn 6000 ha đất"*

Có 2 cách gắn là:

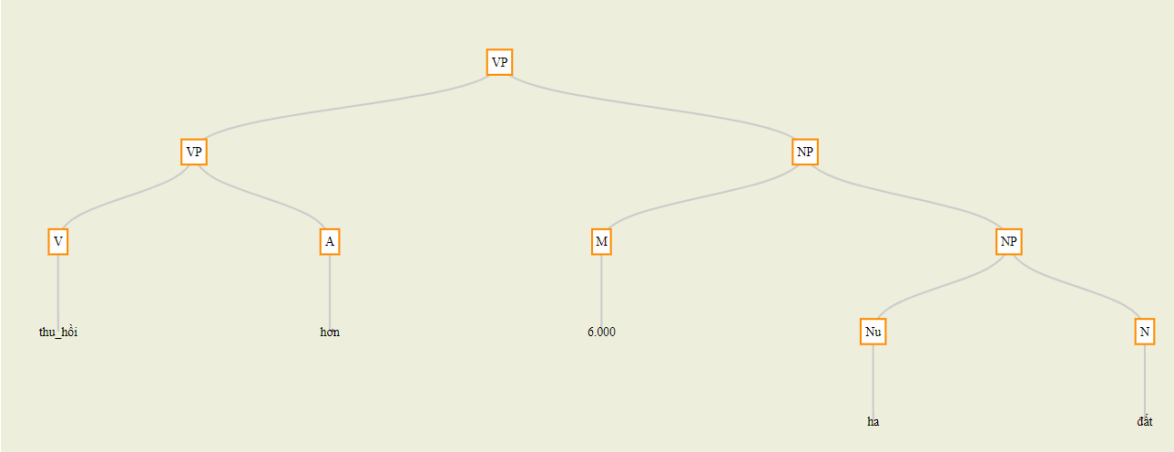
(VP (V thu_hồi) (NP (A hơn) (NP (M 6.000) (NP (Nu ha) (N đất)))))



Từ *'hơn'* bổ nghĩa cho cụm danh từ *"6000 ha đất"*.

hoặc

(VP (VP (V thu_hồi) (A hơn)) (NP (M 6.000) (NP (Nu ha) (N đất))))



Từ *'hơn'* bổ nghĩa cho động từ *'thu_hồi'*.

Nhóm đã thông nhất quyết định trong những trường hợp thế này sẽ gắn theo các thứ nhất là từ *'hơn'* bổ nghĩa cho cụm danh từ *''6000 ha đất''*.

Chương 3. PHÂN TÍCH CÚ PHÁP CẤU TRÚC NGỮ ĐOẠN

Ảnh 3.1 Quy trình giai đoạn 3

3.1. Tạo bộ quy tắc

3.1.1. Văn phạm phi ngữ cảnh CFG

Văn phạm theo một cách đơn giản là các cú pháp đối với từng ngôn ngữ tự nhiên, bao gồm tập các quy tắc cấu tạo từ, và quy tắc liên kết các từ lại thành một câu. Ví dụ như câu "*Tôi đi học*." có thể phân tích cú pháp thành: *(S (N Tôi) (VP (V đi) (N học)) (PU .))*

Với mỗi Văn phạm G (Grammar) là một tập hợp gồm 4 thành phần:

$$G = (N, \sum, \mathrm{P}, \mathrm{S})$$

Trong đó:

- N là tập các từ vựng phụ trợ, gọi là ký hiệu không kết thúc (non-terminal). Ví dụ: *NP, VP…*
- $\sum$ là tập các từ của ngôn ngữ, gọi là ký hiệu kết thúc (terminal). Ví dụ: *Tôi, đi, học…*
- P là tập luật sản sinh. Ví dụ: $S \rightarrow NP\ VP$
- S là điểm khởi đầu cho các sản sinh trong P.

Văn phạm phi ngữ cảnh CFG là một văn phạm G, trong đó các luật trong P phải thỏa A → α, trong đó A là một non-terminal, $\alpha \in (N \cup \sum)^+$

3.1.2. Dạng chuẩn CNF

Một văn phạm CFG kí hiệu $G = (N, \sum, \mathrm{P}, \mathrm{S})$ được xem là đạt dạng chuẩn CNF nếu mỗi luật sản sinh trong P có một trong hai dạng:

- $A \rightarrow X_1\ X_2$ với $A, X_1, X_2 \in (N \cup \sum)$
- $A \rightarrow\ X$ với $A, X \in \sum$

Ví dụ : $S \rightarrow NP\ VP$

$NP \rightarrow NP\ NP$

$NP \rightarrow Anh$

3.1.3. Tạo bộ quy tắc

Tiến hành tạo tập văn phạm CFG theo chuẩn CNF từ bộ ngữ liệu có được từ giai đoạn tách từ và gắn nhãn từ loại, ta được bộ quy tắc gồm 2 loại :

- Lexicon : Tập chứa những luật có dạng $A \rightarrow X$
- Grammar : Tập chứa những luật có dạng $A \rightarrow X_1\ X_2$

```
20  A -> khó                35  M -> 2020
21  A -> quý                36  N -> nghị_quyết
22  AP -> trái_phép         37  N -> bản_thân
23  N -> băng_giá           38  N -> Bạn
24  R -> hàng               39  N -> đường
25  P -> Tôi                40  V -> chối
26  Nc -> người             41  N -> liều
27  V -> gắn                42  V -> khởi_công
28  V -> kẹp                43  Np -> Quốc_hội
29  V -> phạt               44  A -> mới
30  N -> xe_máy             45  V -> ghi_nhận
31  Np -> Nguyễn_Ngọc_Tuấn  46  V -> trình_chiếu
```

Ảnh 3.2: Một số luật trong tập Lexicon

```
1   S -> VP PU      21  NP -> M N      70  VP -> A V
2   S -> S PU       22  NP -> M NP     71  VP -> V AP
3   S -> NP VP      23  NP -> M Nu     72  VP -> V NP
4   S -> Np VP      24  NP -> M Np     73  VP -> V R
5   PP -> E N       25  NP -> A NP     74  VP -> V N
6   PP -> E NP      26  NP -> A N      75  VP -> V A
7   AP -> A A       27  NP -> N A      76  VP -> V PP
8   AP -> R A       28  NP -> N AP     77  VP -> R V
9   AP -> A AP      29  NP -> N N      78  VP -> PU VP
10  AP -> AP A      30  NP -> N Np     79  VP -> R VP
11  AP -> AP AP     31  NP -> N V      80  VP -> V VP
12  AP -> A C       32  NP -> N M      81  VP -> VP V
```

Ảnh 3.3: Một số luật trong tập Grammar

3.2. Xây dựng chương trình phân tích cú pháp tự động

3.2.1. Phân tích cú pháp bằng thuật toán CKY

Phân tích cú pháp trong xử lý ngôn ngữ tự nhiên là quá trình chia nhỏ một câu thành các thành phần của nó để máy tính có thể hiểu được.

Thuật toán CKY là thuật toán dùng để xác định xem một câu có được tạo ra từ văn phạm phi ngữ cảnh hay không. Thuật toán này có thể phân tích cú pháp của một câu dựa trên chiến lược bottom-up với chi phí là O(n^3) với trường hợp tệ nhất.

3.2.2. Xây dựng chương trình

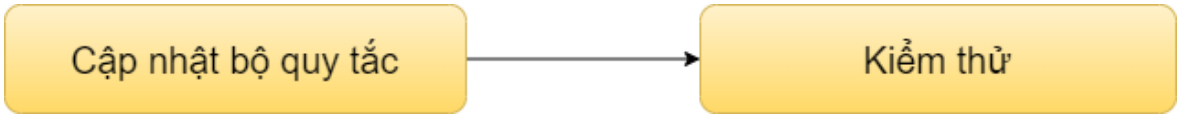
Input: Tập Grammar, Lexicon, Sentence

Output: Parsed Sentence

Các bước xây dựng chương trình:

- B1: Đọc tập Grammar, Lexicon, câu cần phân tích cú pháp
- B2: Tách câu thành các từ với thuật toán Maximum Matching, được tập Sentence
- B3: Xét tập Lexicon
  - Nếu các từ có thể chuyển về non-terminal, tiếp tục.
  - Nếu tồn tại một từ bất kì không thể chuyển về dạng non-terminal, báo lỗi và kết thúc.
- B4: Xét tập Grammar
  - Lần lượt duyệt từng từ. Điền các Grammar tìm được vào bảng T.
  - Nếu không tồn tại giá trị S trong bảng, báo lỗi và kết thúc.
- B5: Truy vết cây cú pháp.
- B6: Xuất kết quả, kết thúc.

Chương 4. **KIỂM THỬ**



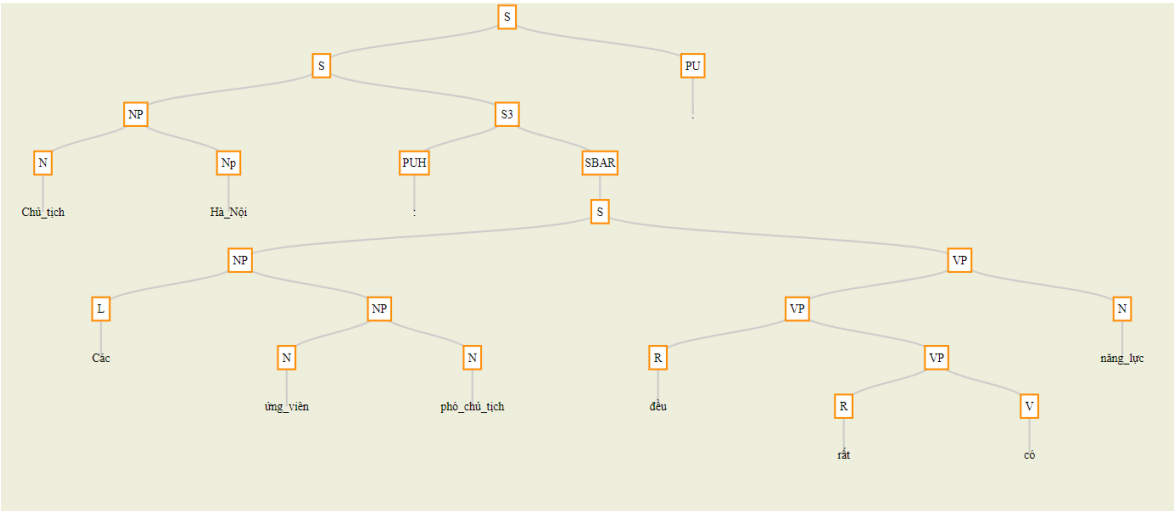
Ảnh 4.1. Qui trình trong giai đoạn cuối

4.1. Cập nhật bộ quy tắc

Sau khi chạy chương trình nhóm phát hiện ra một số câu có cấu trúc đặc biệt hơn các câu khác là câu tường thuật:

Ví dụ:

"Chủ tịch Hà Nội : Các ứng viên phó chủ tịch đều rất các năng lực ."



Ảnh 4.2

Ở trong trường hợp này chưa có luật sinh ra câu từ 1 câu nối bằng ':' (dấu hai chấm) nên sẽ sinh ra luật để dành riêng cho những câu có cấu trúc là câu tường thuật.

Từ đó nhóm đã cập nhật và bổ sung một vài luật mới trong bộ Grammar.

4.2. Kiểm thử

		Precision	**Recall**
CKY	First_Result	0.6616	0.6509
	Average	0.5858	0.5746
StarfordCoreNLP	First_Result	0.8239	0.8022

Bảng 4-1: So sánh kết quả phân tích cú pháp của CKY và StarfordCoreNLP

Chương 5. **KẾT LUẬN**

Như vậy trong đồ án này nhóm đã giải quyết được một cách đơn giản bài toán phân tích cú pháp cấu trúc ngữ đoạn. Nhóm đã biết quy trình các bước để tiếp cận bài trên: xử lý các câu như thế nào và để xây dựng một bộ ngữ liệu cần những bước nào. Biết rõ thêm về thuật toán CKY và biết được rõ các ưu và nhược điểm của thuật toán tách từ Maximum Matching và thuật toán CKY.

Từ đó nhóm đã biết cần phải làm gì để cải thiện kết quả của bài toán:

-Xây dựng bộ ngữ liệu lớn hơn.

-Có một thuật toán tách từ tốt hơn.

-Sử dụng một thuật toán phân tích cú pháp có trọng số để tăng thêm độ tin cậy.

TÀI LIỆU THAM KHẢO

[1] Ming Jiang, Jana Diesner *A Constituency Parsing Tree based Method for Relation Extraction from Abstracts of Scholarly Publications, 2019*

[2] Đinh Điền, *Xử lý ngôn ngữ tự nhiên*. NXB ĐHQGHCM, 2006