TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN
BỘ MÔN CÔNG NGHỆ TRI THỨC

BÙI THANH HUY - 9912567
LÊ PHƯƠNG QUANG - 9912653

NGHIÊN CỨU VÀ CÀI ĐẶT BỘ GÁN NHÃN TỪ LOẠI CHO SONG NGỮ ANH-VIỆT

LUẬN VĂN CỬ NHÂN TIN HỌC

GIÁO VIÊN HƯỚNG DẪN

GS.TSKH **HOÀNG KIẾM**

NIÊN KHÓA 1999 - 2003

Nhận xét của giáo viên hướng dẫn

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TP. Hồ Chí Minh, ngày…. tháng ….năm 2003
Giáo viên hướng dẫn

GS.TSKH Hoàng Kiếm

`

Nhận xét của giáo viên phản biện

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TP. Hồ Chí Minh, ngày…. tháng ….năm 2003

Giáo viên phản biện

`

Lời cảm ơn.

Đầu tiên, chúng em xin chân thành cảm ơn thầy giáo hướng dẫn, GS.TSKH Hoàng Kiếm, người đã tận tình hướng dẫn bọn em trong suốt quá trình làm luận văn. Đồng thời, chúng em xin gửi lời cảm ơn đến các thầy cô trong khoa Công Nghệ Thông Tin trường Đại học Khoa Học Tự Nhiên đã truyền đạt rất nhiều kiến thức quý báu cho chúng em.

Chúng em cũng muốn cảm ơn những người thân trong gia đình đã động viên, giúp đỡ và tạo điều kiện để chúng em có thể hoàn thành tốt luận văn này.

Cuối cùng, chúng em xin gửi lời cảm ơn thầy Đinh Điền và các bạn trong nhóm VCL đã giúp đỡ và hỗ trợ chúng em rất nhiều để hoàn thành luận văn này.

Tp. Hồ Chí Minh, 7-2003
Bùi Thanh Huy - Lê Phương Quang.

Mục lục

Mục lục ..... **ii**

Danh sách các hình ..... **vi**

Lời nói đầu ..... **vii**

Chương 1: Tổng quan **1**

1.1 Giới thiệu **2**

1.2 Tổng quan về gán nhãn từ loại **3**
1.2.1 Gán nhãn từ loại là gì? 3
1.2.2 Vai trò của gán nhãn từ loại 4

1.3 Các vấn đề gặp phải và hướng giải quyết trong bài toán gán nhãn từ loại **6**
1.3.1 Các vấn đề gặp phải khi giải quyết bài toán 6
1.3.2 Hướng giải quyết 7

1.4 Bố cục **8**

Chương 2: Cơ sở lý thuyết **9**

2.1 Máy học và xử lý ngôn ngữ tự nhiên **10**
2.1.1 Hướng tiếp cận thống kê 11
2.1.2 Hướng tiếp cận theo biểu trưng 12
2.1.2.1 Cây quyết định: 12
2.1.2.2 Danh sách quyết định 13
2.1.2.3 Phương pháp học hướng lỗi dựa trên các luật biến đổi trạng thái (TBL) 13
2.1.3 Hướng tiếp cận thay thế biểu trưng 14
2.1.3.1 Mạng Neural 14
2.1.3.2 Thuật toán di truyền (Genetic Algorithm : GA) 14

2.2 Một số giải thuật áp dụng cho bài toán gán nhãn từ loại **15**
2.2.1 Giải thuật học chuyển đổi dựa trên luật cải biến (TBL) 15
2.2.1.1 Sơ đồ của giải thuật TBL 17
2.2.1.2 Mô tả hoạt động của giải thuật 17
2.2.1.3 Trình bày giải thuật 20

`

2.2.1.4 Kết luận: 21
2.2.2 Mô hình mạng neural. 22
2.2.2.1 Giới thiệu: 22
2.2.2.2 Mạng neural: 22
2.2.2.3 Giải thuật gán nhãn từ loại dựa trên mạng neural 25
2.2.2.4 Từ điển: 27
2.2.3 Mô hình Maximum Entropy (ME): 28
2.2.3.1 Giới thiệu: 28
2.2.3.2 Các đặc trưng của gán nhãn từ loại: 29
2.2.3.3 Mô hình kiểm tra: 33
2.2.4 Mô hình TBL nhanh (FnTBL) 34
2.2.4.1 Giới thiệu giải thuật FnTBL: 34
2.2.4.2 Tính điểm và phát sinh luật: 36
2.2.4.3 Giải thuật FnTBL: 39

Chương 3: Mô hình 41

3.1 Một số khái niệm sử dụng trong mô hình: 42
3.1.1 Ngữ liệu(Corpus): 42
3.1.2 Ngữ liệu vàng(Golden Corpus) 44
3.1.3 Ngữ liệu huấn luyện(Training corpus): 45

3.2 Một số mô hình kết hợp hiện nay: 46
3.2.1 Mô hình kết hợp sử dụng nhiều mô hình liên kết 47
3.2.2 Phương pháp kết hợp dựa trên tính điểm cho các nhãn ứng viên 48
3.2.3 Phương pháp kết hợp dựa trên gợi ý của ngữ cảnh. 50
3.2.4 Phương pháp kết hợp dựa trên tính kế thừa kết quả của giải thuật TBL 51

3.3 Mô hình gán nhãn từ loại dựa trên song ngữ Anh-Việt 52
3.3.1 Sơ đồ hoạt động của mô hình: 55
3.3.1.1 Ngữ liệu huấn luyện: 56
3.3.1.2 Quá trình khởi tạo: 58
3.3.1.3 Quá trình huấn luyện: 58
3.3.1.4 Quá trình gán nhãn từ loại trên cặp câu song ngữ 61
3.3.2 Thuật giải 63
3.3.3 Khung luật (Template): 64
3.3.4 Cải tiến 66
3.3.5 Chiếu sang tiếng Việt 67

Chương 4: Cài đặt thử nghiệm và đánh giá kết quả 70

4.1 Cài đặt 71
4.1.1 Cài đặt bộ gán nhãn từ loại dựa trên mô hình kết hợp FnTBL và ME. 71
4.1.2 Cài đặt bộ gán nhãn từ loại có sử dụng thông tin tiếng Việt. 72
4.1.3 Cài đặt mô hình chiếu từ loại từ tiếng Anh sang tiếng Việt 73

4.2 Thử nghiệm..... 74
4.2.1 Thử nghiệm với các mô hình khởi tạo khác nhau. 74
4.2.1.1 Kết quả thử nghiệm dùng Unigram là giải thuật gán nhãn cơ sở. 75
4.2.1.2 Kết quả thử nghiệm với nhãn khởi tạo của mô hình Markov ẩn 78
4.2.1.3 Kết quả thử nghiệm dùng Maximum Entropy làm giải thuật gán nhãn cơ sở. 81
4.2.2 Thử nghiệm với các khung luật khác nhau cho giải thuật TBL nhanh 84
4.2.3 Kết quả gán nhãn từ loại khi dùng thông tin tiếng Việt..... 85

4.3 Nhận xét 85

Chương 5: Tổng kết 86

5.1 Kết quả đạt được..... 87

5.2 Hạn chế 88

5.3 Hướng phát triển: 89

Phụ lục A:Các tập nhãn của Penn Tree Bank 90

Phụ lục B: Bộ nhãn từ loại tiếng Việt. 92

Phụ lục C: Bảng ánh xạ từ loại từ tiếng Anh sang tiếng Việt... 93

Phụ lục D: Một số luật chuyển đổi. 95

Phụ lục E: Kết quả gán nhãn từ loại trong mô hình kết hợp không dùng thông tin tiếng Việt..... 97

Phụ lục F: Kết quả gán nhãn từ loại trong mô hình kết hợp có dùng thông tin tiếng Việt 99

`

Tài liệu tham khảo. 102

`

Danh sách các hình

Hình 1-1: Các giai đoạn của dịch máy 2

Hình 2-1: Sơ đồ hoạt động của giải thuật TBL. 17

Hình 2-2: Mô tả quá trình huấn luyện của giải thuật TBL......19

Hình 2-3:Mạng lan truyền 2 lớp 23

Hình 2-4: Cấu trúc của mô hình gán nhãn 25

Hinh 2-5: Cây từ điển trong mô hình mạng. 27

Hình 3-1: Cây cú pháp trong ngữ liệu......43

Hình 3-2: Sơ đồ hoạt động của mô hình gán nhãn từ loại trên ngữ liệu song ngữ Anh-Việt. 55

Hình 3-4: Mô hình huấn luyện cho nhãn tiếng Anh 60

Hình 3-5: Mô hình gán nhãn cho tiếng Anh trong ngữ liệu song ngữ Anh-Việt 61

Hình 4-1: Sơ gán nhãn cho mô hình kết hợp.....71

Hình 4-2: Sơ đồ mô hình gán nhãn sử dụng thông tin tiếng Việt......72

Hình 4-3: Sơ đồ mô hình chiếu từ loại sang tiếng Việt. 73

`

Lời nói đầu

Ngày nay, khi khoa học công nghệ phát triển hết sức mạnh mẽ, yêu cầu nắm bắt thông tin về khoa học, kỹ thuật, công nghệ nhanh chóng và chính xác là hết sức cần thiết. Hiện nay, đa số các tài liệu đều được viết bằng tiếng Anh. Do đó, việc chuyển các tài liệu này về tiếng Việt là điều rất cần thiết. Nếu làm được điều này, mọi người sẽ có được nhiều cơ hội tiếp cận với các thông tin tri thức mới. Nhưng công việc này tương đối khó khăn mặc dù hiện nay có khá nhiều hệ dịch tự động (như dịch trực tiếp, dịch qua ngôn ngữ trung gian, dịch dựa trên luật hoặc dịch dựa trên thống kê…) nhưng đa số các các hệ dịch này đều chưa đạt kết quả cao. Do đó, việc cải tiến chất lượng các hệ dịch máy luôn được quan tâm. Hiện nay, hệ dịch máy dựa trên chuyển đổi cú pháp được đánh giá khá cao. Hệ dịch máy này bao gồm khá nhiều giai đoạn như tiền xử lý, gán nhãn từ loại, phân tích hình thái, phân tích cú pháp, chuyển đổi trật tự từ, xử lý ngữ nghĩa,…

Dịch máy là một qui trình tương đối phức tạp, do vậy, trong luận văn này chúng tôi chỉ tập trung giải quyết một bài toán trong hệ dịch máy này, đó là giai đoạn gán nhãn từ loại. Đây là một bước cơ sở, làm nền tảng cho các giai đoạn sau. Kết quả của việc gán nhãn từ loại sẽ ảnh hưởng tới các giai đoạn khác. Trong luận văn này, ngoài việc cố gắng cải tiến kết quả của gán nhãn từ loại, chúng tôi còn sử dụng các thông tin có được sau khi gán nhãn từ loại để xây dựng một ngữ liệu về từ loại cho tiếng Việt. Nó sẽ giúp tiết kiệm rất nhiều thời gian và chi phí trong việc xây dựng ngữ liệu tiếng Việt, và ngữ liệu được tạo ra sẽ là nguồn dữ liệu vô cùng quý giá phục vụ cho các mục đích nghiên cứu về tiếng Việt khác.

Chương 1

Tổng quan

Trong chương này, chúng ta sẽ tìm hiểu tổng quan về gán nhãn từ loại và tầm quan trọng của gán nhãn từ loại trong xử lý ngôn ngữ từ loại nói chung và dịch máy nói riêng.

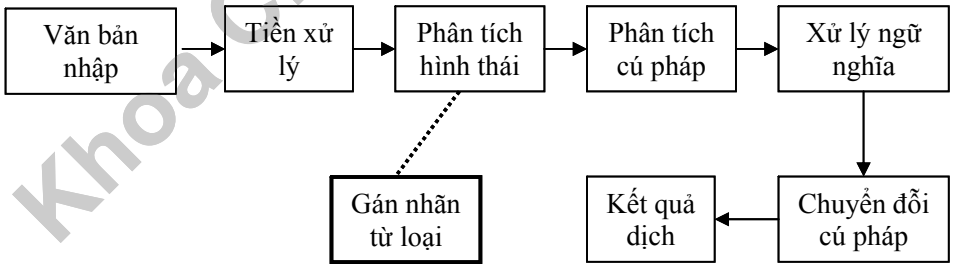
1.1 Giới thiệu

Từ trước đến nay, dịch máy luôn là một bài toán rất khó do ngôn ngữ tự nhiên rất phức tạp. Mặc dù cho đến nay đã có rất nhiều cải tiến nhằm tăng chất lượng dịch máy nhưng kết quả đạt được vẫn còn tương đối hạn chế.

Dịch máy là một quá trình khá phức tạp, gồm nhiều giai đoạn khác nhau như tiền xử lý, gán nhãn từ loại, phân tích cú pháp, chuyển đổi cú pháp, xử lý ngữ nghĩa… Các giai đoạn này đều ảnh hưởng rất lớn đến kết quả của quá trình dịch máy.

Gán nhãn từ loại là một giai đoạn khá quan trọng trong dịch máy. Nó có ảnh hưởng to lớn đến kết quả của các giai đoạn sau nó cũng như kết quả dịch máy. Việc gán nhãn từ loại chính xác không những ảnh hưởng đến kết quả của dịch máy, nó còn ảnh hưởng rất lớn đến kết quả của các bài toán khác trong xử lý ngôn ngữ tự nhiên, khai khoán dữ liệu như bài toán tìm từ đồng nghĩa, gần nghĩa, bài toán trích chọn thông tin, bài toán phân loại, làm chỉ mục…

Vị trí của gán nhãn từ loại trong hệ dịch máy dựa trên chuyển đổi cú pháp:



Hình 1-1: Các giai đoạn của dịch máy

1.2 Tổng quan về gán nhãn từ loại

1.2.1 Gán nhãn từ loại là gì?

Để hiểu rõ hơn về gán nhãn từ loại là gì thì trước tiên, chúng ta cần phải biết một số khái niệm về nhãn từ loại. Vậy nhãn từ loại là gì?

Trong một câu, mỗi từ đóng một vai trò nhất định. Để thể hiện chức năng ngữ pháp của mỗi từ, người ta sử dụng nhãn từ loại. Ví dụ như trong câu tiếng Anh sau:

I want to book a book.

Từ "book" có hai nhãn từ loại là động từ và danh từ.

Hoặc trong câu tiếng Việt sau:

Tôi đi học.

thì nhãn từ loại của từ "tôi" là đại từ, "đi học" là động từ

Trong luận văn này, chúng tôi chỉ tập trung vào việc gán nhãn cho câu tiếng Anh. Do đó, trong phần này chúng tôi sẽ chỉ đề cập các nhãn từ loại cho tiếng Anh.

Hiện nay trên thế giới có khá nhiều bộ nhãn từ loại. Trong luận văn này, chúng tôi sử dụng bộ nhãn của Pen Tree Bank, môt bộ nhãn khá phổ biến hiện nay. Dưới đây là một số nhãn trong bộ nhãn này :

IN Giới từ(Preposition or subordinating conjunction)

JJ Tính từ(Adjective)

NN Danh từ, số ít hay không đếm được(Noun, singular or mass)

NP Danh từ riêng số ít(Proper noun, singular)

RB Trạng từ(Adverb)

VB Động từ dạng nguyên thể không "to"(Verb, base form)

VBP Động từ không phải ngôi 3 số ít hiện tạ (Verb, non-3rd person singular present)

(Tham khảo thêm phần phụ lục A).

Trong một câu, mỗi từ đóng một vai trò ngữ pháp khác nhau, do đó tuỳ theo ngữ cảnh trong câu mà mỗi từ có một loại nhãn thích hợp. Nhưng để

xác định được nhãn từ loại của các từ trong một câu không đơn giản, do đa số các từ đều có nhiều từ loại khác nhau, tuỳ vào ngữ cảnh mà chúng ta có thể chọn nhãn từ loại thích hợp cho từ. Đây chính là công việc chủ yếu của gán nhãn từ loại, tìm nhãn từ loại chính xác cho các từ trong một câu.

1.2.2 Vai trò của gán nhãn từ loại

Gán nhãn từ loại là một giai đoạn trong quá trình dịch máy. Kết quả của gán nhãn từ loại sẽ ảnh hưởng rất lớn đến các giai đoạn khác.

Chẳng hạn như đối với việc chuyển đổi trật tự từ từ tiếng Anh sang tiếng Việt (đây là một công việc hết sức quan trọng trong quá trình dịch máy), nếu từ loại của các từ trong câu được đánh chính xác thì việc chuyển trật tự từ sẽ tốt hơn. Ví dụ như trong cụm danh từ sau:

Tiếng Anh: *A good book*

Câu dịch : *Một hay cuốn sách.*

Tiếng Việt: *Một cuốn sách hay.*

Trong ví dụ trên, từ "good" nằm trước từ "book" nhưng khi dịch ra tiếng Việt, ta phải đảo trật tự hai từ này thì câu tiếng Việt mới có ý nghĩa. Chính vì sự khác nhau về trật tự từ này nên khi dịch từ tiếng Anh sang tiếng Việt, cần phải có sự thay đổi trật tự từ cho thích hợp. Công việc chuyển đổi này dựa trên nhãn từ loại và cây cú pháp của tiếng Anh. Nếu giải quyết tốt vấn đề gán nhãn từ loại thì việc chuyển đổi sẽ gặp ít khó khăn hơn và kết quả đạt được sẽ tốt hơn.

Hoặc đối với vấn đề xử lý ngữ nghĩa (chọn nghĩa đúng cho một từ tuỳ theo ngữ cảnh), từ loại của từ có ảnh hưởng rất lớn. Ta thử xét ví dụ sau:

I want to book two books.

Trong câu trên, mặc dù hai từ "book" giống nhau nhưng chúng có vai trò ngữ pháp và ngữ nghĩa khác nhau. Do đó, muốn chọn nghĩa chính xác cho từng từ thì ta phải biết từ loại của từ đó. Nếu nhãn từ loại bị đánh sai thì sẽ dẫn đến việc chọn nghĩa cho từ sai hoàn toàn.

Ngoài ra, một ứng dụng khác của gán nhãn từ loại là sử dụng các thông tin đã có bên tiếng Anh để gán nhãn từ loại cho câu tiếng Việt. Đây cũng là một phần của luận văn này.

Hiện nay, khi công nghệ thông tin phát triển và các công trình nghiên cứu về ngôn ngữ, đặc biệt là tiếng Việt, ngày càng phát triển thì việc xây dựng một kho ngữ liệu bao gồm các thông tin về tiếng Việt hết sức cần thiết. Với mục đích đó, chương trình gán nhãn ngoài việc gán nhãn từ loại cho tiếng Anh còn sử dụng các thông tin về nhãn từ loại tiếng Anh đã có được kết hợp với các thông tin của tiếng Việt để gán nhãn từ loại cho câu tiếng Việt.

Muốn thực hiện được điều này thì dữ liệu đầu vào của ta cần có một câu tiếng Anh đã được gán nhãn và một câu tiếng Việt đã được dịch tương ứng với câu tiếng Anh trên. Nhãn từ loại trên câu tiếng Anh sẽ được lấy từ kết quả của chương trình. Như ví dụ sau:

Câu tiếng Anh: *I draw a picture.*

Câu tiếng Anh đã được gán nhãn từ loại: *I/PRP draw/VBP a/DT picture/NN*[1]

Câu tiếng Việt: *Tôi vẽ một bức tranh.*

Mục đích cần đạt được chính là câu tiếng Việt được gán nhãn từ như sau:

Tôi/P vẽ/V một/DT bức_tranh/N

Trong đó P là đại từ, V là động từ, DT là mạo từ, N là danh từ. Các nghiên cứu của các nhà ngôn ngữ học đã cho thấy giữa các ngôn ngữ luôn có một liên quan lẫn nhau về cấu trúc, từ loại, ... Do đó, việc chuyển đổi có thể thực hiện được nếu áp dụng một số quy tắc ánh xạ về sự tương ứng giữa các ngôn ngữ.

Bên cạnh đó, để thực hiện được việc này thì các từ tiếng Anh phải được liên kết với các từ tiếng Việt thông qua mối liên kết từ. Ví dụ như câu trên là:

[1] Các nhãn sủ dụng trong câu thuộc bộ nhãn từ loại của Penn Tree Bank, tham khao thêm ở phụ lục A

I --- > Tôi

Draw--------- >vẽ

A ---------- > một

Picture ---------- > bức tranh.

Công việc này được thực hiện qua việc sử dụng mô hình tìm liên kết từ cho song ngữ Anh-Việt, cụ thể ở đây là mô hình thống kê.

Bên cạnh đó, gán nhãn từ loại còn được áp dụng trên nhiều lĩnh vực khác. Trong các ứng dụng trích chọn thông tin, việc gán nhãn từ loại giúp cho quá trình tìm kiếm thông tin tốt hơn. Ngoài ra chúng ta còn có thể áp dụng gán nhãn từ loại vào các bài toán phân loại trong khai khoáng dữ liệu, bài toán tìm từ đồng nghĩa, từ gần nghĩa sẽ hiệu quả hơn.

Trong mức độ của một luận văn, do thời gian có hạn nên chúng tôi chỉ tập trung vào việc gán nhãn từ loại cho các câu tiếng Anh. Sau đó, dựa trên mối liên kết từ giữa tiếng Anh và tiếng Việt để ánh xạ từ loại của từ tiếng Anh sang cho từ tiếng Việt. Từ đó, chúng ta có thể xây dựng một ngữ liệu về từ loại cho tiếng Việt.

1.3 Các vấn đề gặp phải và hướng giải quyết trong bài toán gán nhãn từ loại

1.3.1 Các vấn đề gặp phải khi giải quyết bài toán

Khi thực hiện bài toán gán nhãn từ loại, ta gặp phải một số khó khăn. Khó khăn này chủ yếu là do các từ thường có nhiều hơn một từ loại.

Ta hãy xét câu sau:

I can can a can.

Trong câu này, ta thấy để xác định chính xác nhãn của từ "can" là một việc khá khó khăn. Từ "can" ở đây có ba từ loại là trợ động từ (MD), động từ (VB), danh từ (NN) tương ứng với các vị trí trong câu. Do đó, câu được gán nhãn từ loại đúng như sau:

I/PRP can/MD can/VB a/DT can/DT.

Vấn đề đặt ra của gán nhãn từ loại ở đây là giải quyết nhập nhằng đối với các từ có nhiều từ loại, làm thế nào xác định chính xác nhãn của từ đó trong câu.

1.3.2 Hướng giải quyết

Hiện nay, trên thế giới đã có rất nhiều hướng tiếp cận cho vấn đề này như Unigram, N-gram, mô hình Markov ẩn, Maximum-Entropy, TBL… Mỗi giải thuật đều có những ưu khuyết điểm riêng. Đồng thời, kết quả của các giải thuật này tương đối cao. Do đó, nếu chúng ta làm lại tất cả từ đầu thì sẽ tốn rất nhiều thời gian và công sức. Ngoài ra, do được phát triển từ lâu nên các hướng tiếp cận của này đã khai thác toàn bộ các thông tin có trong tiếng Anh để hỗ trợ cho việc gán nhãn từ loại. Nếu làm lại, chúng ta sẽ khó đạt kết quả cao hơn các mô hình trước đã làm được. Do đó, trong luận văn này, hướng giải quyết của chúng tôi là kế thừa các kết quả đã đạt được. Đồng thời, chúng ta sẽ tận dụng ưu điểm của các giải thuật đó để tạo ra một mô hình mới, một mô hình kết hợp các giải thuật khác nhau với nhau. Mô hình kết hợp này sẽ khai thác triệt để các ưu điểm của mỗi giải thuật có trong mô hình. Bên cạnh đó, chúng tôi còn sử dụng thêm các thông tin của tiếng Việt để cải tiến chất lượng của bộ gán nhãn từ loại. Đó là các thông tin về từ và từ loại của từ tiếng Việt tương ứng với từ tiếng Anh đang xét. Các thông tin này được rút ra từ từ điển và thông qua mối liên kết từ giữa tiếng Anh và tiếng Việt.

Sau một thời gian nghiên cứu về các hướng kết hợp đã có. Chúng tôi quyết định sử dụng mô hình được kết hợp bởi hai giải thuật Maximum Entropy (một mô hình tiếp cận theo hướng xác suất thống kê) của Adwait Ratnaparkhi và TBL nhanh[2] (một mô hình tiếp cận theo hướng biểu trưng) của hai nhà khoa học Radu Florian and Grace Ngai. Bên cạnh đó, chúng tôi có kết hợp sử dụng các thông tin của tiếng Việt như từ loại, ngữ nghĩa để làm

[2] Các giải thuật này sẽ được trình bày cụ thể ở chương 2

tăng kết quả chương trình. Sau khi chúng ta có được kết quả gán nhãn từ loại chính xác trên tiếng Anh chúng tôi sẽ thông qua mối liện kết từ giữa tiếng Anh và tiếng Việt để chọn nhãn từ loại cho từ tiếng Việt để tạo nên một ngữ liệu chính xác về từ loại của tiếng Việt.

1.4 Bố cục

Luận văn được chia làm 5 phần.

Chương 1: Tổng quan. Trình bày khái quát về dịch máy và khái quát công việc cần làm. Các vấn đề gặp phải trong bài toán gán nhãn từ loại và giới hạn vấn đề.

Chương 2: Cơ sở lý thuyết. Trình bày cơ sở lý thuyết của chương trình. Chương này sẽ giới thiệu một số hướng tiếp cận cho bài toán này. Đồng thời sẽ phân tích ưu khuyết điểm của chúng.

Chương 3: Mô hình. Đây chính là trọng tâm của luận văn. Chương này sẽ trình bày về mô hình được sử dụng trong chương trình, bao gồm thuật giải, các khung luật và các cải tiến của mô hình.

Chương 4: Cài đặt thực tiễn. Trình bày các kết quả thực tiễn đạt được của chương trình. Đồng thời, đánh giá, phân tích các kết quả đạt được.

Chương 5: Kết luận. Chương này sẽ tóm tắt lại những gì đã làm được trong và những hạn chế của chương trình. Bên cạnh đó sẽ đưa ra hướng phát triển cho chương trình.

Chương 2

Cơ sở lý thuyết

Trong chương này, chúng tôi sẽ trình bày các cơ sở lý thuyết và các hướng tiếp cận trước đây của mô hình gán nhãn từ loại.

2.1 Máy học và xử lý ngôn ngữ tự nhiên

Trong những năm gần đây, xử lý ngôn ngữ tự nhiên đã có một sự chuyển biến đột ngột từ việc xây dựng cơ sở tri thức về ngôn ngữ một cách thủ công sang tự động hóa từng phần hoặc toàn phần bằng cách sử dụng các phương pháp học, thống kê trên các tập ngữ liệu lớn. Sự chuyển biến này bắt nguồn từ các nguyên nhân sau:

- Sự xuất hiện ngày càng nhiều các tập ngữ liệu học lớn cho máy tính từ nhiều nguồn và trên nhiều ngôn ngữ khác nhau, ví dụ như Penn Tree Bank, Susanne, Brown, …
- Sự phát triển mạnh phần cứng máy tính, cho phép xử lý với một số lượng lớn thông tin và với các thuật toán có chi phí (thời gian, bộ nhớ) cao.
- Sự thành công bước đầu của các mô hình thống kê trong việc giải quyết một số vấn đề ngôn ngữ như nhận dạng tiếng nói, gán nhãn từ loại, phân tích cú pháp, dịch tự động song ngữ Anh-Việt, liên kết từ...
- Sự xuất hiện và phát triển của một số lượng lớn các giải thuật trong xử lý ngôn ngữ tự nhiên, cùng với sự khó khăn trong việc xây dựng cơ sở tri thức cho các phương pháp trước đây, đã làm cho các phương pháp trước đây không còn phù hợp với yêu cầu hiện nay nữa.

Các thống kê trong thời gian gần đây cho thấy xu hướng phát triển trong lĩnh vực xử lý ngôn ngữ tự nhiên: vào năm 1990 chỉ có 12,8% các công trình công bố ở hội nghị hằng năm của tổ chức ngôn ngữ học máy tính (Proceedings of Annnual Meeting of the Association for Computational Linguistics) và 15,4% công trình đăng trên tạp chí Ngôn ngữ học máy tính

(Computational Linguistics) liên quan dến hướng nghiên cứu sử dụng tập dữ liệu, các con số này vào năm 1997 lần lượt là 63,5% và 47,7%.

Về sau, các phương pháp thống kê áp dụng cho việc xử lý ngôn ngữ tự nhiên ngày càng phát triển. Các phương pháp này đặc biệt phù hợp cho việc rút trích tri thức từ vựng và khử nhập nhằng, bên cạnh đó là các nghiên cứu ứng dụng cho việc suy diễn ngữ pháp, phân tích thô, xử lý ngữ nghĩa, chuyển đổi cú pháp...

Các phương pháp máy học được áp dụng trong lĩnh vực xử lý ngôn ngữ tự nhiên được phân loại như sau:

- Hướng tiếp cận theo thống kê (stochastic approach).
- Hướng tiếp cận theo biểu trưng (symbolic approach): học theo ví dụ (instance – based learning), cây quyết định (decision tree), logic quy nạp (inductive logic), phân tách tuyến tính theo ngưỡng (threshold linear separator)... Trong các phương pháp này, đáng chú ý nhất ; là phương pháp học dựa trên các luật chuyển đổi (TBL – Transformation Based Learning). Phương pháp này cho phép đưa ra tập các khung luật tổng quát có thể giải quyết các vấn đề nhập nhằng tương tự nhau (như trong bài toán gán nhãn từ loại).
- Hướng tiếp cận theo biểu trưng thay thế (sybsymbolic approach): mạng nơ-ron (neural network), thuật toán di tuyền (genetic algorithm), ...
- Các hướng khác: học không giám sát (unsupervised approach) và hướng các tiếp cận kết hợp.

2.1.1 Hướng tiếp cận thống kê

Hướng tiếp cận thống kê được xem là một hướng tiếp cận mô tả quá trình thế giới thực tạo ra dữ liệu quan sát được. Các mô hình trong hướng tiếp cận thống kê thường được thể hiện dưới dạng một mạng thống kê các mối quan hệ phụ thuộc giữa các biến ngẫu nhiên. Mỗi nút của mạng có một

phân phối, và từ những phân phối này chúng ta cố gắng tìm ra các phân phối chung của dữ liệu quan sát. Các hướng tiếp cận khác nhau của phương pháp này xuất phát từ cách tạo ra mạng thống kê và cách kết hợp các phân phối của mỗi nút.

Có khá nhiều mô hình trong hướng tiếp cận này được áp dụng trong lĩnh vực xử lý ngôn ngữ tự nhiên. Ví dụ như mô hình phân loại Bayes ngây thơ (Naïve Bayes classifier), nguyên lý hỗn loạn cực đại (Maximum Entropy Principle), mô hình Markov ẩn (Hidden Markov model). Các mô hình này được áp dụng để giải quyết nhiều bài toán trong xử lý ngôn ngữ tự nhiên như : sửa lỗi chính tả theo ngữ cảnh, gán nhãn từ loại, nhận dạng mệnh đề, nhận dạng tiếng nói …

Hiện nay trong bài toán gán nhãn từ loại thì hướng tiếp cận thống kê được xem là một trong những hương tiếp cận có kết quả cao. Trong luận văn chúng tôi có sử dụng một trong các hướng tiếp cận này là Maximum Entropy

2.1.2 Hướng tiếp cận theo biểu trưng

Tiếp cận theo biểu trưng gồm một số hướng sau đây

2.1.2.1 Cây quyết định:

Các phương pháp dựa trên cây quyết định được áp dụng vào việc học giám sát các mẫu là một trong những cách tiếp cận thông dụng của trí tuệ nhân tạo để giải quyết các bài toàn về phân lớp. Phương pháp cây quyết định học dựa trên việc xấp xỉ hàm đích có giá trị rời rạc mà trong đó hàm học được biểu diễn bằng cây quyết định. Phương pháp này học trên một tập thực thể đã được phân lớp từ trước và kết quả nhận được là một tập các câu hỏi dùng để phân loại các thực thể mới. Nó sẽ cố gắng lựa chọn các câu hỏi sao cho sự phân loại các thực thể thành các tập con mà trong đó các tập con thuần nhất nhất. Quá trình phân chia các thực thể lại tiếp tục trên các tập con chưa thuần nhất cho đến khi tất cả các tập con đều thuần nhất. Các cây quyết định được dùng để lưu trữ các luật được rút ra trong quá trình học dưới dạng

các cấu trúc phân cấp tuần tự, qua đó phân hoạch dữ liệu một cách đệ quy. Cây quyết định đã được áp dụng từ lâu trong các ứng dụng trong các lĩnh vực như : thống kê, nhận dạng dạng mẫu, lý thuyết quyết định và xử lý tín hiệu số. Trong các ứng dụng này, cây quyết định được dùng để thao tác trên dữ liệu với mục đích mô tả phân loại và tổng quát hoá.

Trong lĩnh vực xử lý ngôn ngữ tự nhiên, ứng dụng của cây quyết định cũng rất đáng chú ý trong việc xử lý nhập nhằng trong các bài toán gán nhãn từ loại, phân tích cú pháp, phân loại tài liệu …

2.1.2.2 Danh sách quyết định

Danh sách quyết định bao gồm một danh sách các luật kết hợp có thứ tự, các luật kết hợp này sẽ được áp dụng vào dữ liệu bằng cách kiểm tra xem trong danh sách các luật, luật phù hợp đầu tiên sẽ được chọn. phương pháp này phù hợp cho các lĩnh vực cần tránh sự phân mảnh dữ liệu.

Trong xử lý ngôn ngư tự nhiên, phương pháp này được áp dụng để giải quyết các nhập nhằng về mặt từ vựng

2.1.2.3 Phương pháp học hướng lỗi dựa trên các luật biến đổi trạng thái (TBL)

Phương pháp TBL được giới thiệu bởi Eric Brill, thuộc đại học Pennsylvania, vào năm 1993. Hiện nay phương pháp này là một trong những phương pháp được áp dụng rộng rãi trong các lĩnh vực của xử lý ngôn ngữ tự nhiên. Trong quá trình huấn luyện, phương pháp này sẽ tiến hành tạo ra các luật ứng viên dựa trên các mẫu luật cho trước, các luật úng viên này sẽ được tính điểm dựa trên số trường hợp luật chỉnh ngữ liệu từ sai thành đúng và từ đúng thành sai. Các luật có điểm cao sẽ được giữ lại cho việc gán nhãn. Đây là một trong những phương pháp rất trực quan và linh động. Chúng ta có thể can thiệp vào quá trình học của thuật toán bằng cách quản lý mẫu luật.

2.1.3 Hướng tiếp cận thay thế biểu trưng

2.1.3.1 Mạng Neural

Mạng Neural là một trong những phương pháp phổ biến trong lĩnh vực máy học. Mạng Neural học dựa trên số bằng cách xác định một hàm sao cho càng khớp với đường cong đi qua các điểm không gian gian của các mẫu huấn luyện càng tốt. Các yếu tố ngữ cảnh ảnh hưởng đến quyết định nào đó được biểu diễn bằng các giá trị đã được lượng hoá, nhân với trọng số và gán cho các nút của tầng nhập. Chính việc lượng hoá các các yếu tố ngữ cảnh đã làm cho phương pháp này không còn trực quan về mặt ngôn ngữ học. Ngoài ra, không phải yếu tố ngôn ngữ nào cũng có thể lượng hoá dễ dàng, điều này làm cho phương pháp mạng Nueral không thể áp dụng trong hầu hết các bài toán trong xử lý ngôn ngữ tự nhiên. Ngoài ra, phương pháp mạng Neural có độ rộng ngữ cảnh chính là số nút của tầng nhập nên phương pháp này không có tính linh động trong trường hợp ngữ cảnh thay đổi. Trong xử lý ngôn ngữ tự nhiên mạng Neural được áp dụng trong các bài toán nhận dạng ký tự (OCR), gán nhãn từ loại, nhận dạng và tổng hợp tiếng nói. Các mô hình xử lý cơ bản sử dụng các mạng Neural feed-forward đa tầng được huấn luyện bằng giải thuật lan truyền ngược, bên cạnh đó cũng xuất hiện kiểu mạng hồi quy và kết hợp các mạng Neural đơn lẻ.

2.1.3.2 Thuật toán di truyền (Genetic Algorithm : GA)

Giải thuật di truyền đã được dùng để rút ra loại từ và cấu trúc cú pháp từ nguồn thông tin duy nhất là tập dữ liệu không được chú thích và không sử dụng thêm tri thức nào. Hướng tiếp cận này cũng được kết hợp với phương pháp học không giám sát cho bài toán phân vùng.

Bài toán gán nhãn từ loại là một trong những bài toán xuất hiện tương đối sớm trong lĩnh vực xử lý ngôn ngữ tự nhiên,và nó cũng là một bài toán

làm tiền đề cho các bài toán khác (chẳng hạn như bài toán phân tích cú pháp, chuyển đổi cây cú pháp, xử lý ngữ nghĩa …). Kết quả của nó sẽ ảnh hưởng tới các giai đoạn sau. Chẳng hạn như trong bài toán phân tích cú pháp : nếu như kết quả việc gán nhãn từ loại sai thì sẽ dẫn tới việc chọn cây cú pháp và cấu trúc cây sai. Một cấu trúc câu có thể bị thay đổi nếu như từ loại của một từ nào đó trong câu bị thay đổi. Trong bài toán xử lý ngữ nghĩa, một trong những yếu tố quan trọng nhất đó là từ loại. Một từ có từ loại sai thì dẫn đến việc chọn nghĩa cho từ sẽ sai. Ví dụ trong câu "I can can a can" cả 3 từ "can" trong câu đều có ý nghĩa khác nhau. Từ "can" đầu tiên là trợ động từ nó có nghĩa là "có thể", từ "can" thứ 2 là động từ chính của câu nó có ý nghĩa là "đóng" (hay "đóng hộp") còn từ "can" cuối cùng là một danh từ có nghĩa là "cái hộp". Nếu như một trong 3 từ "can" này bị gán sai nhãn từ loại thì việc chọn nghĩa cho câu trên chắc chắn sai.

Vì bài toán gán nhãn từ loại là một trong những bài toán quan trọng làm tiền đề cho các bài toán khác trong xử lý ngôn ngữ tự nhiên nên bài toán này đã được rất nhiều người quan tâm. Cho đến hiện nay đã có nhiều giải thuật cho kết quả có độ chính xác khá cao, chúng tôi xin giới thiệu một số phương pháp cho kết quả khá cao trong vấn đề này.

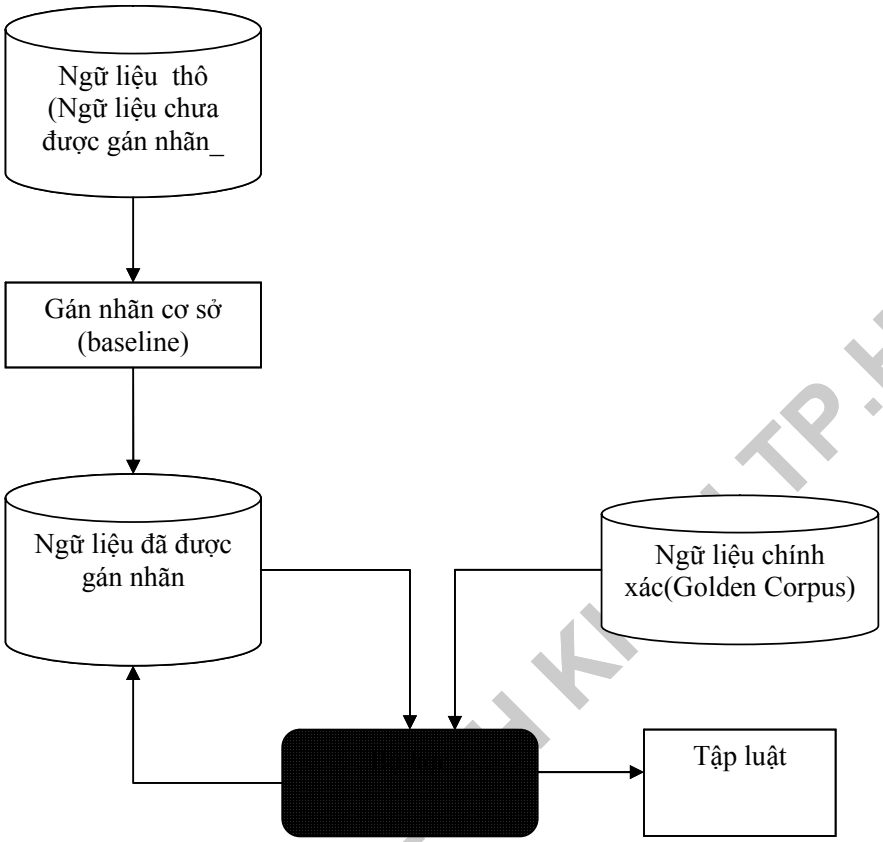
2.2 Một số giải thuật áp dụng cho bài toán gán nhãn từ loại

2.2.1 Giải thuật học chuyển đổi dựa trên luật cải biến (TBL)

Giải thuật TBL (Transformation-Based Learning)là một giải thuật học giám sát được Eric Brill đưa ra trong luật văn tiến sĩ của ông năm 1993. Giải thuật TBL được áp dụng rộng rãi trong xử lý ngôn ngữ tự nhiên và được đánh giá là một trong những giải thuật cho kết quả khả quan nhất đối với các bài toán xử lý ngôn ngữ tự nhiên như : các bài toán tách từ, tách câu, gán nhãn từ loại, phân tích cú pháp khử nhập nhằng ngữ nghĩa…

Trong các bài toán trên, kết quả nhận được khi sử dụng giải thuật TBL là khá cao, có thể so sánh với nhiều giải thuật tiên tiến khác. Sở dĩ giải thuật TBL có được những kết quả cao như vậy là do nó có được những ưu điểm mà nhiều giải thuật khác không có, đó là tính trực quan, dễ hiểu, dễ kiểm soát. Chúng ta có thể quan sát, theo dõi và can thiệp vào quá trình học cũng như quá trình thực thi của giải thuật. Một đặc điểm nổi bật khác của giải thuật TBL là tính kế thừa, giải thuật khã năng phát triển lên từ kết quả trung gian, kết quả đầu ra của một giải thuật khác.

2.2.1.1 Sơ đồ của giải thuật TBL



Hình 2-1: Sơ đồ hoạt động của giải thuật TBL.

2.2.1.2 Mô tả hoạt động của giải thuật

❖ **Quá trình huấn luyện**

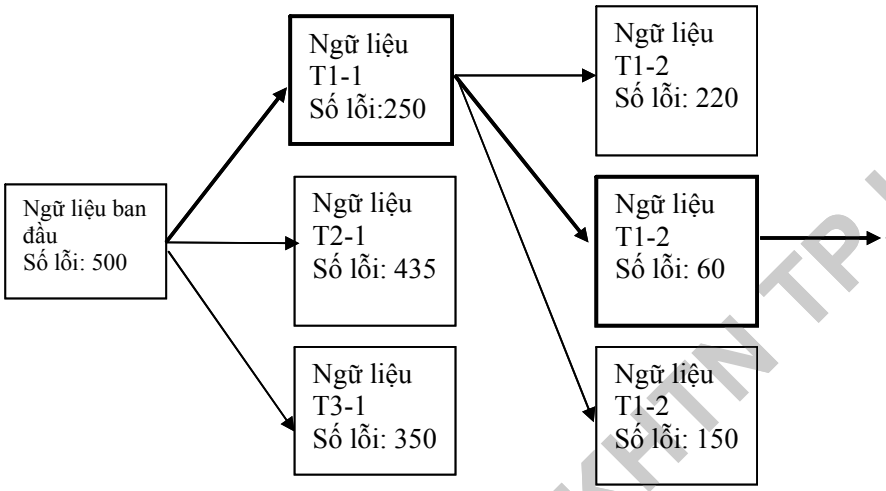
Quá trình học của giải thuật được bắt đầu với một ngữ liệu thô (ngữ liệu chưa được gán nhãn). Sau đó, ngữ liệu này được tiến hành gán nhãn cơ sở, hay còn gọi là gán nhãn ban đầu(initial state). Việc gán nhãn cơ sở chỉ là gán cho ngữ liệu một giá trị ban đầu. Việc gán nhãn có sở có thể không chính xác, chẳng hạn gán nhãn từ loại cho các từ trong câu là danh từ, hoặc cũng có rất chính xác, chúng ta có thể chọn kết quả của một

giải thuật nào đó làm nhãn cơ sở. Nhãn này có thể chính xác hoặc không chính xác. Sau khi dữ liệu đã nhận trạng thái khởi tạo, dữ liệu này được so sánh với các trạng thái đúng của chúng (ngữ liệu vàng). Qua việc so sánh này, các lỗi của dữ liệu hiện hành được xác định. Thông qua các lỗi này chúng xác định được các luật chuyển đổi nhằm biến đổi ngữ liệu từ trạng thái ngây thơ (trong quá quá trình khởi tạo) hay trạng thái hiện hành (đã có áp dụng qua luật chuyển đổi) thành dạng giống hơn so với các trạng thái đúng. Một tập hợp các khung luật lúc này được sử dụng để tạo ra các luật ứng viên. Các khung luật được xác định trước như quy tắc xác định trạng thái "ngây thơ" ở giai đoạn khởi tạo. Mỗi khung luật chứa các biến điều kiện chưa xác định giá trị. Ví dụ mẫu luật sau:

"Nếu nhãn đứng trước X là Z thì đổi nhãn X thành Y". X, Y, và Z là các biến. Với mỗi bộ giá trị của X, Y, Z ta được một luật phát sinh từ mẫu luật này. Trong khung luật trên X và Y là các biến, nó có thể nhận bất kì một giá trị nào trong bộ nhãn mà chúng ta đề ra.

Thuật toán sinh ra các luật ứng viên bằng cách thay các giá trị có thể vào cho các biến trong khung luật. Luật ứng viên sau khi được tạo ra nó sẽ được áp dụng vào trong ngữ liệu đang được gán nhãn hiện hành để tạo ra ngữ liệu được gán nhãn khi áp dụng luật ứng viên này. Ngữ liệu được gán nhãn theo luật ứng viên vừa tạo ra sẽ được so sánh đối chiếu với ngữ liệu đúng (hay ngữ liệu vàng). Khi so sánh với ngữ liệu chính xác chúng ta sẽ biết được luật ứng viên vừa tạo ra chỉnh ngữ liệu từ đúng thành sai bao nhiêu trường hợp và từ sai thành đúng bao nhiêu trường hợp. Từ đó ta tính ra được điểm cho luật ứng viên này. Điểm của luật ứng viên này chính là hiệu số giữa số trường hợp luật chỉnh ngữ liệu từ sai thành đúng và số trường hợp luật chỉnh ngữ liệu từ đúng thành sai. Sau khi tất cả các luật ứng viên được tạo ra chúng ta sẽ biết được luật ứng viên nào có điểm cao nhất, luật ứng viên có điểm cao nhất sẽ được giữ lại cho các lần gán nhãn sau nếu như luật này thoả mãn điều kiện nó có điểm lớn hơn một

mức ngưỡng mà chúng ta cho trước. Luật này sẽ được áp dụng để chuyển ngữ liệu ở trạng thái thứ k sang trạng thái mới trạng thái thứ k+1. Ngữ liệu ở trạng thái mới này lại lần lượt thử trên các luật ứng viên để chọn ra luật tối ưu mới. Quá trình này sẽ được lặp đi lặp lại cho đến khi không còn có luật tối ưu nào có điểm lớn hơn mức ngưỡng.



Hình 2-2: Sơ đồ quá trình huấn luyện của giải thuật TBL.

Kết thúc giai đoạn huấn luyện chúng ta sẽ thu được một danh sách các luật tối ưu. Các luật tối ưu này sẽ được sử dụng vào quá trình thực thi của giải thuật theo thư tự các luật có điểm cao được áp dụng trước các luật thấp được áp dụng sau.

❖ **Quá trình thực thi**

Cũng tương tự như quá trình huấn luyện, dữ liệu muốn gán nhãn phải được gán nhãn cơ sở. Quá trình gán nhãn cơ sở này giống như quá trình gán nhãn cơ sở của quá trình học. Nhãn cơ sở này có thể là nhãn ngây thơ cũng có thể là nhãn chính xác hay đầu ra của một mô hình gán nhãn khác.

Chúng ta lần lượt áp dụng các luật tối ưu mà chúng ta nhận được trong quá trình học vào ngữ liệu. các luật có số điểm cao trong quá trình huấn luyện sẽ được áp dụng trước các luật có điểm thấp được áp dụng sau.

Sau quá trình áp dụng tất cả các luật chúng ta sẽ nhận được một kết quả với nhãn chính xác cho từng từ.

2.2.1.3 Trình bày giải thuật

Trong bài toán gán nhãn từ loại chúng ta có một số quy ước sau:

T : tập hợp các nhãn từ loại ví dụ *T*= {PRP,VB,NN,… }

μ *:* vị từ được định nghĩa trên không gian C+, C+ thường là một dãy các trạng thái, ví dụ (word_-1,PRP) $\wedge$ (word_1,NN) hay dãy các mẫu như : (word_-1=a) v (word_-1=the). Các vị từ là các thể hiện của khung luật

Một luật l được định nghĩa như một cặp (μ , t) gồm một vị từ μ và một nhãn từ loại t. Luật l sẽ được biểu diễn dưới dạng là μ => t nghĩa là luật l sẽ được áp dụng trên mẫu x nếu vị từ vị từ μ thoả mãn, khi đó mẫu x sẽ được gán nhãn mới t.

Cho một trạng thái c=(x,t) và luật l=(μ,t'), thì trạng thái kết quả của việc áp dụng luật l trên trạng thái c được định nghĩa :

$$l(c) = \begin{cases} c & \text{Nếu } \mu(c) = \text{Sai} \\ (x, t') & \text{Nếu } \mu(c) = \text{Đúng} \end{cases}$$

D : tập các mẫu huấn luyện đã được gán nhãn đúng.

Điểm được tính cho mỗi luật l chính là hiệu số khác biệt giữa kết quả thực hiện của luật l so với tình trạng ban đầu theo công thức :

$$\text{Diem}(l) = \sum_{c \in D} diem(l(c)) - \sum_{c \in D} diem(c)$$

trong đó :

$$diem((x, t)) = \begin{cases} 1 \text{ Nếu t = True (x)} \\ 0 \text{ Nếu t} \neq \text{True(x)} \end{cases}$$

❖ **Giải thuật TBL nguyên thuỷ được trình bày như sau :**

Bước 1 :khởi tạo mỗi mẫu x trong tập huấn luyện với một nhãn thích hợp nhất. Chẳng hạn với từ I thì xác xuất xuất hiện cao nhất là PRP, ta gọi ngữ liệu ở bước này là D_0.

Bước 2 : Xem xét tất cả các luật chuyển đổi l tác động trên dữ liệu D_k ở lượt thứ k và chọn luật nào có diem(r) cao nhất và áp dụng luật l này trên dữ liệu D_k để nhận được dữ liệu mới D_{k+1}. ta có $D_{k+1} = l(D_k) = \{l(c)|c \in D_k\}$ nếu không còn một luật nào thoả diem(l) > β thì giải thuật dừng. β là mức ngưỡng mà chúng ta chọn trước. Với mỗi bài toán chúng ta có thể chọn mức ngưỡng β khác nhau. Mức ngưỡng β được chọn dựa trên yêu cầu thực tế bài toán.

k=k+1;

Bước 3 : lặp lại từ bước 2.

Khả năng dừng (hội tụ) của giải thuật: gọi Err_k là số lỗi so với ngữ liệu chính xác của ngữ liệu hiện hành sau khi áp dụng luật l, ta có $Err_{k+1} = Err_k$ - Diem(l) , do Diem(l) > 0, nên $Err_{k+1} < Err_k$ với mọi k và $Err_k \in N$ nên thuật toán sẽ dừng sau một số bước hữu hạn

Chi phí của thuật toán : O(n*t*c) trong đó n là kích thước của tập huấn luyện (số lượt từ); t là kích thước của tập luật chuyển đổi khả dĩ (số luật ứng viên); c: là kích thước của ngữ liệu thoả mãn điều kiện áp dụng luật.

2.2.1.4 Kết luận:

Mô hình này là một phương pháp tương đối uyển chuyển trong các phương pháp gán nhãn từ loại. Ta có thể thêm bớt thay đổi các đặc trưng của nó. Tuy nhiên hạn chế lớn của mô hình là đòi hỏi một bộ dữ liệu tương đối lớn thì kết quả sẽ khả quan hơn.

2.2.2 Mô hình mạng neural.

2.2.2.1 Giới thiệu:

Đối với từ, hiện tượng nhập nhằng về từ loại rất hay xảy ra. Như trong tiếng Anh từ "store" vừa có thể vừa là danh từ vừa là động từ. Thông thường các sự nhập nhằng này được giải quyết bằng cách dựa vào ngữ cảnh của từ. Ví dụ như câu sau:

Today, hard drive can store a large information.

Trong câu trên, từ "store" chỉ có một từ loại là động từ nguyên thể.

Gán nhãn từ loại là một hệ thống tự động gán nhãn cho các từ sử dụng các thông tin có trong ngữ cảnh. Ứng dụng chủ yếu của gán nhãn tồn tại trong nhiều lãnh vực như nhận dạng tiếng nói, tổng hợp tiếng nói, dịch máy và sự phục hồi thông tin.

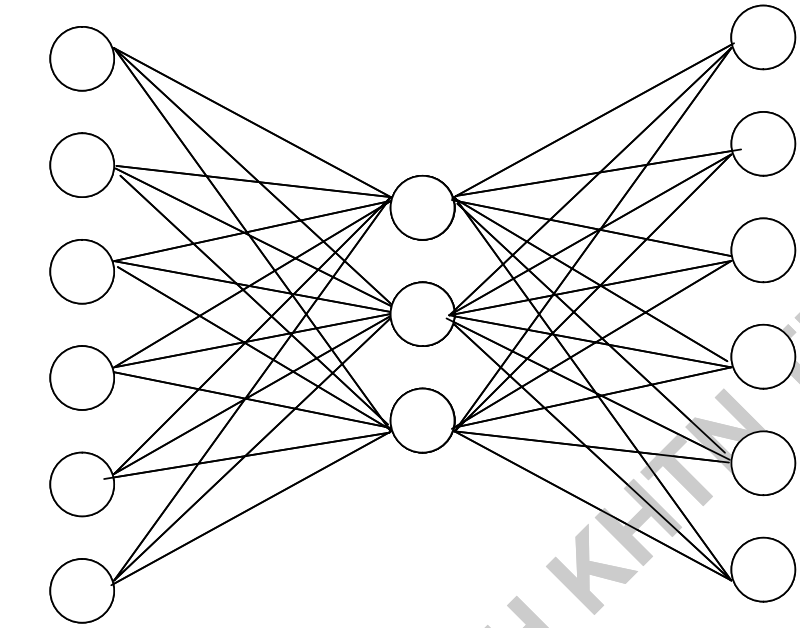
Có khá nhiều hướng để tiếp cận với vấn đề gán nhãn từ loại như thống kê, dùng luật, máy học. Trong phần này, ta sẽ tìm hiểu về một hệ thống gán nhãn sử dụng "mạng neural nhân tạo". Đây là một mô hình khá thông dụng trong lĩnh vực nhận dạng tiếng nói. Bên cạnh đó, nó còn có thể áp dụng trong lĩnh vự nhận dạng văn bản. Và gần đây là gán nhãn từ loại, cũng được áp dụng tương đối thành công.

2.2.2.2 Mạng neural:

Mạng neural nhân tạo bao gồm một số lượng lớn các đơn vị xử lý đơn giản. Các đơn vị này được nối liền trực tiếp với nhau bằng các liên kết trọng số. Liên quan đến mỗi đơn vị là các giá trị hoạt hoá. Thông qua các mối liên kết, các giá trị này sẽ được lan truyền đến các đơn vị khác.

Mạng gồm ba lớp: lớp nhập(input), lớp ẩn, và lớp xuất(output). Mỗi nút trong lớp nhập nhận giá trị của một biến độc lập và chuyển vào mạng. Dữ liệu từ tất cả các nút trong lớp nhập được tích hợp - ta gọi tổng trọng hoá- và chuyển kết quả cho các nút trong lớp ẩn. Gọi là "ẩn", vì các nút trong lớp

này chỉ liên lạc với các nút trong các lớp nhập và xuất; Tương tự các nút trong lớp xuất cũng nhận các tín hiệu tổng trọng hoá từ các nút ẩn. Mỗi nút trong lớp xuất tương ứng một biến phụ thuộc.



Hình 2-3:Mạng lan truyền 2 lớp

Trong quá trình xử lý mạng, sự hoạt động được lan truyền từ các đơn vị nhập thông qua các đơn vị xuất tới các đơn vị lớp xuất. Ở mỗi vị trí j, trọng số nhập $a_i w_{ij}$ được cộng vào và tham số về độ lệch θ được cộng vào:

$$net_i = \sum_i a_i w_{ij} + \theta_j$$

Kết quả của mạng nút nhập net_j sau đó được thông qua một hàm giải phẫu (ta thường sử dụng hàm logic) để hạn chế khoảng giá trị của a_j trong khoảng [0,1]

$$a_{ai} = \frac{1}{1 + e^{-net_j}}$$

Mạng học bằng cách thích nghi trọng số của các liên kết của các đơn vị, cho đến khi kết xuất đúng được tạo ra. Một phương pháp mở rộng được sử dụng là lan truyền ngược mà nó sẽ giảm độ dốc trên bề mặt. Trọng số cập nhật w_{ij}

$$\Delta w_{ij} = \eta a_{pi} \delta_{pj}$$

$$\delta_{pj} = \begin{cases} a_{pj}(1-a_{pj})(t_{pj}-a_{pj}) & \text{nếu j là một đơn vị xuất} \\ a_{pj}(1-a_{pj})\sum_k \delta_{pk} w_{jk} & \text{nếu j là một đơn vị ẩn} \end{cases}$$

Ở đây, t_p là một đích nhắm của vector xuất mà mạng phải học.

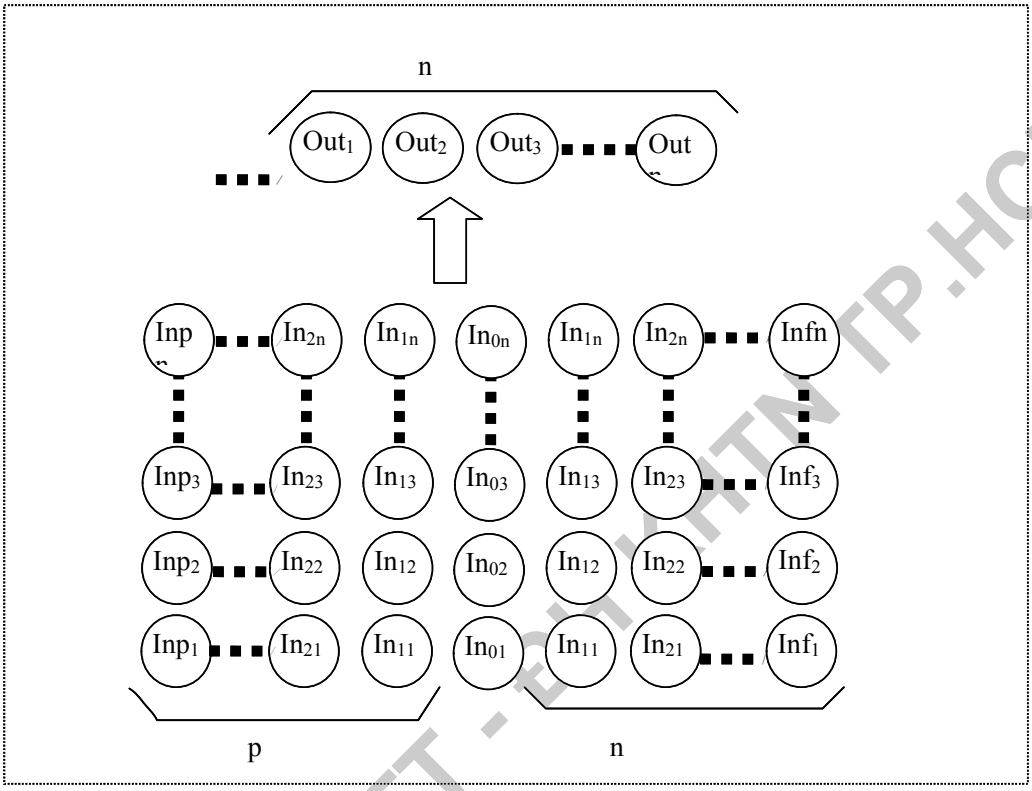
Huấn luyện mạng với sự lan truyền ngược, các luật sẽ bảo đảm một cực tiểu địa phương của bề mặt lỗi sẽ được tìm thấy, mặc dù điều này không cần thiết cho các biến cục bộ.

Để tăng tốc độ huấn luyện, một thuật ngữ về xung lượng được giới thiệu trong công thức cập nhật.

$$\Delta w_{ij}(t+1) = \eta a_{pi} \delta_{pj} + \alpha \Delta w_{ij}(t)$$

2.2.2.3 Giải thuật gán nhãn từ loại dựa trên mạng neural

Mạng gán nhãn bao gồm một mạng "multilayer perceptron networks " (MLP-nets works)([5]) và các từ vựng. (Hình 2-2)



Hình 2-4: Cấu trúc của mô hình gán nhãn

Trong lớp xuất của mạng MLP, mỗi đơn vị tương ứng với một nhãn trong tập nhãn. Mạng sẽ học trong suốt quá trình huấn luyện để làm kích hoạt các đơn vị xuất mà biểu diễn cho các nhãn đúng và ngừng kích hoạt đối với tất cả các đơn vị xuất khác. Từ đây, trong mạng huấn luyện, các đơn vị xuất có độ hoạt động cao nhất sẽ được chỉ ra, mà nhãn nên được gán vào từ mà đang được xử lý.

Đầu vào của mạng sẽ bao gồm tất cả các thông tin mà hệ thống có về từ loại của từ hiện tại, p từ trước và f từ sau. Để chính xác hơn, với mỗi nhãn từ loại pos_j và mỗi p+f+1 từ trong ngữ cảnh, có các đơn vị nhập mà sự hoạt động in_{ij} đại diện cho xác suất của từ $word_i$ có nhãn là pos_i .

Đối với mỗi từ đang được gán nhãn và các từ theo sau, xác suất từ loại từ vựng $P(pos_j|word_i)$ là tất cả chúng ta biết về từ loại. Xác suất này không gây ra ảnh hưởng ngữ cảnh nào. Vì vậy, chúng ta sẽ nhận đầu vào sau tượng trưng cho các nhãn hiện tại của từ và các từ theo sau:

$$in_{ij} = P(pos_j| word_i) \qquad \text{nếu } i \geq 0.$$

Đối với các từ phía trước, có nhiều thông tin có sẵn, bởi vì chúng đã được gán nhãn từ loại. Các giá trị hoạt động của đơn vị xuất tại một thời điểm xử lý được sử dụng thay vì xác suất từ loại của từ vựng:

$$in_{ij} = out_j(t+i) \qquad \text{nếu } i < 0$$

Chép tất cả các giá trị xuất của mạng vào giá trị mạng sẽ mở đầu cho sự quay lại mạng. Điều này làm phức tạp quá trình huấn luyện, bởi vì đầu ra của mạng không chính xác và khi quá trình huấn luyện bắt đầu và nó không thể quay trở lại trực tiếp, khi huấn luyện bắt đầu. Thay vì trọng số trung bình của kết suất thật sự và đích kết xuất được sử dụng. Khi bắt đầu huấn luyện, trọng số của đích sẽ cao. Nó sẽ giảm xuống 0 trong suốt qúa trình huấn luyện.

Mạng được huấn luyện trên một tập dữ liệu đã được gán nhãn. Đích kích hoạt là 0 cho tất cả các đơn vị xuất, ngoại trừ đơn vị mà tương ứng với nhãn đúng, nên được gán bằng 1.

Kiến trúc mạng có và không có lớp ẩn đã được huấn luyện và kiểm tra. Nhìn chung, mạng MLP với lớp ẩn mạnh hơn các mạng khác, nhưng nó cũng cần được huấn luyện nhiều và có rủi ro khá cao.

Trong cả hai loại mạng, gán nhãn từ loại cho một từ được thực hiện bằng cách chép xác suất nhãn của từ hiện tại và lân cận của nó vào các đơn vị nhập, lan truyền sự kích hoạt thông qua mạng tới các đơn vị xuất và xác

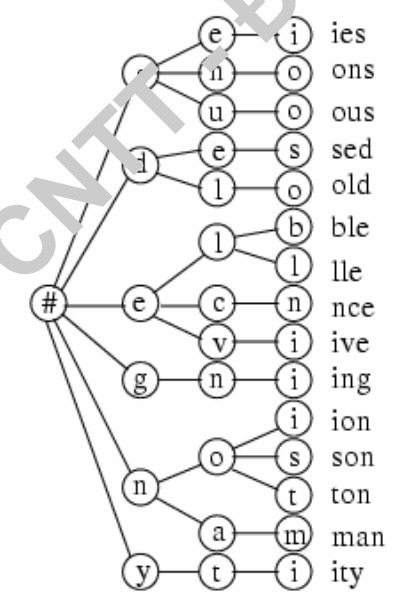
định đơn vị xuất mà có độ hoạt động cao nhất. Nhãn tương ứng đơn vị này sẽ được gán vào từ hiện tại.

2.2.2.4 Từ điển:

Từ điển chứa các xác suất nhãn ưu tiên của mỗi từ . Nó gồm 3 phần: một từ điển đầy đủ, một từ điển tiếp đầu ngữ, và một mục mặc định. Trong quá trình tra từ điển trong mạng gán nhãn, từ điển đầy đủ sẽ được tìm trước. Nếu từ được tìm thấy ở đây, xác suất nhãn tương ứng sẽ được trả về. Ngược lại, các ký tự hoa của từ sẽ được chuyển sang chữ thường và quá trình tìm kiếm trong từ điển đầy đủ sẽ được tiếp tục. Nếu lại thất bại, từ điển tiếp đầu ngữ sẽ được tìm kiếm tiếp theo. Nếu không có bước nào thành công, thì mục từ mặc định sẽ được trả về.

Từ điển đầy đủ được tạo từ tập dữ liệu huấn luyện đã được gán nhãn (khoảng 2 triệu từ trong Penn Treebank Corpus). Đầu tiên, số lần xuất hiện của mỗi từ/nhãn sẽ được đếm. Sau đó, các nhãn đối với mỗi từ sẽ được ước lượng xác suất.

Phần thứ hai của từ điển, từ điển tiếp đầu ngữ , tạo nên một cây.



Hinh 2-5: Cây từ điển trong mô hình mạng.

Mỗi nốt của cây(ngoại trừ các nút gốc) có nhãn là các ký tự. Tại các nốt lá, xác suất được gan vào. Trong quá trình tìm kiếm, cây tìm kiếm sẽ được tìm từ nốt gốc. Cây tiếp đầu ngữ được xây dựng trên dữ liệu huấn luyện. Đầu tiên, cây tiếp đầu ngữ được xây dựng từ các tiếp đầu ngữ có chiều dài 5 ký tự với các từ có từ loại mở. Sau đó, độ thường xuyên được đếm cho tất cả các tiếp đầu ngữ và lưu giữ tất cả các nốt tương ứng.

Các mục từ mặc định được tạo bằng cách loại bỏ các nhãn thường gặp của tất cả các của cây tiếp đầu ngữ đã được chặt.

2.2.3 Mô hình Maximum Entropy (ME):

2.2.3.1 Giới thiệu:

Mô hình ME([7]) được định nghĩa thông qua tập hợp HxT trong đó H là tập các từ có thể và nhãn trong ngữ cảnh và T là tập hợp các nhãn cho phép. Mô hình xác suất là một "history" h kết hợp với nhãn t được định nghĩa như sau:

$$p(h,t) = \pi\mu\prod_{j=1}^{k}\alpha_j^{fi(h,t)}$$

Trong đó π là hằng số tiêu chuẩn, ($\mu, \alpha_1, \ldots \alpha_k,$) là các thong số rõ ràng của mô hình và ($f_1,..f_k$) là các đặc trưng trong đó $f_j(h,t) \in (0,1)$ chú ý mối thông số α_i tương ứng với các nhãn ti và một dãy các từ($t_1,..t_k$) thuộc dữ liệu huấn luyện, thì hi là một history có sẵn đối với các nhãn t_i trước. Thống số ($\mu, \alpha_1, \ldots \alpha_k,$) được chọn sau đó để cực đại hoá lân cận của dữ liệu huấn luyện P:

$$L(p) = \prod_{i=1}^{n} p(h_i, t_i) = \prod_{i=1}^{n} \mu\pi \prod_{j=1}^{k} \alpha_j^{f_j(h_i,t_i)}$$

Ở đây, entropy của phân phối p được định nghĩa như sau:

$$H(p) = -\sum_{h \in H, t \in T} p(h,t) \log p(h,t)$$

Và các ràng buộc được định nghĩa:

$$Ef_i = \tilde{E} f_i$$

Trong đó các đặc trưng kỳ vọng của mô hình là:

$$Ef_i = \sum_{h \in H, t \in T} p(h,t) f_j(h,t)$$

Các đặc trưng giám sát là:

$$\tilde{E} f_i = \sum_{h \in H, t \in T} \tilde{p}(h,t) f_j(h,t)$$

Trong đó $\tilde{p}(h_i, t_i)$ biểu hiện các xác suất quan sát của (h_i,t_i) trong ngữ liệu huấn luyện. Như vậy sự ràng buộc đối với mô hình là phải kết hợp các ràng buộc kỳ vọng và ràng buộc giám sát trong dữ liệu huấn luyện . Trong thực tế h rất lớn và Ef_i không thể tính toán trực tiếp do đó xấp xỉ sau đây được sử dụng:

$$Ef_i \approx \sum_{i=1}^{n} \tilde{p}(h_i) p(t_i \mid h_i) f_j(h_i, t_i)$$

Trong đó $\tilde{p}(h_i, t_i)$ là xác suất giám sát của history h trong tập huấn luyện.

2.2.3.2 Các đặc trưng của gán nhãn từ loại:

Xác suất kết hợp của history h và nhãn t được xác định bởi các thông số đặc trưng lưu động, như là những α i sao cho $f_j(h,t) = 1$. Một đặc trưng có bởi (h,t), có thể tác động vào bất cứ từ nào hoặc nhãn nào của history h, và phải được mã hoá thành thông tin mà có thể giúp dự đoán t, như là vần của

từ hiện tại, xác định hai nhãn phía trước. Các từ và nhãn trọng một ngữ cảnh cụ thể có sẵn đối với một đặc trưng được cho bởi định nghĩa sau của history h_i :

$$h_i = \{w_1, w_{i+1}, w_{i+2}, w_{i-1}, w_{i-2}, t_{i-1}, t_{i-2} \}$$

Ví dụ như:

$$f_j(h_i, t_i) = \begin{cases} 1 & \text{Nếu suffix}(w_i)\text{="ing" \& } t_i\text{ =VBG} \\ 0 & \text{Nếu thuộc trường hợp khác} \end{cases}$$

Nếu như đặc trưng trên tồn tại trong tập đặc trưng của mô hình, các thông số tương ứng của mô hình sẽ đóng góp cho xác suất kết hợp $p(h_i,t_i)$ khi w_i kết thúc với "ING" và khi nhãn t_i =VBG. Nhờ vậy tham số α i của mô hình ảnh hưởng đối với các ngữ cảnh đoán trước chắc chắn, trong trường hợp tiếp vĩ ngữ "ING", đối với giám sát của một nhãn chắc chắn, trong trường hợp này là VBG.

Mô hình sẽ phát sinh không gian đặc trưng bằng cách kiểm tra mỗi cặp (h_i,t_i) trong dữ liệu huấn luyện với các đặc trưng mẫu cho bởi bảng 1. Với h_i như là history hiện tại, một đặc trưng luôn yêu cầu các câu trả lời Yes/No, và thêm vào đó là các ràng buộc chắc chắn giữa các nhãn chắc chắn. Ví dụ về các biến X,Y, và T trong bảng 1 chứa một số điều trong dữ liệu huấn luyện.

Điều kiện	Các đặc trưng
w_i không hiếm	w_i=X &t_i = T
w_i hiếm	X là tiếp đầu ngữ của w_i , |X|≤4 &t_i = T
	X là tiếp vĩ ngữ của w_i , |X|≤4 &t_i = T
	w_i chứa số & t_i = T
	w_i chứa chữa viết hoa & t_i = T
∀ w_i	t_{i-1} =X &t_i = T
	t_{i-1} t_i`=XY &t_i = T
	w_{i-}1 = X &t_i = T
	w_{i-2} = X &t_i = T
	w_{i+1} = X &t_i = T
	w_{i+1} = X &t_i = T

Bảng 1: Các đặc trưng của history h_i hiện tại.

Sự phát sinh các đặc trưng cho việc gán nhãn đối với các từ chưa biết dựa trên lý thuyết về sự phân biệt mà các từ hiếm trong dữ liệu huấn luyện tương tự đối với các từ chưa biết trong dữ liệu kiểm tra. Đặc trưng về các từ hiếm trong bảng 1, sẽ được áp dụng cho cả hai trường hợp từ hiếm và từ không biết trong dữ liệu kiểm tra.

Ví dụ như, bảng hai chứa một đoạn trích trong dữ liệu huấn luyện trong khi bảng 3 chứa các đặc trưng phát sinh trong khi kiểm tra (h3,t3), trong đó từ hiện tại là "about", và bảng 4 chứa các đặc trưng phát sinh trong khi kiểm tra (h4,t4), trong đó, từ hiện tại là "well-heeled", chỉ xuất hiệnt trong dữ liệu huấn luyện 3 lần nên được xem là từ hiếm.

Cách xử lý đối với các đặc trưng xuất hiện rất hiếm trong dữ liệu huấn luyện thường rất khó dự đoán, vì xác suất của nó rất khó tin cậy. Do đó, mô hình có sử dụng một heristic mà bất kỳ đặc trưng nào xuất hiện ít hơn mười lần trong dữ liệu huấn luyện thì không đáng tin cậy và bỏ qua các đặc trưng mà số lượng ít hơn 10.

Word	The	story	about	well-heeled	communities	and	developers.
Tag	DT	NNS	IN	JJ	NNS	CC	NNS
Pos	1	2	3	4	5	6	7

Bảng 2: Dữ liệu mẫu.

w_i = about	&t_i=IN
w_{i-1} = story	&t_i=IN
w_{i-2} = the	&t_i=IN
w_{i-2} = well-heeled	&t_i=IN
w_{i+2} = communities	&t_i=IN
t_{i-1} = NNS	&t_i=IN
$t_{i-2}t_{i-1}$ = DT NNS	&t_i=IN

Bảng 3: Các đặc trưng rút ra từ h_3 từ bảng 2

w_{i-1} = story	&t_i=JJ
w_{i-2} = the	&t_i=JJ
w_{i-2} = well-heeled	&t_i=JJ
w_{i+2} = communities	&t_i=JJ
t_{i-1} = NNS	&t_i=JJ
$t_{i-2}t_{i-1}$ = DT NNS	&t_i=JJ
prefix(w_i) = w	&t_i =JJ
prefix(w_i) = we	&t_i =JJ
prefix(w_i) = wel	&t_i =JJ
prefix(w_i) = well	&t_i =JJ
sufix(w_i) = d	&t_i =JJ
sufix(w_i) = ed	&t_i =JJ
sufix(w_i) = led	&t_i =JJ
sufix(w_i) = eled	&t_i =JJ

Bảng 4: Các đặc trưng phát sinh bởi h_4 rút bởi bảng 2.

2.2.3.3 Mô hình kiểm tra:

Mô hình kiểm tra yêu cầu một thuật toán tìm kiếm để liệt kê danh sách các nhãn ứng cử viên cho một câu và dãy nhãn có xác suất cao nhất được chọn làm câu trả lời.

Thuật toán tìm kiếm

Thuật toán tìm kiếm chủ yếu dựa trên thuật toán "tìm kiếm theo tia" sử dụng xác suất nhãn có điều kiện.

$$P(t|h) = \frac{p(h,t)}{\sum_{t' \in T} p(h,t')}$$

Với câu $\{w_1,..,w_n\}$, các nhãn ứng viên là $\{t_1,..,t_n\}$ thì xác suất điều kiện là:

$$p(t_1..t_n \mid w_1..w_n) = \prod_{i=1}^{n} p(t_i,h_i)$$

Thêm vào đó, thuật toán tìm kiếm còn tra cứu từ điển nhãn, mà đối với mỗi từ, danh sách các nhãn sẽ xuất hiện trong dữ liệu huấn luyện. Nếu từ điển nhãn có ảnh hưởng, thì thuật toán tìm kiếm, đối với mỗi từ chỉ phát sinh các nhãn có trong mục từ của từ điển, trong khi đối với các từ không biết thì phát sinh tất cả các nhãn có trong tập nhãn. Nếu không có từ điển nhãn thì thuật toán sẽ phát sinh tất cả các nhãn có trong tập nhãn.

Giả sử W=$\{w_1,...,w_n\}$ là một câu và xem s_{ij} là xác suất cao nhất thứ j và bao gồm cả từ w_i . Thuật toán được mô tả như sau:

Phát sinh nhãn cho w_1 , tìm giới hạn N, thiết lập giá trị cho s_j

$1 \le j \le N$.

Khởi tạo i=2

Khởi tạo j=1

Phát sinh nhãn cho w_i , với $s_{(i-1)j}$ là nhãn ngữ cảnh phía trước. Và thêm vào $s_{(i-1)j}$ tạo ra dãy mới.

j = j+1, lặp lại b nếu $j \le N$

Tìm N dãy xác xuất cao nhất được phát sinh bởi vòng lặp trên, và đặt s_{ij} $1 \leq j \leq N$ tương ứng.

i = i +1 , lặp lại a nếu $i \leq N$

Trả về xác suất cao nhất của dãy. S_{n-1}.

2.2.4 Mô hình TBL nhanh (FnTBL)

2.2.4.1 Giới thiệu giải thuật FnTBL:

Bên cạnh những ưu điểm của giải thuật TBL đã được trình bày ở trên thì TBL mắt phải một số khuyết điểm đó là kết quả học phụ thuộc nhiều vào kết quả gán nhãn cơ sở (số luật tăng theo số lỗi phát sinh trong quá trình gán nhãn cơ sở), ngữ liệu học phải lớn, đặc biệt là thời gian học của giải thuật TBL là khá lơn. Để khắc phục khuyết điểm này, có nhiều giải thuật cải tiến của giải thuật TBL đã được đưa ra như : LazyTBL([4]) , TBL xác suất, TBL đa chiều đặc biệt cải tiến đáng kể nhất là giải thuật Fast TBL (FnTBL).

Giải thuật FnTBL là giải thuật cải tiến của giải thuật TBL về mặt tốc độ. Giải thuật FnTBL có thời gian học ngắn hơn rất nhiều so với thời gian học của TBL, thời gian học bằng giải thuật FnTBL giảm so với thời gian học bằng giải thuật TBL từ 10 đến 130 lần, trong khi kết quả không bị ãnh hưởng. Giải thuật FnTBL được Radu Florian và Grace Ngai đưa ra vào năm 2001. Giải thuật FnTBL đã khắt phục triệt để khuyết điểm của TBL về thời gian huân luyện (nhất là huấn luyện trên ngữ liệu lớn). nguyên nhân chính làm cho thời gian huấn luyện của giải thuật TBL có thời gian huấn luyện quá lâu là do qua mỗi bước lập trong quá trình học, giải thuật TBL tiến hành thử tất của các luật ứng viên. Với mỗi luật ứng viên tác động lên ngữ liệu huấn luyện, giải thuật TBL tiến hành tính điểm cho luật ứng viên này bằng cách duyệt qua toàn bộ ngữ liệu huấn luyện để tìm ra các thay đổi trên ngữ liệu, điểm của luật là hiệu số của số thay đổi sai thành đúng và số thay đổi đúng thành sai. Với số luật ứng viên lớn và ngữ liệu lơn, việc duyệt qua toàn bộ dữ liệu khi tính điểm cho các luật ứng viên đã làm giải thuật TBL có thời gian học lớn. để khắt phục khuyết điểm về thời gian học của TBL, trước FnTBL

đã có môt số giải thuật được đề nghị như : TBL thống kê của Ramshaw và Marcus, ICA của Hepple, Lazy TBL của Samuel, các giải thuật này đã giảm được thời gian học nhưng nó ãnh hưởng đến độ chính xác hoặc có chi phí bộ nhớ quá lơn.

Thời gian học của TBL lâu là do việc tính điểm của mỗi luật ứng viên phải duyệt qua toàn bộ ngữ liệu học. để giảm thời gian học xuống, trong quá trình tính điểm cho mỗi luật ứng viên, FnTBL không tiến hành duyệt qua toàn bộ ngữ liệu học mà chỉ duyệt qua phần ngữ liệu bị thay đổi khi áp dụng luật ứng viên. mỗi luật ứng viên chỉ làm thay đổi một phần khá nhỏ trong ngữ liệu học nên thời gian tính điểm cho mỗi luật ứng viên giảm xuống đang kể, nó làm cho thời gian huân luyện của giải thuật giảm xuống đáng kể. Kết quả nhận được là giải thuật FnTBL làm giảm thời gian huấn luyện đi từ 10 đến 130 lần so với giải thuật TBL gốc, trong khi bộ nhớ tăng lên không đáng kể và không làm giảm độ chính xác.

Để dễ minh họa giải thuật, chúng sử dụng một số quy ước sau:

- C : tập các nhãn ngôn ngữ để gán cho các mẫu (có thể là từ loại, cú pháp, ngữ nghĩa, …)
- C[s] : chỉ sự gán nhãn cho mẫu (ví dụ gán từ loại cho từ).
- T[s] : chỉ sự gán nhãn đúng cho mẫu (ví dụ gán từ loại "VB" cho từ "go").
- p : vị từ được định nghĩa trên không gian S.
- Một luật r được định nghĩa như một cặp (p, t) gồm vị từ p và nhãn $t \in C$. Có nghĩa là mẫu $s \in S$ sẽ được gán nhãn t nếu vị từ p thoả trên s.
- R: Tập tất cả các luật.
- Nếu r=(p,t), thì lúc đó p_r sẽ chỉ thành phần p và t_r sẽ chỉ thành phần t trong luật r.

- Một luật $r=(p,t)$ được áp dụng trên mẫu s nếu mệnh đề $(p_r(s)=\text{True})\wedge(t_r\neq C[s])$ là Đúng. *r(s)* là kết quả áp dụng luật *r* trên nhãn *s*.
- Hàm đánh giá:
  - $f(r) = good(r) - bad(r)$ với:

good(r) = $\left|\{s \mid C[s] \neq T[s] \wedge C[r(s)] = T[s]\}\right|$

bad(r) = $\left|\{s \mid C[s] = T[s] \wedge C[r(s)] \neq T[s]\}\right|$

  - good(r): là số lượng những mẫu s mà được luật r sửa từ sai thành đúng.
  - bad(r) là số lượng những mẫu s mà bị luật r sửa từ đúng thành sai.

Trong tập luật ứng viên chúng ta chỉ quan tâm đến những luật nào sửa được ít nhất một lỗi f(b)≥0 và luật ứng viên nào có điểm cao nhất qua mỗi bước lập và có số điểm lớn hơn ngưỡng được giữ lại. cũng giống như giải thuật TBL, giải thuật FnTBL sẽ ngừng nếu như không cơ một luật tối ưu (luật ứng viên có điểm cao nhất trong mỗi bước lặp) lớn hơn ngưỡng.Giải thuật FnTBL khác với giải thuật TBL chủ yếu ở quá trình tính điểm cho luật ứng viên, do đó chúng tôi chỉ trình bày điểm khác biệt này.

2.2.4.2 Tính điểm và phát sinh luật:

Trong giải thuật FnTBL, thay vì phải phát sinh các luật ứng viên dựa trên các khung luật tại mỗi thời điểm, thì các luật ứng viên sẽ được phát sinh một lần và được giữ lại trong bộ nhớ. với mỗi luật ứng viên, giải thuật sẽ giữ kèm 2 giá trị good(r) và bad(r).

G(r) = $\{s \in S \mid (p_r(s) = true) \wedge (C[s] \neq t_r) \wedge (t_r = T[s])\}$: Tập các mẫu mà luật chuyển thành đúng, do đó good(r)=|G(r)|

B(r) = $\{s \in S \mid (p_r(s) = true) \wedge (C[s] \neq t_r) \wedge (C[s] = T[s])\}$: Tập các mẫu mà luật chuyển từ đúng thành sai; do đó bad(r)=|B(r)|

Khi một luật b mới học được tác động lên không gian mẫu S, chúng ta cần xác định được các luật r (đã được học trước đó) bị ảnh hưởng. Vì không phải toàn bộ ngữ liệu bị thay đổi nên sẽ có những luật r không bị tác động bởi luật b và chúng ta chỉ cần tính điểm lại cho các luật r nào bị luật b tác động.

Trong thực tế, khi luật b tác động lên mẫu s thì nó ảnh hưởng gián tiếp đến lân cận của s. Ta gọi vùng lân cận của một mẫu *s* này là $V(s)$. Nếu các mẫu độc lập với nhau, thì $V(s) = \{s\}$.

Khi một luật tối ưu *b* tác động lên mẫu $s \in S$ ($b(s) \neq C(s)$). Chúng ta cần xác định được những luật *r* nào chịu ảnh hưởng khi mẫu s thay đổi thành b(s). Chúng ta phải cập nhật f(r) nếu và chỉ nếu tồn tại ít nhất một mẫu s' thoả điều kiện sau:

$$+(s' \in G(r)) \wedge (b(s') \notin G(r))$$
$$+(s' \in B(r)) \wedge (b(s') \notin B(r))$$
$$+(s' \notin G(r)) \wedge (b(s') \in G(r))$$
$$+(s' \notin G(r)) \wedge (b(s') \in G(r))$$

Mỗi điều kiện trên đây tương ứng với số lần cập nhật cụ thể các giá trị good(r) hoặc bad(r). Khi luật b áp dụng lên mẫu s thì chỉ những mẫu thuộc tập V(s) mới bị ãnh hưởng, vì vậy chúng ta chỉ cần kiểm tra trên V(s).

Với $s' \in V(s)$ chúng ra cần phải xem xét hai trường hợp là luật b tác động lên s' và luật b không tác động lên s'.

- Trường hợp 1
  - **$C[s'] = C[b(s')]$ (b không ảnh hưởng tới s').**

Ta có điều kiện sau:

$$(s' \in G(r)) \wedge (b(s') \notin G(r))$$
$$\Leftrightarrow (p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s']) \wedge (p_r(b(s')) = false \quad \textbf{(5)}$$

bởi vì chúng ta có

$$(s' \in G(r)) \wedge (b(s') \notin G(r))$$

$\Leftrightarrow (p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s']) \wedge (p_r(b(s')) = false \vee C[b(s')] = t_r \vee t_r \neq T[b(s')])$

$\Leftrightarrow (p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s']) \wedge (p_r(b(s')) = false \vee C[s'] = t_r \vee t_r \neq T[s'])$

(vì $C[s'] = C[b(s')]$ và $T[s'] = T[b(s')]$)

$\Leftrightarrow p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s'] \wedge p_r(b(s')) = false$

bằng cách sử dụng luật DeMorgan và các điều kiện sau:

$[(C[s'] \neq t) \wedge (C[s'] = T[s'])] \wedge [(C[s'] = t) \vee (t_r \neq T[s'])] = false$

$\Leftrightarrow p_r(b(s')) = true \wedge C[s'] \neq t_r \wedge C[s'] = T[s'] \wedge p_r(s') = false$

Từ đó, một phương pháp được đề nghị để phát sinh luật r mà bị ảnh hưởng bởi sự tác động bởi luật b như sau:

- Tạo ra tất cả vị từ p (dựa vào các mẫu luật) thoả mẫu s'.
- **If** $C[s'] \neq T[s']$ **then**

  (a) If $p(b(s')) = false$ **then** giảm $good(r)$ trong đó $r = (p, T[s'])$.

- **else**

  (b) If $p(b(s')) = false$ **then** giảm $bad(r)$ với tất cả các luật r có vị từ là p và $t_r \neq C[s']$.

❖ Trường hợp 2

➢ $C[s'] \neq C[b(s')]$ (b có ảnh hưởng tới s')

Trong trường hợp này,

$p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s']) \wedge (p_r(b(s')) = false$

$\Leftrightarrow (p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s']) \wedge (p_r(b(s')) = false \vee t_r = C[b(s')])$

là do:

$\Leftrightarrow (p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s']) \wedge (p_r(b(s')) = false \vee C[b(s')] = t_r \vee t_r \neq T[b(s')])$

(do $T[s'] = T[b(s')]$)

$\Leftrightarrow p_r(s') = true \wedge C[s'] \neq t_r \wedge t_r = T[s'] \wedge (p_r(b(s')) = false \vee C[b(s')] = t_r)$

Thuật toán được sửa đổi bằng việc thay thế kiểm tra p(b(s'))=false với kiểm tra $p_r(b(s'))$=false V $C[b(s)]=t_r$ trong công thức (1) và bỏ đi các kiểm tra hoàn chỉnh cho trường hợp (2). Công thức được sử dụng để phát sinh luật r có

thể có số lần đếm tăng có dạng tương tự trong trường hợp 1 và 2 bằng cách chuyển đổi vai trò cảu s và b(s).

2.2.4.3 Giải thuật FnTBL:

Lặp với mỗi (s∈S| C[s]≠T[s])
phát sinh ∀ r | good(r)>0; tăng good(r).
Lặp với mỗi (s∈S| C[s]=T[s])
phát sinh ∀ p | (p(s)=true); Lặp với mỗi (r=(p,t)| (pr=p)∧ (tr≠C[s]) tăng bad(r).
1: Tìm luật b = argmaxr∈Rf(r)
Nếu (f(b)<ngưỡng học) ∨ (ngữ liệu học xong) thì chấm dứt.
Lặp với mỗi (p|R(p)={r|pr=p})
Lặp với mỗi (s∈S, s'∈V(s)| C[s]≠C[b(s)])
Nếu (C[s']=C[b(s')]) thì
Lặp với mỗi (p| p(s')=true)
Nếu (C[s']≠T[s']) thì
Nếu (p(b(s'))=false) thì giảm good(r) với r=(p,T[s']}
Ngược lại
Nếu (p(b(s'))=false) thì giảm bad(r) cho ∀r∈R(p)∧(tr≠C[s'])
Lặp với mỗi (p| p(b(s'))=true)
Nếu (C[b(s')]≠T[s']) thì
Nếu (p(s')=false) thì tăng good(r) với r=(p,T[s']}
Ngược lại
Nếu (p(s')=false) thì tăng bad(r) cho ∀r∈R(p)∧(tr≠C[s'])
Ngược lại
Lặp với mỗi (p| p(s')=true)
Nếu (C[s']≠T[s']) thì
Nếu (p(b(s'))=false)∨(C[b(s')=tr) thì giảm good(r) với r=(p,T[s'])
Ngược lại
giảm bad(r) cho ∀r∈R(p)∧(tr≠C[s'])
Lặp với mỗi (p| p(b(s'))=true)
Nếu (C[b(s')]≠T[s']) thì

Nếu (p(s')=false)∨(C[s']=tr) thì tăng good(r) với r=(p,T[s']}

Ngược lại

tăng bad(r) cho ∀r∈R(p)∧(tr≠C[b(s')])

Quay lại từ bước 1.

`

Chương 3

Mô hình

Trong chương này chúng tôi xin trình bày mô hình được dùng cho bài toán gán nhãn từ loại của mình. Đây là mô hình kết hợp bao gồm các mô hình gán nhãn được đánh giá là có độ chính xác nhất hiện nay. Bên cạnh đó, trong mô hình của mô hình của mình, chúng tôi có sử dụng thêm thông tin tiếng để cải tiến chất lượng của bộ gán nhãn.

3.1 Một số khái niệm sử dụng trong mô hình:

3.1.1 Ngữ liệu(Corpus):

Ngữ liệu là các nguồn dữ liệu được sử dụng cho các bài toán trong lĩnh vực xử lý ngôn ngữ tự nhiên. Ngữ liệu thường là tập hợp các câu dưới dạng tiếng nói hay văn bản, trong đó có chứa các thông tin cần thiết cho từng bài toán cụ thể trong xử lý ngôn ngữ tự nhiên. Các thông tin này được trích chọn sao cho phù hợp với các yêu cầu của bài toán.

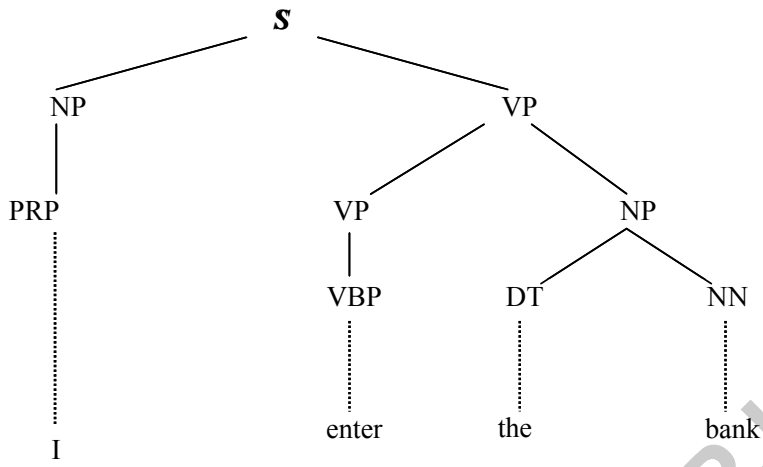
Ví dụ trong bài toán gán nhãn từ loại ngữ liệu có thể có dạng như sau :

- *The/DT woman/NN had/VBD nearly/RB died/VBN.*

  Trong đó "The/DT" cho biết từ The trong câu trên có nhãn từ loại là định từ (Determiner), "woman/NN" cho biết woman có nhãn từ loại là danh từ (Nuon), "had/VBD" cho biết had là động từ ở thì quá khứ (Verb)…

- *(S1 (S (NP (PRP I)) (VP (VBP enter) (NP (DT the) (NN bank)))))*

  Đây là một dạng cấu trúc dữ liệu của cây cú pháp.Trong đó các dấu ngoặc biểu diễn cho cấu trúc cây cú pháp. Cấy cú pháp được biểu diễn như trên trong ngữ liệu sẽ có dạng như sau :



Hình 3-1: Cây cú pháp trong ngữ liệu.

Các ngữ liệu trong đó không chứa các thông tin về ngôn ngữ được gọi là ngữ liệu thô (hay ngữ liệu trắng). Việc thêm thông tin vào ngữ liệu thô thường được làm bằng tay, đôi khi có sự hỗ trợ nhất định của phần mềm.. Có thể xem ngữ liệu như một cơ sở tri thức thô, trong đó, thông tin được thêm vào để chuẩn bị cho việc trích chọn tri thức về sau được dễ dàng hơn. Với nguồn ngữ liệu càng lớn thì việc trích chọn các tri thức về ngôn ngữ càng chính xác và đầy đủ hơn.

Để trích chọn thông tin về ngôn ngữ trên các nguồn ngữ liệu chúng ta thường dùng các giải thuật học. Các giải thuật học có thể sử dụng thông tin trong các ngữ liệu để rút ra (một cách tự động hay bán tự động) tập các luật cần thiết cho xử lý ngôn ngữ tự nhiên. Tập các luật này chính là cơ sở tri thức về ngôn ngữ có trong ngữ liệu đem huấn luyện.

Để trích chọn các tri thức về ngôn ngữ một cách chính xác, chúng ta cần có các ngữ liệu hoàn toàn chính xác, các ngữ liệu như thế được gọi là ngữ liệu vàng (golden corpus).

3.1.2 Ngữ liệu vàng(Golden Corpus)

Ngữ liệu vàng cũng là một dạng ngữ liệu trong đó có chứa thông tin hoàn toàn chính xác.

Trong mô hình của bài toán gán nhãn từ loại mà luận văn này đề cập đến, ngữ liệu vàng chính là một tập hợp các câu tiếng Anh đã được gán nhãn từ loại hoàn toàn chính xác.

Để xây dựng một bộ ngữ liệu vàng, chúng ta cần tốn rất nhiều công sức và thời gian, nên các bộ ngữ liệu vàng thường có giá thành rất cao. Trong quá trình làm luận văn chúng tôi đã sử dụng các bộ ngữ liệu vàng nhỏ , miễn phí.

Một ví dụ mẫu về ngữ liệu vàng:

Từ	**Từ loại**
List	VB
The	DT
Four	CD
Parts	NNS
Of	IN
A	DT
Computer	NN
System	NN
.	.

Trong đó cột thứ nhất là từ trong câu, cột thứ 2 là từ loại chính xác của từ trong cột thứ nhất.

Trong luận văn này chúng tôi sử dụng ba bộ ngữ liệu đó là SUSANNE, Cadasa, và một phần ngữ liệu Penn Tree Bank với số lượng từ như sau :

Bộ ngữ liệu	Số lượng từ trong ngữ liệu
SUSANNE	138000 từ
Cadasa	88000 từ
Một phần ngữ liệu Penn Tree Bank	125000 từ

3.1.3 Ngữ liệu huấn luyện(Training corpus):

Ngữ liệu huấn luyện là ngữ liệu được tạo ra từ ngữ liệu vàng để chuẩn bị cho quá trình học. Ngữ liệu huấn luyện có thể là ngữ liệu vàng, cũng có thể chứa thêm một số thông tin khác để phù hợp với giải thuật học trên ngữ liệu này.

Trong luận văn, ngữ liệu huấn luyện dùng trong mô hình kết hợp (được trình bày trong phần sau) có định dạng như sau :

Từ	Nhãn cơ sở	Nhãn đúng
I	PRP	PRP
Can	MD	MD
Can	MD	VB
A	DT	DT
Can	MD	NN

Trong đó cột thứ nhất là các từ trong câu, cột thứ 2 là nhãn cơ sở _ nhãn có sở là nhãn từ loại được giải thuật Maximum Entropy gán cho từ trong cột thứ nhất _ cột thứ 3 là nhãn đúng của từ trong cột thứ nhất, nhãn đúng này được trích ra trong ngữ liệu vàng.

Ngữ liệu huấn luyện được sử dụng trong phương pháp kết hợp thông tin với tiếng Việt để tăng độ chính xác cho việc gán nhãn từ loại trên tiếng Anh có định dạng như sau

Từ tiếng Anh	**Từ tiếng Việt**	**Nhãn tiếng Việt**	**Nhãn cơ sở**	**Nhãn đúng**
I	Tôi	P	PRP	PRP
Can	Có thể	A	MD	MD
Can	Đóng	V	MD	VB
A	Một	N	DT	DT
Can	Cái hộp	N	MD	NN

Trong đó cột thứ nhất là từ trong câu tiếng Anh, cột thứ 2 là từ trong câu tiếng Việt được liên kết với từ trong câu tiếng Anh ở cột thứ nhất thông qua mối liên kết từ, cột thứ 3 là nhãn từ loại của từ tiếng Việt, nhãn từ loại này được chọn là một từ loại bất kì trong số các từ loại của từ tiếng Việt, cột thứ 4 là nhãn cơ sở, nhãn này là kết quả của việc gán nhãn trên mô hình kết hợp các bộ gán nhãn cho tiếng Anh (đơn ngữ). Và cột cuối cùng la nhãn đúng của từ tiếng Anh trong cột thứ nhất....

3.2 Một số mô hình kết hợp hiện nay:

Hiện nay, trong các hệ thống xử lý ngôn ngữ tự nhiên, chúng ta có thể tìm thấy nhiều mô hình xử lý, sử dụng các tri thức ngôn ngữ để dự đoán, mô tả hay giải quyết vấn đề trong các bài toán ngôn ngữ([13]). Việc xử lý ngôn ngữ tự nhiên trong thế giới thực đòi hỏi chúng ta phải xem xét các khía cạnh của ngôn ngữ một cách toàn diện, nhưng các hệ thống xử lý ngôn ngữ thường chỉ sử dụng một phần thông tin hữu hạn, chúng thường phát sinh lỗi khi chúng ta thử nghiệm trên các ngữ liệu mới. Để chỉnh sửa cho các lỗi phát sinh này, một phương pháp thường được sử dụng đó là cố gắng mô tả các lỗi

sai và đưa ra các tri thức ngôn ngữ để chỉnh sửa các lỗi phát sinh, hay sử dụng thêm ngữ liệu để huấn luyện với hy vọng có thể rút ra các tri thức về ngôn ngữ bao quát cho bài toán. Với nguồn ngữ liệu hạn chế thì phương pháp trên không khả thi.

Một phương pháp mà chúng ta có thể quan tâm ở đây là phương pháp kết hợp các hệ thống lại với nhau. Vì với các hệ thống xử lý ngôn ngữ tự nhiên khác nhau thì chúng có các mô hình, hình thức xử lý khác nhau, chúng bao hàm các tri thức về ngôn ngữ khác nhau, vì vậy mà lỗi phát sinh trên các hệ thống cũng khác nhau. Việc kết hợp các hệ thống khác nhau sẽ giúp chúng ta có thể loại bỏ một số lỗi đáng kể.

Trong luận văn của mình, để có thể nâng cao độ chính xác cho bài toán gán nhãn từ loại, chúng tôi đã thử nghiệm mô hình kết hợp các bộ gán nhãn từ loại hiện nay lại với nhau.

3.2.1 *Mô hình kết hợp sử dụng nhiều mô hình liên kết*

Hiện nay, có khá nhiều mô hình có thể áp dụng cho việc gán nhãn từ loại như dùng xác suất thống kê, MAXIMUM ENTROPY(ME)[3], học hướng lỗi… . Tuy nhiên độ chính xác của các phương pháp này chỉ dừng lại khoảng 96%. Do đó, có khá nhiều phương pháp cải tiến được đưa ra nhằm làm tăng độ chính xác.

Do mỗi mô hình đều có những ưu điểm riêng nên có một cách tiếp cận được đưa ra đó là phối hợp các mô hình lại với nhau. Mô hình kết hợp này sẽ tận dụng các ưu điểm của các mô hình khác nhau. Trong quá trình gán nhãn từ loại, tuỳ theo trường hợp mà mô hình sẽ quyết định nhãn được lấy từ mô hình nào.

Như trong mô hình kết hợp giữa gán nhãn từ loại bằng ME và thống kê chẳng hạn. Giả sử câu được gán nhãn do hai mô hình đánh ra có sự khác nhau.

[3] Giải thuật này đã được trình bày cụ thể ở chương 2_Cơ sở lý thuyết

Ví dụ như câu "I go to school", kết quả gán nhãn từ loại của giải thuật ME là:

I/PRP go/VBP to/TO school/VB

Và kết quả gán nhãn từ loại dựa trên hướng tiếp cận thống kê như sau:

I/PRP go/VBP to/TO school/NN

Thì mô hình chính có nhiệm vụ quyết định chọn nhãn của từ "school" là của mô hình nào do ở đây trong hai mô hình có thể sẽ có một nhãn đúng. Ở trường hợp này mô hình thống kê đánh đúng. Do mỗi mô hình có một ưu điểm khác nhau mà ở mỗi trường hợp riêng, tỉ lệ chính xác của mỗi mô hình là khác nhau.

Chẳng hạn đối với mô hình thống kê, nếu các câu được gán nhãn từ loại có cùng phạm vi với dữ liệu được huấn luyện thì tỉ lệ chính xác sẽ rất cao. Nhưng đối với các trường hợp mà các câu không nằm trong dữ liệu huấn luyện hoặc đối với các từ chưa biết hoặc không có trong dữ liệu huấn luyện thì mô hình ME tỏ ra chính xác hơn. Chính vì vậy, mô hình tổng hợp phải biết chọn mô hình nào khi kết quả khác nhau.

Ở đây, việc chọn kết quả nào là hết sức khó khăn. Do đó, tuy kết quả của mô hình kết hợp có tăng nhưng vẫn còn khá hạn chế.

3.2.2 Phương pháp kết hợp dựa trên tính điểm cho các nhãn ứng viên

Đây là phương pháp kết hợp đơn giản nhất. Trong phương pháp này, các giải thuật gán nhãn tốt nhất hiện nay sẽ được chọn ra để tiến hành gán nhãn ban đầu cho ngữ liệu cần gán nhãn từ loại. Dựa trên danh sách các nhãn ban đầu này chúng ta sẽ tiến hành tính điểm cho từng nhãn tù loại. Các nhãn từ loại nào có điểm cao nhất sẽ được chọn làm nhãn chính xác cho mô hình. điểm của từng nhãn từ loại sẽ được tính theo công thức sau :

$$P(w_i,t_j) = \sum_{k=1}^{n} Out_k(w_i,t_j)$$

Trong đó :

+ w_i là từ thứ i trong ngữ liệu

+ t_j là nhãn thứ j trong tập nhãn có thể có của từ w_i

+ $P(w_i,t_j)$ là số bộ gán nhãn từ loại gán nhãn t_j cho từ w_i trong ngữ liệu.

+ $Out_k(w_i,t_j)$ là số lần bộ gán nhãn thứ k gán cho từ w_i nhãn t_j

$$Out_k(w_i,t_j) = \begin{cases} 1 & \text{Nếu từ } w_i \text{ có nhãn là } t_j \\ 0 & \text{Ngược lại} \end{cases}$$

Để thử nghiệm cho phương pháp này chúng tôi đã sử dụng ba bộ gán nhãn tốt nhất hiện nay là Unigram, Maximum Entropy và TBL.

Ví dụ để gán nhãn từ loại cho câu "I go to school" kết quả đầu ra của các bộ gán nhãn từ loại là :

Kết quả gán nhãn của Unigram là :

I/PRP go/VBP to/TO school/VB

Kết quả gán nhãn của Maximum Entropy là :

I/PRP go/VBP to/TO school/NN

Kết quả gán nhãn của TBL là :

I/PRP go/VBP to/TO school/VB

để chọn nhãn đúng cho từ "school" chúng ta tính điểm cho từ này là

P(school,VB) = 1 + 0 + 1 = 2

P(school,NN) = 0 + 1 + 0 = 1

Vậy điểm của trường hợp từ "school" trong câu trên có nhãn là VB là cao nhất vậy chúng ta chọn nhãn VB cho từ "school".

Vấn đề nảy sinh cho phương pháp này là trường hợp có từ hai nhãn trở lên có cùng số điểm và số điểm này là số điểm cao nhất như vậy câu hỏi đặt ra là chúng ta chọn nhãn nào là nhãn cho mô hình? Đối với vấn đề này

chúng tôi có đưa ra một heuristic là nếu có nhiều nhãn cùng số điểm, chúng ta sẽ nhân thêm một trọng số cho mỗi đầu ra của các giải thuật được chọn. Trọng số này do chúng ta đặt ra dựa trên độ chính xác của mỗi giải thuật, khi đó công thức tính điểm của sẽ là :

$$P(w_i,t_j) = \sum_{k=1}^{n} Out_k(w_i,t_j)*\beta_k$$

Trong đó β_k là trọng số dùng để nhân cho giải thuật thứ k.

Ví dụ lấy lại ví dụ ơ trên với trọng số cho giải thuật cho Unigram là 0.5 của TBL là 1 và của Maximum Entropy là 2 thì chúng ta có được điểm như sau :

P(school,VB) = 1*0.5 + 0*2 + 1*1 = 1.5

P(school,NN) = 0*0.5 + 1*2 + 0*1 = 2

Vậy nhãn được chọn là NN chứ không phải là VB như ở trên.

Một trong những nhược điểm của phương pháp này là nếu một nhãn nào đó có số phiếu bầu cao nhưng lại là nhãn sai trong khi các nhãn khác có số phiếu bầu thấp hơn lại là nhãn đúng thì việc chọn nhãn cho mô hình sẽ bị sai.

3.2.3 Phương pháp kết hợp dựa trên gợi ý của ngữ cảnh.

Việc kết hợp các giải thuật như trên sẽ gặp khó khăn trong trường hợp nếu có nhiều nhãn có cùng số điểm. Mặc dù đã dùng thêm các trong số vào việc tính điểm nhưng vấn đề vẫn chưa giải quyết hoàn toàn. Trong trường hợp có nhiều nhãn có cùng số điểm và các nhãn đúng lại là kết quả của các bộ gán nhãn có trọng số thấp thì rõ ràng chúng sẽ chọ kết quả sai. Để tránh những lỗi này chúng tôi đã dùng thêm thông tin ngữ cảnh của từ được xem xét để chọn nhãn chính xác cho mô hình. Thông tin ngữ cảnh được chúng tôi sử dụng đó là nhãn của từ phía trước và phía sau của từ hiện tại đối với mỗi

bộ gán nhãn. Các thông tin ngữ cảnh áp dụng cho việc kết hợp các bộ gán nhãn Unigram, Trigram, Maximum Entropy và TBL như sau :

W_{i-1}	W_i	W_{i+1}
Unigram_Tag_{i-1}	Unigram_Tag_i	Unigram_Tag_{i+1}
Trigram_Tag_{i-1}	Trigram_Tag_i	Trigram_Tag_{i+1}
TBL_Tag_{i-1}	TBL_Tag_i	TBL_Tag_{i+1}
MaxEnt_Tag_{i-1}	MaxEnt_Tag_i	MaxEnt_Tag_{i+1}

Mỗi ngữ cảnh xuất hiện trong ngữ liệu huấn luyện sẽ giúp cho chúng ta chọn được nhãn đúng cho từ. Xác suất mà nhãn xuất hiện trong ngữ cảnh đó sẽ được lưu lại, trong quá trình gán nhãn cho ngữ liệu mới nó sẽ giúp cho chúng ta chọn được nhãn chính xác cho mô hình.

Qua một thời gian thử nghiệm các phương pháp kết hợp chúng tôi đã chọn được một phương pháp kết hợp cho mô hình của mình, đó là sử dụng tính kế thừa của giải thuật TBL để kết hợp với giải thuật khác nhằm khử nhập nhằng trên cả hai phương .

3.2.4 Phương pháp kết hợp dựa trên tính kế thừa kết quả của giải thuật TBL

Trong phương pháp này chúng tôi kết hợp hai giải thuật đó là TBL và Maximum Entropy, đây là hai giải thuật được xem là một trong những giải thuật cho kết quả khả quan nhất. Chúng tôi dựa trên đặc điểm hai giải thuật này có cách sử dụng ngữ cảnh khác nhau trong việc chọn từ loại cho từ để kết hợp. Giải thuật Maximum Entropy chọn thông tin ngữ cảnh là năm từ chung quanh từ hiện tại (hai từ phía trước, hai phía sau và từ hiện tại) và nhãn của hai từ phía trước còn TBL thì chọn ngữ cảnh phục thuộc vào các mẫu luật do chúng ta đưa ra. Chính nhờ sự linh động này của TBL mà chúng

ta có thể chọn các ngữ cảnh cho TBL sao cho chỉnh được các trường hợp gây nhập nhằng trong giải thuật Maximum Entropy.

Một đặc điểm khác đã khiến chúng tôi chọn phương pháp này là tính kế thừa của TBL. TBL có thể kế thừa kết quả của các bộ gán nhãn khác. Giải thuật TBL có thể gán nhãn cho một ngữ liệu không phải là ngữ liệu thô mà đã được gán nhãn cơ sở bởi một mô hình khác.Việc dùng TBL để gán nhãn cho ngữ liệu đã được gán nhãn từ trước bằng một bộ gán nhãn khác sẽ làm cho chất lượng của bộ gán nhãn tăng lên. Chúng ta có thể thấy ngay việc gán nhãn cơ sở cao thì việc dùng các luật của TBL để chỉnh sẽ làm cho kết quả cao hơn. Mặt khác, TBL dùng luật để sửa các lỗi sai nên sẽ chỉnh được các lỗi sai do bộ gán nhãn ban đầu tạo ra. Như vậy việc kết hợp hai mô hình này sẽ tạo ra một mô hình mới có tình khả thi và chất lượng cao hơn.

3.3 Mô hình gán nhãn từ loại dựa trên song ngữ Anh-Việt

Mặc dù mô hình trên tương đối khả thi và cho kết quả tương đối cao, nhưng hạn chế lớn nhất của mô hình trên chính là tốc độ. Thời gian huấn luyện của mô hình TBL khá lâu, đặc biệt mỗi khi chúng ta đổi dữ liệu huấn luyện. Do đó, trong luận văn này chúng tôi quyết định một mô hình kết hợp khác,cũng tương tự như mô hình trên, nhưng chúng tôi sẽ sử dụng mô hình FnTBL[4] thay thế cho TBL. Sự thay thế này đã khắc phục được nhược điểm về mặt tốc độ của mô hình. Đồng thời chúng tôi còn tích hợp vào bộ gán nhãn từ loại dựa trên thông kê để cải tiến chất lượng cho quá trình gán nhãn cơ sở. Trong mô hình này của chúng tôi, FnTBL đóng trò là mô hình chính trong quá trình gán nhãn, hai mô hình kia đóng vai trò khởi tạo cho mô hình này.

Đối với mô hình FnTBL, đây là mô hình tương tự với mô hình TBL. Như đã trình bày ở trên, hai mô hình này đều có ưu điểm là dễ dàng kiểm soát và

[4] Giới thiệu về mô hình này đã được trình bày ở chương 2.

cải tiến. Trong quá trình gán nhãn ta có thể kiểm tra được lỗi phát sinh từ đâu và có khả năng giải quyết được vấn đề. Tuy nhiên, khó khăn đối với cả hai mô hình này chính là tập dữ liệu huấn luyện. Tập dữ liệu huấn luyện càng tốt thì bộ luật phát sinh ra sẽ hiệu quả hơn. Nhưng để có được một kho dữ liệu lớn là một điều hết sức khó khăn. Bên cạnh đó, do đây là mô hình học hướng lỗi nên càng về sau thì khả năng sửa lỗi của mô hình ngày càng bị bảo hòa. Khi tới một ngưỡng nào đó thì khả năng sửa được lỗi của chúng gần như tiến về 0 .

Do đó, trong luận văn này ,chúng tôi có đưa ra một số cải tiến bằng cách sử dụng thêm thông tin tiếng Việt. Thông tin này được rút ra từ từ điển tiếng Việt và trên ngữ song ngữ Anh-Việt đã được liên kết từ.

Chúng tôi đã tiến hành rút trích thông tin trên hai ngôn ngữ để làm thông tin khử nhập nhằng trong việc chọn từ loại cho bài toán gán nhãn từ loại. Mặc dù lượng thông tin là rất lớn nhưng chúng ta làm sao để nhận ra đâu là thông tin cần thiết cho việc chọn từ loại là một vấn đề khó khăn.

Về mặt từ loại, đối với hai ngôn ngữ khác nhau về loại hình như tiếng Việt và tiếng Anh thì từ loại của cùng một từ là khác nhau. Từ trên ngôn ngữ này có từ loại là X nhưng khi nó được dịch ra trên ngôn ngữ khác có thể có từ loại khác. Mặt khác từ trên ngôn ngữ này có thể có nhiều từ loại nhưng trên ngôn ngữ khác nó chỉ có một từ loại. Ví dụ như từ “can” trong tiếng Anh có nhiều từ loại (có thể có từ loại là Modal, Verb, Noun) còn từ “có thể”, từ “đóng”, hay từ “cái hộp” trong tiếng Việt chỉ có một từ loại. Khi từ “can” liên kết với từ một trong ba từ “có thể”, “đóng”, hay “cái hộp” thì từ loại của từ “can” có thể được xác định chính xác, không bị nhập nhằng nữa.

Trong bài toán gán nhãn từ loại của mình chúng tôi đã tiến hành học trên ngữ liệu song ngữ(bằng mô hình FnTBL) để tìm ra các mối quan hệ giữa từ và từ loại trên hai ngôn ngữ là tiếng Anh và tiếng Việt. Những mối liên hệ này chính là cơ sở cho việc chọn từ loại cho từ trong ngữ liệu cần gán nhãn. Vì không giống như bài toán gán nhãn từ loại trước đây chỉ làm trên ngữ liệu

đơn ngữ, chúng tôi phải tiến hành xây dựng lại các khung luật cho việc học. Trong quá trình học bằng phương pháp FnTBL, các khung luật(hay còn gọi là template[5])góp phần rất quan trọng cho độ chính xác của giải thuật.

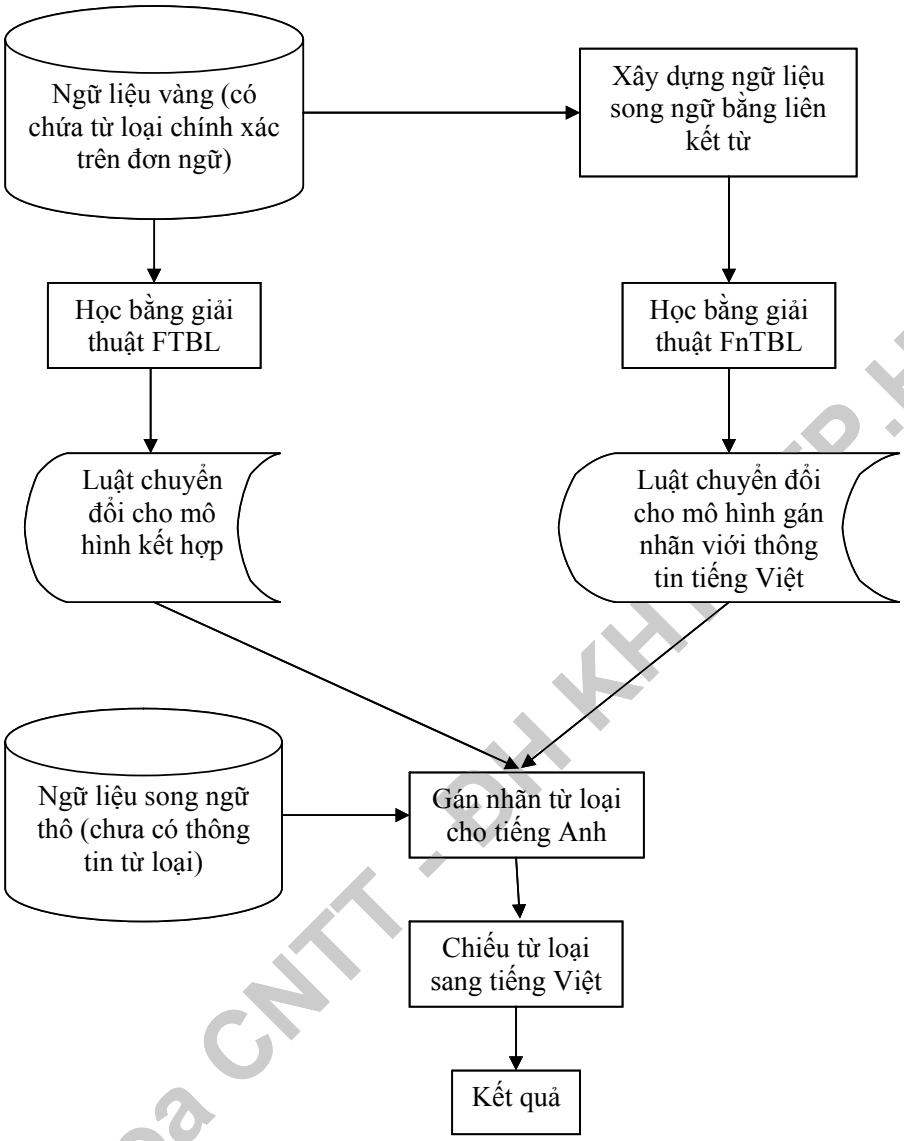
Các thông tin mà chúng tôi dùng để khử nhập nhằng ở đây là từ và từ loại của từ tiếng Việt liên kết với từ tiếng Anh đang xét, các từ và từ loại của từ tiếng Việt đứng trước và sau từ tiếng Việt liên kết với từ tiếng Anh đang xét.

Việc sử dụng thông tin tiếng Việt để khử nhập nhằng cho việc gán nhãn từ loại trên tiếng Anh đã làm cho kết quả của gán nhãn tăng lên rõ rệt. Chúng tôi đã tiến hành học trên 150000 từ trong ngữ liệu SUSANNE và đánh giá trên 20000 từ còn lại của ngữ liệu SUSANNE kết quả nhận được khi đánh giá trên 20000 từ này là 98,5% nhãn được gán chính xác so với kết quả nhận được khi đánh giá trên cùng ngữ liệu này của giải thuật TBL là 96,4% và Maximum Entropy là 96,6%.

Một phần khác của mô hình này là việc sử dụng các thông tin về nhãn đã có bên tiếng Anh kết hợp với các thông tin tiếng Việt để ánh xạ từ loại qua tiếng Việt. Nhờ đó, ta có thể xây dựng một bộ ngữ liệu về từ loại cho tiếng Việt. Đó sẽ là một ngữ liệu hết sức quý báu.

[5] Phần này sẽ được trình bày cụ thể hơn trong phần mô hình

3.3.1 Sơ đồ hoạt động của mô hình:



Hình 3-2: Sơ đồ hoạt động của mô hình gán nhãn từ loại trên ngữ liệu song ngữ Anh-Việt.

Trên đây chính là mô hình hoạt động của mô hình. Mô hình này được hoạt động dựa trên mô hình chính là mô hình FnTBL nên việc chuẩn bị một

dữ liệu học cho chương trình là hết sức cần thiết. Tập dữ liệu học này sẽ ảnh hưởng rất nhiều đến kết quả của chương trình.

3.3.1.1 Ngữ liệu huấn luyện:

Do các luật học sẽ được rút ra từ ngữ liệu nên các dữ liệu trong ngữ liệu phải đảm bảo độ chính xác. Hiện nay, để tìm được những nguồn dữ liệu lớn để thực hiện việc huấn luyện là hết sức khó khăn. Do đó, trong luận văn này chúng tôi chỉ sử dụng một ngữ liệu nhỏ, miễn phí có dữ liệu chính xác, đó là ngữ liệu SUSANNE(khoảng 138000 từ). Các thông tin về nhãn từ loại sẽ được rút ra từ trong ngữ liệu này. Trong ngữ liệu này, dữ liệu là những câu tiếng Anh thuộc nhiều lĩnh vực đã được gán nhãn từ loại. Như ví dụ sau, đây là một số câu đã được gán nhãn rút ra từ ngữ liệu SUSANNE:

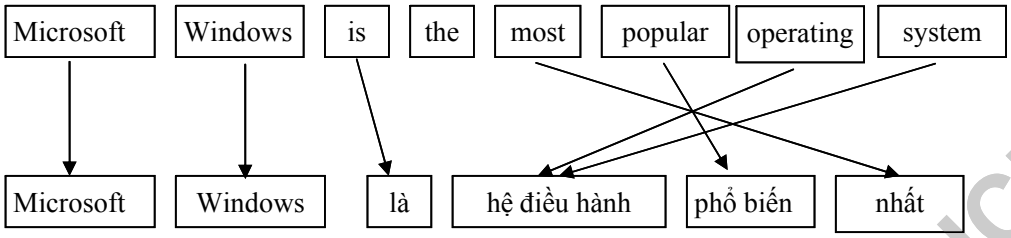
*The/**DT** Fulton/**NNP** County/**NN** Grand/**JJ** Jury/**NN** said/**VBD** Friday/**NN** an/**DT** investigation/**NN** of/**IN** Atlanta/**NNP** 's/**POS** recent/**JJ** primary/**JJ** election/**NN** produced/**VBD** "/`` no/**DT** evidence/**NN** "/" that/**IN** any/**DT** irregularities/**NNS** took/**VBD** place/**NN** ./.*

*The/**DT** jury/**NN** further/**WRB** said/**VBD** in/**IN** term/**NN** -/: end/**NN** presentments/**NNS** that/**IN** the/**DT** City/**NN** Executive/**JJ** Committee/**NN** ,/, which/**WDT** had/**VBD** over-all/**JJ** charge/**NN** of/**IN** the/**DT** election/**NN** ,/, "/`` deserves/**VBZ** the/**DT** praise/**NN** and/**CC** thanks/**NNS** of/**IN** the/**DT** City/**NN** of/**IN** Atlanta/**NNP** "/" for/**IN** the/**DT** manner/**NN** in/**IN** which/**WDT** the/**DT** election/**NN** was/**VBD** conducted/**VBN** ./*

Trong mô hình này, để làm tăng thêm độ chính xác của bộ gán nhãn, chúng tôi đã sử dụng thêm một số thông tin của tiếng Việt để cải tiến thêm chất lượng của mô hình. Do đó, trong ngữ liệu học còn có thêm các thông tin tiếng Việt. Trong mô hình này chúng tôi chỉ sử dụng nghĩa và từ loại của tiếng Việt để làm thông tin bổ sung. Trong ngữ liệu, mỗi câu tiếng Anh sẽ có tương ứng một câu tiếng Việt. Đồng thời, các từ ở hai câu đều được liên kết với nhau.Ví dụ như sau:

Microsoft Windows is the most popular operating system.

Microsoft Windows là một hệ điều hành phổ biến nhất



Hình 3-3: Sơ đồ liên kết từ.

Sau đó, các câu trong ngữ liệu đã được gỡ nhãn sẽ được gán nhãn khởi tạo(hay còn gọi là quá trình gán nhãn ngây thơ) để tạo ra dữ liệu huấn luyện cho chương trình. Cuối cùng, ngữ liệu huấn luyện sẽ có cấu trúc như sau:

Từ gốc	*Nhãn khởi tạo*	*Nhãn đúng*	*Nghĩa tiếng Việt*
I	*PRP*	*PRP*	*Tôi*
want	*VBP*	*VBP*	*muốn*
To	*To*	*To*	*#*
Book	*NN*	*VB*	*đặt*
Two	*CD*	*CD*	*hai*
Books	*NNS*	*NNS*	*cuốn sách*
List	*VB*	*VB*	*Liệt kê*
Five	*CD*	*CD*	*năm*
Units	*NNS*	*NNS*	*đơn vị*
Of	*IN*	*IN*	*Về*
Measure	*NN*	*NN*	*độ đo*
For	*IN*	*IN*	*cho*
Computer	*NN*	*NN*	*máy tính*
Memory	*NN*	*NN*	*bộ nhớ*
And	*CC*	*CC*	*Và*

Storage	*NN*	*NN*	*lưu trữ*

Ở đây, ngữ liệu được lưu làm 4 trường: Từ gốc, nhãn khởi tạo, nhãn đúng và cuối cùng là nghĩa của từ. Mỗi từ sẽ nằm trên một hàng và các câu được cách nhau bằng một dòng trắng.

3.3.1.2 Quá trình khởi tạo:

Đối với mô hình FnTBL thì quá trình khởi tạo nhãn ban đầu khá quan trọng và sẽ ảnh hưởng phần nào đến kết quả của chương trình gán nhãn. Do đó, trong quá trình khởi tạo này, chúng tôi đã quyết định sử dụng một mô hình có độ chính xác tương đối cao là ME để gán nhãn khởi tạo cho các đơn vị ngôn ngữ ban đầu được sử dụng làm dữ liệu học.

Sau đó, ta tiếp tục sử dụng mô hình thống kê để sửa một số nhãn còn chưa đúng trong quá trình khởi tạo trước. Ở quá trình này, nhãn của các từ sẽ được lọc qua tập bộ nhãn cho phép đối với mỗi từ. Các nhãn không hợp lệ sẽ được loại bỏ và thay thế là nhãn có xác suất cao nhất. Đây là toàn bộ bước một của mô hình FnTBL_ khởi tạo. Ở quá trình này, các từ sẽ được gán các nhãn gần đúng nhất có thể.

Quá trình khởi tạo này được thực hiện đối với các câu trong ngữ liệu vàng đã được tách nhãn. Đây là quá trình chuẩn bị dữ liệu học cho mô hình FnTBL. Sau quá trình này là quá trình huấn luyện và rút luật của mô hình FnTBL.

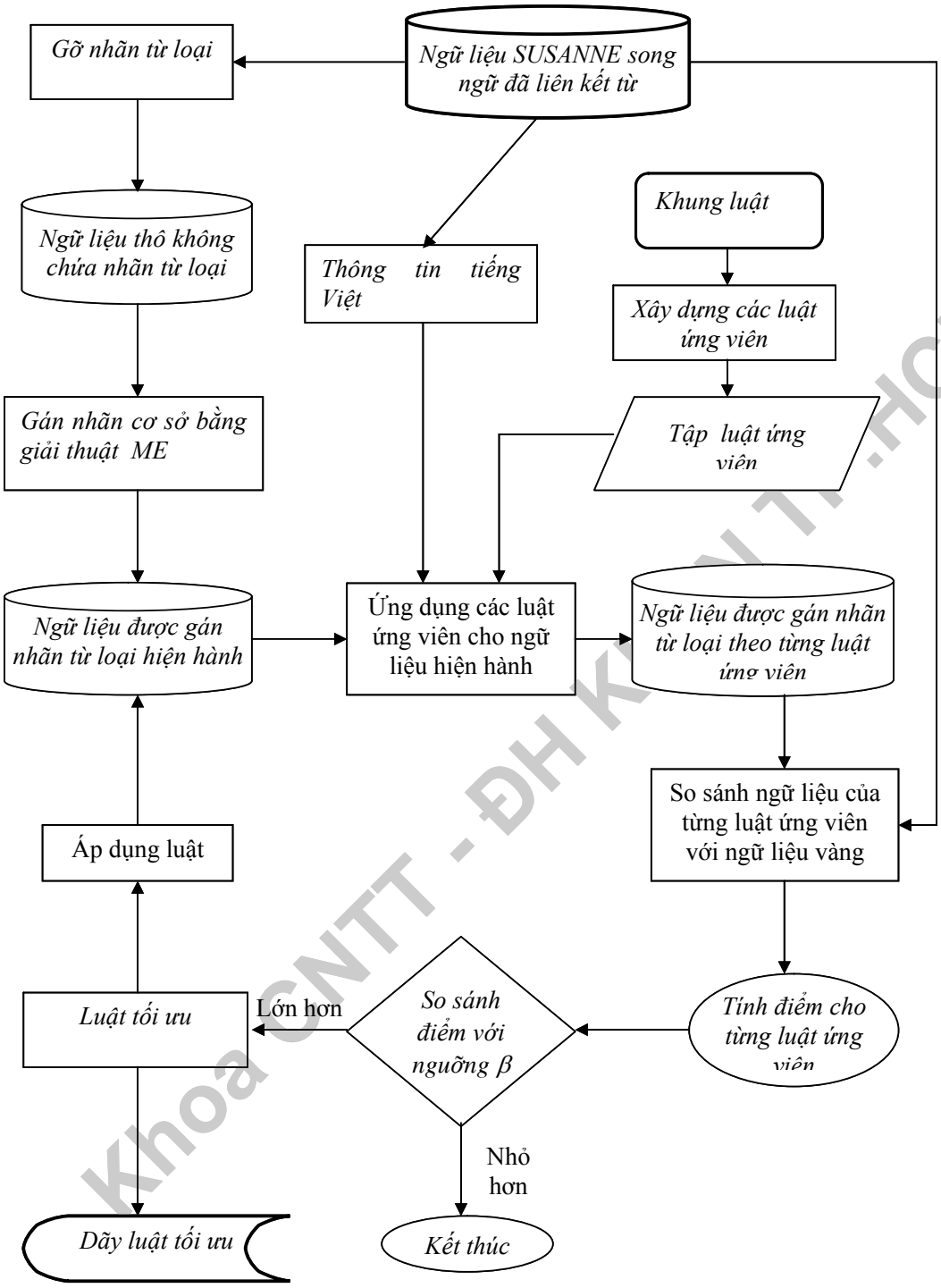
3.3.1.3 Quá trình huấn luyện:

Sau đó là quá trình huấn luyện của mô hình FnTBL. Quá trình này cũng tương tự quá trình huấn luyện của mô hình TBL.

Ngữ liệu học (ngữ liệu được tạo ta ở quá trình khởi tạo) sẽ được áp dụng lần lượt các luật ứng viên. Các luật ứng viên đều thuộc những dạng khung luật đã được định sẵn (template). Từ các khung luật này, các luật cụ thể sẽ được phát sinh và áp dụng thử lên ngữ liệu. Ngữ liệu này sẽ được so sánh với

ngữ liệu vàng để đánh giá số điểm cho luật vừa được áp dụng.(Chỉ tiêu tính điểm là hiệu số nhãn đúng/sai trước và sau khi áp dụng luật ứng viên). Quá trình như vậy tiếp tục được lặp lại và chỉ những luật có điểm cao nhất sau mỗi vòng lặp mới được giữ lại. Quá trình phát sinh luật ở đây hoàn toàn tương tự với thuật toán TBL(tham khao chương 2).

- Mô hình huấn luyện cho bộ gán nhãn từ loại tiếng Anh



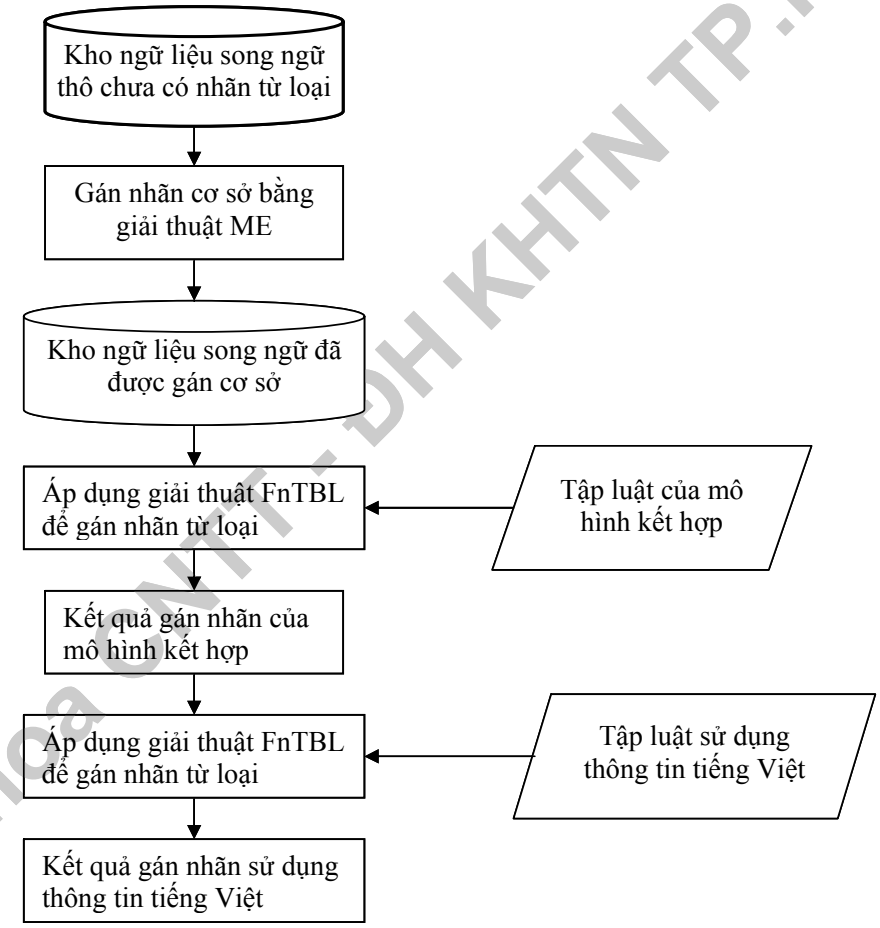
Hình 3-4: Mô hình huấn luyện cho nhãn tiếng Anh

3.3.1.4 Quá trình gán nhãn từ loại trên cặp câu song ngữ

Các luật tạo ra ở phần huấn luyện sẽ được áp dụng vào phần gán nhãn. Tương tự như quá trình huấn luyện, các câu đưa vào cần được gán nhãn khởi tạo. Sau đó, sẽ được sửa lỗi bằng các luật rút ra từ quá trình huấn luyện.

Khác với các mô hình khác, trong mô hình này có sử dụng thêm thông tin tiếng Việt. Các câu tiếng Anh đã được liên kết với tiếng Việt trong song ngữ Anh-Việt. Các thông tin tiếng Việt có được là nhờ các mối liên kết từ và từ loại tiếng Việt rút ra trong từ điển.

Mô hình gán nhãn cho tiếng Anh trong song ngữ Anh-Việt.



Hình 3-5: Mô hình gán nhãn cho tiếng Anh trong ngữ liệu song ngữ Anh-Việt

Khó khăn chính của bộ gán nhãn từ loại (POS-tagger) là phải giải quyết các trường hợp nhập nhằng từ loại. Nghĩa là một từ có thể có nhiều từ loại, nhưng trong một ngữ cảnh cụ thể, nó chỉ có thể có một từ loại đúng mà thôi. Ví dụ: trong câu "I can can a can", thì bộ gán nhãn từ loại phải gán được từ loại như sau: "I/PRP can/MD can/VB a/DT can/NN".

Mặc dù phương pháp FnTBL được sử dụng trong mô hình này tỏ ra khá hiệu quả và có nhiều ưu thế so với các phương pháp khác nhưng độ chính xác của phương pháp này chỉ đạt tới một ngưỡng mà thôi. Do đó, chúng tôi có sử dụng một số thông tin tiếng Việt để cải tiến chất lượng cho mô hình.

Cơ sở lý luận của mô hình giải quyết bài toán này chính là dựa trên sự khai thác thế mạnh của ngữ liệu song ngữ trong việc giúp nhau khử nhập nhằng. Vì các chương trình gán nhãn từ loại mạnh nhất của nước ngoài đã khai thác tối đa các thông tin có thể có trong câu tiếng Anh để gán nhãn từ loại, chính vì vậy,muốn tăng kết quả của bộ gán nhãn thì cần phải có các thông tin phụ. Khi câu tiếng Anh được liên kết với câu tiếng Việt trong song ngữ Anh-Việt, thì chúng ta có thêm nguồn thông tin mới vô cùng quí giá: đó là từ loại (lấy từ từ điển) của các từ tiếng Việt tương ứng đã được liên kết với các từ tiếng Anh đang cần khử nhập nhằng đó.

Chẳng hạn: từ "can" trong tiếng Anh có nhiều từ loại khác nhau: trợ động từ (có thể), động từ (đóng hộp), danh từ (cái hộp) và đến nay khó có bộ gán nhãn từ loại nào có thể gán từ loại chính xác cho từ "can" đó trong nhiều ngữ cảnh khác nhau. Nhưng một khi từ "can" này được liên kết với từ tiếng Việt tương ứng trong ngữ liệu song ngữ Anh-Việt, thì từ loại của nó lại được xác định một cách dễ dàng (ví dụ: từ "can" mà được liên kết với từ "có thể" thì chắc chắn từ loại của nó là trợ động từ, còn nếu nó được liên kết với từ "đóng hộp" thì chắc chắn từ loại của nó sẽ là "động từ",…).

Nguồn thông tin quí giá (bên ngoài câu tiếng Anh) này sẽ được giải thuật FnTBL đưa vào khung luật (template) bên cạnh các thông tin thông thường mà trước đó nhiều bộ gán nhãn tự loại tiếng Anh đã khai thác.

3.3.2 Thuật giải

Bài toán gán nhãn từ loại chủ yếu dựa trên giải thuật FnTBL. Giải thuật của mô hình gán nhãn từ loại dựa trên song ngữ Anh Việt như sau :

Đầu vào : ngữ liệu song ngữ, trong đó với một câu tiếng Anh sẽ có tương ứng một câu tiếng Việt, là câu dịch của câu tiếng Anh.

Đầu ra : tập nhãn từ loại kết quả của câu tiếng Anh trong ngữ liệu song ngữ.

Bước 1 : tiền xử lý ngữ liệu đầu vào. Câu tiếng Anh và tiếng Việt được tách từ trong bước tiền xử lý nay.

Bước 2 : tiến hành liên kết từ cho ngữ liệu song ngữ Anh-Việt . Mỗi từ tiếng Anh có thể liên kết một hay nhiều từ tiếng Việt. Trong ngữ liệu song ngữ được liên kết từ trong bước 2 này có thể tồn tại các từ tiếng Anh không liên kết với từ tiếng Việt. Các từ tiếng Anh không liên kết với từ tiếng Việt nào thường là hư từ.

Bước 3 : tiến hành gán nhãn cơ sở cho câu tiếng Anh. Câu tiếng Anh sẽ được gán nhãn cơ sở bằng giải thuật Maximum Entropy. Kết quả gán nhãn ở bước này sẽ được kiểm tra trước khi đưa vào làm nhãn cơ sở cho giải thuật FnTBL. Việc kiểm tra được thực hiện với một từ điển tiếng Anh trong đó có chứa thông tin từ loại. Kết quả gán nhãn trên sẽ kiểm tra xem nhãn từ loại của một từ trong ngữ liệu có trong tập nhãn có thể có của từ này hay không, nếu không có thì mô hình sẽ chọn một nhãn có xác suất xuất hiện cao nhất.

Bước 4: áp dụng luật chuyển đổi cho mô hình kết hợp vào ngữ liệu vừa gán nhãn cơ sở ở bước 3. Giải thuật FnTBL sẽ tiến hành áp dụng tất cả các luật trong tập luật đã học được vào ngữ liệu. Các luật có số điểm cao sẽ được áp dụng trước các luật có điểm thấp sẽ được áp dụng sau.

Bước 5: Áp dụng thông tin tiếng Việt vào việc chọn nhãn từ loại.

Chúng ta tiến hành tra từ điển tiếng Việt để nhận được các nhãn từ loại của tất cả các từ tiếng Việt, các nhãn này có thể không chính xác. Các nhãn từ loại và từ tiếng Việt sẽ được dùng làm thông tin ngữ cảnh để chọn nhãn cho

từ tiếng Anh. Giải thuật FnTBL sẽ tiến hành duyệt qua các luật chuyển đổi trong tập luật có chứa thông tin tiếng Việt, các luật có điểm cao sẽ được xét đến trước, các luật nào phù hợp với thông tin tiếng Việt trong ngữ liệu sẽ được áp dụng.

3.3.3 Khung luật (Template):

Một trong các yếu tố ảnh hưởng đến kết quả của bộ gán nhãn chính là khung luật. Khung luật là các mẫu luật học có sẵn do chúng ta đưa ra, chúng sẽ được sử dụng trong quá trình huấn luyện của chương trình. Mẫu luật học đưa ra sẽ ảnh hưởng rất nhiều đến khả năng sửa lỗi của chương trình. Các bộ khung luật thông thường có dạng là $\bigcup_{i\in[-a,b]} Word_i$ hoặc $\bigcup_{i\in[-a,b]} Tag_i$ hoặc $\bigcup_{i\in[-a,b]} Word_i \wedge Tag_i$, ngoài ra trong mô hình của chúng tôi còn có sử dụng thêm các thông tin tiếng Việt để bổ sung thông tin cho bộ gán nhãn. Do đó, có một số khung luật về tiếng Việt như sau: $\bigcup_{i\in[-a,b]} VNTag_i$. Trong đó, VNtag là nhãn từ loại của từ tiếng Việt tương ứng với từ tiếng Anh đang xét. Đây là điểm khác biệt của mô hình chúng tôi so với mô hình khác.

Dưới đây là một số khung luật mà chúng tôi đã sử dụng trong mô hình:

- Các khung luật tương đối cụ thể (dùng để sửa các trường hợp đặc biệt xuất hiện ít trong ngữ liệu huấn luyện).

  pos_0 word_0 word_1 word_2 => pos
  pos_0 word_-1 word_0 word_1 => pos
  pos_0 word_0 word_-1 => pos
  pos_0 word_0 word_1 => pos
  pos_0 word_0 word_2 => pos
  pos_0 word_0 word_-2 => pos
  pos_0 word:[1,2] => pos
  pos_0 word:[-2,-1] => pos
  …

- Các khung luật tổng quát

pos_0 pos_1 pos_-1 => pos

pos_0 pos:[1,3] => pos

pos_0 pos:[1,2] => pos

pos_0 pos:[-3,-1] => pos

pos_0 pos:[-2,-1] => pos

pos_1 word_0 word_-1 => pos

pos_0 pos_1 pos_2 => pos

pos_0 pos_1 pos_2 word_1 => pos

Ngoài ra, còn có một số khung luật áp dụng trên tiếng Việt như:

word_0 pos_0 posvn_0 wordvn_0 => pos

word_0 pos_0 posvn_1 wordvn_1 => pos

word_0 pos_0 posvn_2 wordvn_2 => pos

Dựa vào các khung luật trên, mô hình FnTBL sẽ phát sinh các luật để sửa các lỗi xuất hiện trong ngữ liệu huấn luyện. Mỗi luật được sinh ra đều có một số điểm nhất định. Dưới đây là một số luật học đã được mô hình rút ra.

- Các luật tổng quát

GOOD:808 BAD:0 SCORE:808 RULE: pos_0=AUX pos:[-3,-1]=NN => pos=VBZ

GOOD:459 BAD:0 SCORE:459 RULE: pos_0=AUX pos:[-3,-1]=NNS => pos=VBP

GOOD:184 BAD:0 SCORE:184 RULE: pos_0=AUX pos:[-3,-1]=PRP => pos=VBP

GOOD:169 BAD:15 SCORE:154 RULE: pos_0=VBZ pos_-1=NNS pos_-2=NN => pos=VBP

GOOD:149 BAD:12 SCORE:137 RULE: pos_0=VBP pos:[-3,-1]=MD => pos=VB

GOOD:120 BAD:0 SCORE:120 RULE: pos_0=AUX pos:[1,3]=DT => pos=VBZ

GOOD:123 BAD:3 SCORE:120 RULE: pos_0=VBZ pos:[-3,-1]=MD => pos=VB

GOOD:98 BAD:9 SCORE:89 RULE: pos_0=NNS pos_1=CD => pos=NNP

GOOD:81 BAD:5 SCORE:76 RULE: pos_0=NNP pos_-1=VB pos_1=CD => pos=NN

- Các luật cụ thể

GOOD:137 BAD:0 SCORE:137 RULE: pos_0=NN pos_1=NN word_0=operating word_1=system => pos=VBG

GOOD:31 BAD:3 SCORE:28 RULE: pos_0=NN word_0=data pos_1=NN => pos=NNS

GOOD:23 BAD:0 SCORE:23 RULE: pos_0=NNP word_0=PC pos_-1=JJ => pos=NN

GOOD:20 BAD:1 SCORE:19 RULE: pos_0=NNP word_0=matrix pos_1=NNS => pos=NN

GOOD:13 BAD:0 SCORE:13 RULE: pos_-1=RB pos_0=VBP word_-1=not word_0=need => pos=VB

GOOD:15 BAD:2 SCORE:13 RULE: pos_0=NNP word:[1,2]=printer => pos=NN

Mặc dù số luật số luật của mô hình phát sinh là khá lớn nhưng không phải tất cả các luật đều dùng được. Chúng phải được kiểm tra lại bằng tay để đảm bảo được độ chính xác của luật.(Tham khảo thêm ở phụ lục D)

3.3.4 Cải tiến

Như đã trình bày ở trên, các mô hình gán nhãn hiện nay trên thế giới đã tận dụng gần như triệt để các thông tin mà câu tiếng Anh cung cấp. Do đó, muốn tăng kết quả của bộ gán nhãn thì ta phải sử dụng thêm các thông tin lấy từ ngôn ngữ khác. Do đó, khi tiến hành gán nhãn từ loại trên ngữ liệu song ngữ Anh-Việt, ta sẽ có được một nguồn thông tin bổ sung vô cùng phong phú.Với các thông tin này, chúng ta sẽ giải quyết được khá nhiều trường hợp nhập nhằng. Như ví dụ sau:

Ta có cặp câu sau:

*Even if a computer can do its job without a person sitting in front of it, people still design, build, **program**, and repair computer systems.*

*Thậm chí một máy tính có thể làm các công việc của nó mà không cần người ngồi trước mặt, nhưng con người vẫn thiết kế, xây dựng, **lập trình** và chuẩn bị các hệ thống máy tính đấy chứ.*

*A computer file is simply a set of data or **program** instructions that has been given a name.*

*Một tập tin máy tính chỉ đơn giản là một tập hợp dữ liệu hoặc các chỉ thị của các **chương trình**, và được đặt cho một cái tên.*

Khi gán nhãn cho hai câu trên, chương trình sẽ gặp khó khăn khi phải gán nhãn từ loại cho từ *program,* do từ *program* có hai từ loại : danh từ (NN), và động từ (VB). Nhưng khi có thêm thông tin tiếng Việt, ở đây là thông tin có được do sử dụng liên kết từ,

xác của câu. Vì khi từ *program* liên kết với từ "lập trình", sẽ có từ loại là động từ, còn khi liên kết với từ "chương trình" sẽ có từ loại là danh từ.

Nhờ các thông tin bổ sung của tiếng Viêt, ta sẽ giải quyết được khá nhiều lỗi còn sót lại sau khi sửa lỗi bằng các bộ luật của FnTBL. Các thông tin tiếng Việt, cụ thể ở đây là mối liên kết từ, sẽ được sử dụng bằng cách tạo ra một ngữ liệu đúng (golden corpus) trong đó có cả thông tin về từ loại và các mối liên kết từ. Sau đó, ta sẽ dùng mô hình FnTBL rút ra các luật để sửa các lỗi còn lại. Trong quá trình huấn luyện rút luật này, quá trình gán nhãn khởi tạo là kết quả của mô hình kết hợp ở trên.

Ngoài ra, ta còn có thể sử dụng thông tin về từ loại tiếng Việt để bổ sung cho các liên kết từ mà từ tiếng Việt chỉ có một từ loại.

3.3.5 *Chiếu sang tiếng Việt*

Hiện nay, các bài toán xử lý ngôn ngữ tự nhiên trên tiếng Việt còn gặp rất nhiều hạn chế. Các giải thuật áp dụng cho bài toán xử lý ngôn ngữ nhất khi áp dụng trên tiếng Việt không nhận được kết quả như mong muốn, đặc biệt hiện nay chúng ta chưa có được những bộ ngữ liệu cần thiết cho việc áp dụng các giải thuật trên tiếng Việt. Chi phí để tạo nên một bộ ngữ liệu đủ lớn cho các bài toán về ngôn ngữ là rất lớn. Một trong những hướng tiếp cận hiện nay để giải quyết tình trạng thiếu các nguồn ngữ liệu là dùng kết quả sẵn có trên các ngôn ngữ khác để tạo nên nguồn ngữ liệu cho tiếng Việt.

Để xây dựng nguồn ngữ liệu có gán nhãn từ loại cho tiếng Việt chúng ta có thể dùng các bộ ngữ liệu đã có gán nhãn từ loại của tiếng Anh, thông qua mỗi liên kết từ với tiếng Việt, chúng ta sử dụng các ánh xạ về từ loại giữa hai ngôn ngữ Anh - Việt để chiếu các từ loại từ tiếng Anh sang tiếng Việt([12]).

Việc chiếu từ loại từ tiếng Anh sang tiếng Việt không chỉ đơn giản từ loại của từ tiếng Việt tương ứng với từ tiếng Anh sẽ có cùng từ loại với từ tiếng Anh, vì chúng ta biết rằng tiếng Việt và tiếng Anh thuộc hai loại hình

khác nhau nên từ loại của cặp từ tương ứng trong tiếng Việt và tiếng Anh là không giống nhau. Vì chúng ta chưa có các ngữ liệu về từ loại trên tiếng Việt nên chúng ta phải tạo nên các ánh xạ từ tiếng Anh sang tiếng Việt bằng tay. Các ánh xạ này có dạng với từ loại tiếng Anh là A thì từ loại tiếng Việt là B.
Ví dụ câu :

I	*Can*	*Can*	*A*	*Can*
PRP	*MD*	*VB*	*DT*	*NN*
Tôi	*Có thể*	*Đóng*	*một*	*Cái hộp*
P	*MD*	*V*	*DT*	*N*

Từ bảng trên chúng ta có thể tạo nên những ánh xạ có dạng từ loại tiếng Anh là PRP (Pronuon) thì từ loại tiếng Việt là P (Pronoun), từ loại tiếng Anh là VB (Verb) thì từ loại tiếng việt là V (Verb). Chúng ta sẽ tiến hành tạo các ánh xạ cho tất cả các từ loại trong tiếng Anh.

Các ánh xạ này sẽ được áp dụng vào một bộ ngữ liệu , kết quả của việc áp dụng này sẽ được dùng để kiểm tra xem ánh xạ trên có chính xác hay không đồng thời tìm ra các ánh xạ mới mà trong quá trình làm bằng tay chúng ta chưa nhận ra. Với những ánh xạ không đồng nhất như từ loại tiếng Anh là A thì từ loại tiếng Việt là B và C thì chúng ta tạo thêm các ánh xạ trong đó có kèm thêm ngữ cảnh để việc chiếu được chính xác hơn. Với các ánh xạ trên chúng ta có thể xây dựng được một bộ dùng trong việc ánh xạ từ loại từ từ tiếng Anh sang tiếng Việt. Các luật này có định dạng như sau :

pos_0=PRP => posvn_0=P

pos_0=VB => posvn_0=V

Các luật này có thể dùng để tạo bộ ngữ liệu về từ loại trên tiếng Việt từ bộ ngữ liệu về từ loại trên tiếng Anh, hay cũng có thể dùng để tạo một bộ ngữ liệu mới bằng cách sử dụng chương trình gán nhãn từ loại cho tiếng Anh và chiếu kết quả của việc gán nhãn từ loại này sang tiếng Việt.

Việc chiếu từ loại này mang một ý nghĩa to lớn trong việc xây dựng bộ ngữ liệu tiếng Việt. Bộ ngữ liệu nhận được có kết quả không hoàn toàn chính xác nhưng nó sẽ giúp cho công việc gán nhãn từ loại giảm nhẹ đi rất nhiều, công việc chuyển từ việc phải tiến hành gán nhãn từ loại bằng tay thành việc chỉnh sửa các lỗi xảy ra do quá trình làm tự động này.

Chương 4

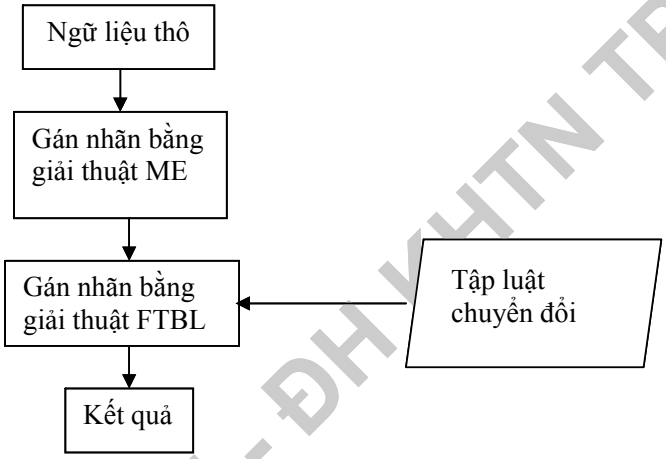
Cài đặt thử nghiệm và đánh giá kết quả.

Trong chương này, chúng tôi xin trình bày về cài đặt cụ thể của chương trình, kết quả thử nghiệm với các mô hình khác nhau, và đánh giá kết quả của các mô hình này.

4.1 Cài đặt

4.1.1 Cài đặt bộ gán nhãn từ loại dựa trên mô hình kết hợp FnTBL và ME.

- **Sơ đồ mô hình**



Hình 4-1: Sơ gán nhãn cho mô hình kết hợp

- **Cài đặt**

Đầu vào : câu tiếng Anh cần gán nhãn từ loại.

Đầu ra : tập nhãn kết quả.

Bước 1 : tiền xử lý.

Bước 2 : Gán nhãn cơ sở bằng giải thuật ME

lặp, với từ w_i trong câu cần gán nhãn từ loại.

tính xác suất từ w_i có nhãn t_j : $p(w_i,t_j)$

chọn nhãn có xác suất p cao nhất.

Bước 3 : gán nhãn từ loại bằng FnTBL

Lặp, với luật r_i trong tập luật huân luyện trên mô hình kết hợp

Lặp, với từ w_j trong câu cần gán nhãn từ loại.

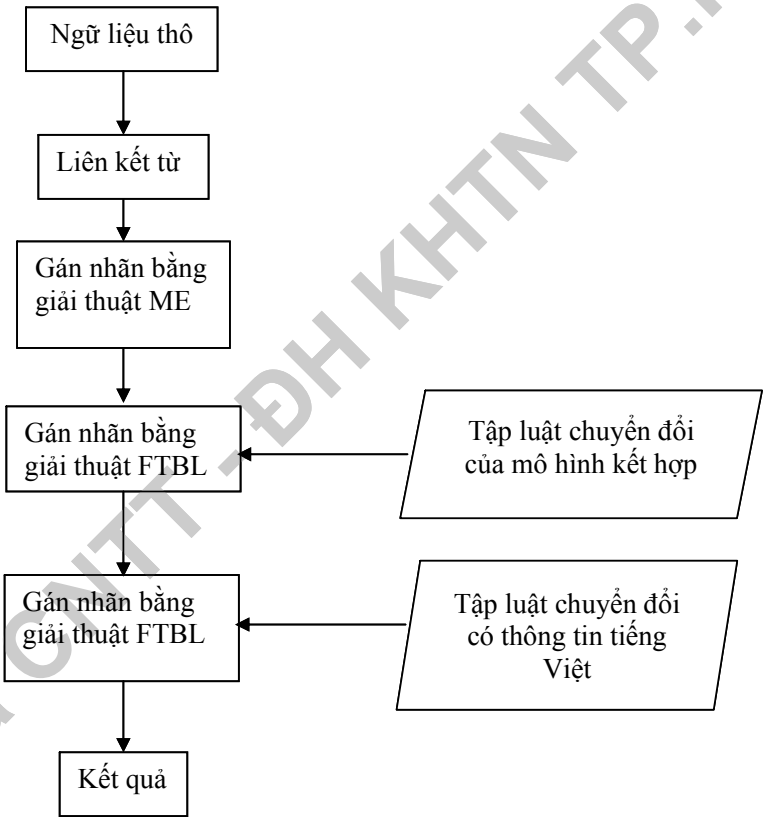
nếu ngữ cảnh của từ w_j phù hợp với ngữ cảnh trong luật r_i

áp dụng luật r_i cho từ w_j

return kết_quả

4.1.2 Cài đặt bộ gán nhãn từ loại có sử dụng thông tin tiếng Việt.

- **Sơ đồ mô hình**



Hình 4-2: Sơ đồ mô hình gán nhãn sử dụng thông tin tiếng Việt.

- **Cài đặt**

Đầu vào : câu tiếng Anh cần gán nhãn từ loại và câu tiếng Việt được dich từ câu tiếng Anh trên.

Đầu ra : tập từ loại của câu tiếng Anh

Bước 1 : tiền xử lý trên hai câu tiếng Việt và tiếng Anh.

Bước 2 : liên kết từ hai câu tiếng anh và tiếng Việt.

Bước 3 : gán nhãn từ loại cho câu tiếng Anh bằng mô hình kết hợp.

Bước 4 : Sử dụng thông tin tiếng Việt để chỉnh sửa các nhãn sai trong mô hình kết hợp.

Lặp, với các luật r_i trong tập luật chuyển đổi có thông tin tiếng Việt

Lặp, với các từ wi trong câu tiếng Anh

kiểm tra ngữ cảnh tiếng Anh trong luật r_i có phù hợp với ngữ cảnh của từ w_i trong câu tiếng Anh không?

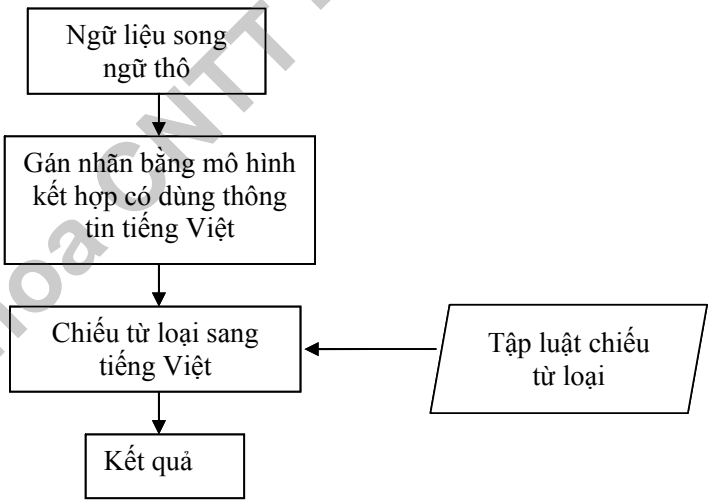
nếu đúng thông qua mối liên kết từ với câu tiếng Việt kiểm tra thông tin tiếng Việt đi kèm trong luật chuyển đổi có đúng không.

nếu đúng áp dụng luật r_i cho từ w_i

return kết_quả

4.1.3 Cài đặt mô hình chiếu từ loại từ tiếng Anh sang tiếng Việt

- **Sơ đồ mô hình**



Hình 4-3: Sơ đồ mô hình chiếu từ loại sang tiếng Việt.

Cài đặt

Đầu vào : cặp câu song ngữ Anh-Việt chưa gán nhãn từ loại

Đầu ra : kết quả gán nhãn từ loại trên câu tiếng Anh và tiếng Việt

Bước 1 : tiền xử lý trên cặp câu song ngữ

Bước 2 : liên kết từ cho cặp câu song ngữ

Bước 3 : gán nhãn từ loại cho câu tiếng Anh bằng mô hình kết hợp có dùng thêm thông tin tiếng Anh

Bước 4 : chiếu kết quả từ loại trên câu tiếng Anh sang tiếng Việt

Lặp, với mỗi từ w_i trong câu tiếng Anh

với mỗi luật chiếu từ loại r_j

kiểm tra điều kiện ngữ cảnh có phù hợp với luật r_j

nếu đúng thì áp dụng luật chiếu r_j cho từ tiếng Việt w_k

return kết_quả

4.2 Thử nghiệm

4.2.1 Thử nghiệm với các mô hình khởi tạo khác nhau.

Trong hướng tiếp cận này chúng tôi sử dụng phương pháp kết hợp các giải thuật gán nhãn tiên tiến và có độ chính xác cao nhất hiện nay. Như đã trình bày ở phần trên, hướng tiếp cận này là dùng một giải thuật tốt nhất hiện nay để gán nhãn cơ sở, từ đó mô hình sẽ dùng giải thuật FnTBL để chỉnh sửa các lỗi trên nhãn cơ sở để nhận được kết quả tốt nhất. Việc chọn bộ gán nhãn cơ sở là tương đối quang trọng vì như đã trình bày giải thuật dùng gán nhãn cơ sở phải sử dụng thông tin ngữ cảnh phù hợp để sau đó chúng ta dùng giải thuật FnTBL chỉnh sửa có hiệu quả. Ngoài ra việc chọn mức ngưỡng trong quá trình học của giải thuật FnTBL cũng khá quan trọng. Nếu chúng ta chọn mức ngưỡng để học không phù hợp thì kết quả gán nhãn sẽ không đạt như chúng ta

mong muốn. Nếu chọn mức ngưỡng quá lớn thì kết quả sẽ không chính xác ngược lại thì kết quả gán nhãn sẽ rơi vào trường hợp quá chi tiết.

4.2.1.1 Kết quả thử nghiệm dùng Unigram là giải thuật gán nhãn cơ sở.

Giải thuật Unigram là giải thuật gán nhãn dựa trên thống kê các từ loại trong ngữ liệu học. Trong quá trình gán nhãn cho ngữ liệu nếu từ tồn tại trong ngữ liệu học thì giải thuật chọn nhãn có tầng số xuất hiện cao nhất trong ngữ liệu học cho từ hiện tại, ngược lại giải thuật sẽ dùng các heuristic để gán cho từ hiện tại một từ loại nào đó. Giải thuật này không dùng đến thông tin ngữ cảnh mà từ đó tồn tại (như từ hay từ loại của từ phía trước, từ phía sau hay từ hiện tại). Việc không dùng ngữ cảnh làm cho giải thuật này không có độ chính xác cao như các giải thuật khác. Tính ưu việt của giải thuật là ở chổ giải thuật đơn giản. Trong quá trình thử nghiệm chúng tôi đã dùng giải thuật này để tiến hành gán nhãn cơ sở cho mô hình của mình. Quá trình thử nghiệm trên việc gán nhãn cơ sở bằng giải thuật Unigram với các mức ngưỡng khác nhau cho kết quả như sau.

- ***Với mức ngưỡng là 4***
  - Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*176*
SUSANNE	*150368*	*345*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*289*
Toàn bộ ngữ liệu	*363908*	*413*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,3%*	*94,4%*
SUSANNE	*10081*	*89,1%*	*93,5%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,1%*	*95,3%*

❖ ***Với mức ngưỡng là 3***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*275*
SUSANNE	*150368*	*957*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*1018*
Toàn bộ ngữ liệu	*363908*	*1521*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,3%*	*95,6%*
SUSANNE	*10081*	*89,1%*	*94,9%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,1%*	*95,7%*

❖ ***Với mức ngưỡng là 2***

- Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*603*
SUSANNE	*150368*	*2013*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*1845*
Toàn bộ ngữ liệu	*363908*	*2267*

- Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,3%*	*96,4%*
SUSANNE	*10081*	*89,1%*	*95,9%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,1%*	*96,2%*

❖ ***<u>Với mức ngưỡng là 1</u>***

- Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*1307*
SUSANNE	*150368*	*3576*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*4012*
Toàn bộ ngữ liệu	*363908*	*4312*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,3%*	*96,3%*
SUSANNE	*10081*	*89,1%*	*96,0%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,1%*	*96,2%*

4.2.1.2 Kết quả thử nghiệm với nhãn khởi tạo của mô hình Markov ẩn

❖ ***Với mức ngưỡng là 4***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*167*
SUSANNE	*150368*	*354*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*314*
Toàn bộ ngữ liệu	*363908*	*413*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,5%*	*93,4%*
SUSANNE	*10081*	*91,5%*	*94,1%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,3*	*93,8%*

❖ ***Với mức ngưỡng là 3***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*234*
SUSSANNE	*150368*	*489*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*512*
Toàn bộ ngữ liệu	*363908*	*567*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,5%*	*95,4%*
SUSSANNE	*10081*	*91,5%*	*95,1%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,3*	*95,8%*

❖ ***Với mức ngưỡng là 2***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*543*
SUSANNE	*150368*	*671*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*673*
Toàn bộ ngữ liệu	*363908*	*741*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,5%*	*96,6%*
SUSANNE	*10081*	*91,5%*	*96,7%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,3*	*96,5%*

❖ ***Với mức ngưỡng là 1***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*890*
SUSANNE	*150368*	*1045*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*1145*
Toàn bộ ngữ liệu	*363908*	*1342*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*90,5%*	*96,5%*
SUSANNE	*10081*	*91,5%*	*96,4%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*91,3*	*96,5%*

4.2.1.3 Kết quả thử nghiệm dùng Maximum Entropy làm giải thuật gán nhãn cơ sở.

Kết quả nhận được khi thử nghiệm với việc dùng thuật toán maximum Entropy làm giải thuật gán nhãn cơ sở đã nhận được những kết quả hết sức khả quan.

- ***Với mức ngưỡng là 4***
  - Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*203*
SUSANNE	*150368*	*316*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*233*
Toàn bộ ngữ liệu	*363908*	*360*

  - Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*96%*	*97,1%*
SUSANNE	*10081*	*95,3%*	*96,8%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*95,9%*	*96,5%*

❖ ***Với mức ngưỡng là 3***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*275*
SUSANNE	*150368*	*391*
Mổt phân bộ ngữ liệu Penn Tree Bank	*125202*	*350*
Toàn bộ ngữ liệu	*363908*	*420*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*96%*	*97,2%*
SUSANNE	*10081*	*95,3%*	*97,1%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*95,9%*	*96,8%*

❖ ***Với mức ngưỡng là 2***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*492*
SUSANNE	*150368*	*615*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*540*
Toàn bộ ngữ liệu	*363908*	*750*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*96%*	*97,8%*
SUSANNE	*10081*	*95,3%*	*97,4%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*95,9%*	*97,5%*

❖ ***Với mức ngưỡng là 1***

➢ Kết quả học

Bộ ngữ liệu	*Số từ trong ngữ liệu học*	*Số luật học được*
Cadasa	*88338*	*1611*
SUSANNE	*150368*	*2161*
Một phần bộ ngữ liệu Penn Tree Bank	*125202*	*1904*
Toàn bộ ngữ liệu	*363908*	*2371*

➢ Quá trình gán nhãn

Bộ ngữ liệu	*Số từ trong ngữ liệu đánh giá*	*Kết quả gán nhãn cơ sở*	*Kết quả đầu ra của Fast TBL*
Cadasa	*6202*	*96%*	*97,8%*
SUSANNE	*10081*	*95,3%*	*97,4%*
Một phần bộ ngữ liệu Penn Tree Bank	*12504*	*95,9%*	*97,6%*

4.2.2 Thử nghiệm với các khung luật khác nhau cho giải thuật TBL nhanh

Với các khung luật khác nhau chúng ta nhận được các kết quả học khác nhau. Số lượng các khung luật cũng như ngữ cảnh mà mỗi khung luật quy định ảnh hưởng rất nhiều đến kết quả của việc gán nhãn.

Với các khung luật khác nhau thì số lượng luật học nhận được là khác nhau. Khi có càng nhiều khung luật được dùng trong quá trình học thì số lượng luật học ra giảm xuống, nhưng số luật giảm không đều trên số lượng các khung luật khác nhau.

Số lượng khung luật	*Số luật học được*
13	*623*
20	*572*
40	*492*

Với các khung luật có độ rộng ngữ cảnh nhỏ thì số luật học ra sẽ ít nhưng kết quả việc gán nhãn sẽ bị giảm đi vì khi đó giải thuật dễ rơi vào tình trạng quá chi tiết.

Với các khung luật trong đó chọn ngữ cảnh chỉ dựa trên ngữ cảnh là các từ loại của các từ chung quanh mà không xem xét đến các từ chung quanh nó, thì luật chúng ta học ra sẽ mang tính tổng quát các luật này sẽ chỉnh được các trường hợp tổng quát Nhưng với các trường hợp bất qui tắc hay những trường hợp có ngữ cảnh đặc biệt ít xuất hiện thì các luật học ra sẽ không chỉnh sửa được. Ngược lại các khung luật có thông tin từ bên cạnh nó thì sẽ chỉnh được các trường hợp bất qui tắc hay các trường hợp có ngữ cảnh đặc biệt nhưng các luật này không tổng quát. Trong mô hình này chúng tôi kết hợp tất cả các khung luật có ngữ cảnh khác nhau chứa thông tin của từ bên cạnh nó nhằm tránh các nhược điểm của ở trên.

4.2.3 Kết quả gán nhãn từ loại khi dùng thông tin tiếng Việt

Bộ ngữ liệu	*Số lượng từ trong ngư liệu đánh giá*	*Không sử dụng thông tin tiếng Việt*	*Sử dụng thông tin tiếng Việt*
Cadasa	*6202*	*97,8%*	*98,5%*
Susan	*10081*	*97,4%*	*98,1%*

4.3 Nhận xét

Qua các thử nghiệm trên các bộ gán nhãn cơ sở cũng như các mức ngưỡng khác nhau chúng ta thấy với mỗi mức ngưỡng và với mỗi bộ gán nhãn khác nhau kết quả chúng ta nhận được là khác nhau và kết quả tối ưu là kết quả với việc chọn giải thuật Maximum Entropy làm bộ gán nhãn cơ sở và với mức ngưỡng là 2 thì chúng ta nhận được kết quả tối ưu nhất

Chương 5

Tổng kết

5.1 Kết quả đạt được

Như đã đề cập ở phần đầu, hiện nay, khá nhiều bộ gán nhãn từ loại tiếng Anh trên thế giới đạt được độ chính xác cao. Các bộ gán nhãn này đã tận dụng gần hết thông tin về từ vựng cũng như ngữ cảnh được cung cấp. Do đó, nếu ta chỉ cải tiến các giải thuật trên một cách thông thường thì khó có thể tăng độ chính xác của bộ gán nhãn. Vì vậy, chúng tôi đã kết hợp một số mô hình gán nhãn đã đạt kết quả lại với nhau. Tận dụng ưu điểm của từng bộ gán nhãn để làm tăng kết quả của bộ gán nhãn. Bên cạnh đó, để sử dụng thêm thông tin tiếng Việt, chúng tôi đã tiến hành gán nhãn từ loại trên cặp câu song ngữ Anh-Việt thay vì chỉ gán nhãn từ loại trên đơn ngữ.

Trong mô hình này, chúng tôi đã tận dụng ưu điểm của mô hình FnTBL là có thể sửa nhãn trên một bộ dữ liệu đã được gán nhãn bởi một mô hình khác để cải tiến chất lượng mô hình. Do đó, chúng tôi đã chọn mô hình gán nhãn từ loại tiếp cận theo hướng thống kê_MAXIMUM ENTROPY, để làm chương trình gán nhãn khởi tạo cho mô hình FnTBL. Do mô hình FnTBL là mô hình học hướng lỗi, nên mô hình chỉ sửa các lỗi sai của quá trình khởi tạo. Điều này đảm bảo sự kết hợp của hai mô hình này chắc chắn sẽ làm tăng kết quả của bộ gán nhãn lên. Chúng tôi đã tiến hành đánh giá kết quả các mô hình trên dữ liệu Suasn và kết quả đạt được như sau:

Mô hình	*MAXIMUM ENTROPY*	*FnTBL*	*Mô hình kết hợp*	*Mô hình kết hợp có sử dụng thông tin tiếng Việt*
Độ chính xác(%)	*96,2*	*96,6*	*97,8*	*98,5*

Ngoài ra chúng tôi còn kiểm tra trên một số dữ liệu tin học và kết quả cũng tương tự như vậy. Đặc biệt, kết quả của mô hình tốt hơn khi áp dụng với các dữ liệu về tin học.

Tuy nhiên, nếu chỉ kết hợp hai mô hình trên thì vẫn còn một số trường hợp mà chúng không đủ thông tin để khử nhập nhằng. Để giải quyết vấn để này, chúng tôi đã quyết định sử dụng thêm các thông tin bổ sung từ tiếng Việt. Đó là từ và từ loại của từ tiếng Việt tương ứng với từ tiếng Anh đang xét. Các thông tin này có được là nhờ mối liên kết từ trong song ngữ Anh-Việt.

Đây chính là điểm khác biệt của mô hình chúng tôi so với các mô hình gán nhãn từ loại hiện nay. Trong bài toán gán nhãn từ loại của mình chúng tôi đã tiến hành học trên ngữ liệu song ngữ(bằng mô hình FnTBL) để tìm ra các mối quan hệ giữa từ và từ loại trên hai ngôn ngữ là tiếng Anh và tiếng Việt. Những mối liên hệ này chính là cơ sở cho việc chọn từ loại cho từ trong ngữ liệu cần gán nhãn. Thông qua quá trình học sẽ rút ra các luật sửa các lỗi mà mô hình FnTBL chưa sửa được. Và kết quả gán nhãn đã tăng thêm khoảng 0,5% nhờ áp dụng các thông tin về tiếng Việt.

Ngoài ra, chúng tôi có sử dụng kết quả đạt được từ bộ gán nhãn để xây dựng một bộ ngữ liệu song ngữ Anh-Việt, trong đó các câu tiếng Việt cũng được gán nhãn từ loại. Chúng tôi đã sử dụng các thông tin có sẵn về nhãn từ loại của tiếng Anh, các thông tin về tiếng Việt và môi liên hệ giữa hai ngôn ngữ này để thực hiện chiếu kết quả từ câu tiếng Anh sang câu tiếng Việt. Tuy nhiên, kết quả đạt được ở phần này còn tương đối hạn chế.

5.2 Hạn chế

Các kết quả đạt được trong luận văn này còn nhiều hạn chế. Mặc dù mô hình đã tận dụng được một số thông tin có trong song ngữ Anh-Việt nhưng kết quả đạt được vẫn còn hạn chế. Lý do chính ở đây là do bộ ngữ liệu huấn luyện tương đối nhỏ, chưa đủ lớn để bao quát các trường hợp lỗi. Do đó, vẫn còn một số lỗi mà chương trình vẫn chưa sửa được.

Do ngữ liệu được sử dụng trong mô hình này là ngữ liệu song ngữ đã được liên kết từ nên tốn khá nhiều thời gian để xây dựng ngữ liệu này. Vì vậy, chúng tôi chỉ đủ thời gian xây dựng một ngữ liệu nhỏ đủ để làm dữ liệu huấn luyện cho mô hình.

Ngoài ra, các luật phát sinh trong quá trình học vẫn có một số luật không tốt. Do đó, chúng tôi vẫn phải hiệu chỉnh lại bằng tay để làm tăng kết quả của bộ gán nhãn. Và qua kiểm tra, chúng tôi thấy răng sau khi hiệu chỉnh luật thì đã hạn chế được một số lỗi của bộ gán nhãn.

5.3 Hướng phát triển:

Mặc dù kết quả đạt được của mô hình này tương đối cao, khoảng 98,5%. Nhưng mô hình này vẫn còn có khả năng phát triển tiếp tục. Nếu ngữ liệu liệu được xây dựng tốt hơn thì độ chính xác của chương trình sẽ còn tăng lên khá nhiều. Ngoài ra, còn khá nhiều thông tin về tiếng Việt mà chúng ta có thể phát khai thác để nâng cao kết quả chương trình.

Bên cạnh đó, do thời gian hạn chế nên trong luận văn này chúng tôi chỉ tập trung gán nhãn từ loại trên câu tiếng Anh. Còn phần ánh xạ kết quả qua tiếng Việt vẫn còn chưa tốt. Nếu ta có đủ thời gian để xây dựng một ngữ liệu tốt hơn nữa thì kết quả của việc gán nhãn từ loại sẽ tiếp tục tăng lên. Nhờ vậy, chất lượng của hệ dịch máy cũng sẽ tăng.

Phụ lục A:Các tập nhãn của Penn Tree Bank

Trong luận văn, để đánh dấu nhãn từ loại (POS) cũng như các thành phần cú pháp, tôi sử dụng bộ nhãn của Penn Tree Bank (ngữ liệu tiếng Anh thông dụng trên thế giới hiện nay):

- CÁC NHÃN TỪ LOẠI (gồm 36 nhãn, không tính dấu ngắt):

STT	**Nhãn từ loại**	**Từ viết tắt**	**Ý nghĩa**
1	CC	Coordinating conjunction	Liên từ
2	CD	Cardinal number	Số từ
3	DT	Determiner	Định từ
4	EX	Existential "there"	Có
5	FW	Foreign word (	Từ nước ngoài
6	IN	Preposition or subordinating conjunction	Giới từ
7	JJ	Adjective	Tính từ
8	JJR	Adjective, comparative	Tính từ so sánh hơn
9	JJS	Adjective, superlative	Tính từ so sánh cực cấp
10	LS	List item marker	Dấu liệt kê
11	MD	Modal	Từ tình thái
12	NN	Noun, singular or mass	Danh từ, số ít hay không đếm được
13	NNS	Noun, plural	Danh từ số nhiều
14	NNP	Proper noun, singular	Danh từ riêng số ít
15	NNPS	Proper noun, plural	Danh từ riêng số nhiều
16	PDT	Predeterminer	Tiền chỉ định từ
17	POS	Possessive ending	Dấu cuối của sở hữu cách
18	PRP	Personal pronoun	Đại từ nhân xưng

19	PRP$	Possessive pronoun	Đại từ sở hữu
20	RB	Adverb	Trạng từ
21	RBR	Adverb, comparative	Trạng từ so sánh hơn
22	RBS	Adverb, superlative	Trạng từ so sánh cực cấp
23	RP	Particle	Tiểu từ
24	SYM	Symbol	Ký hiệu
25	TO	"to"	Từ "to"
26	UH	Interjection	Thán từ
27	VB	Verb, base form	Động từ dạng nguyên thể
28	VBD	Verb, past tense	Động từ quá khứ
29	VBG	Verb, gerund or present participle	Danh động từ / hiện phần từ
30	VBN	Verb, past participle	Động từ quá phân từ
31	VBP	Verb, non-3rd person singular present	Động từ không phải ngôi 3 số ít hiện tại
32	VBZ	Verb, 3rd person singular present	Động từ ngôi 3 số ít, hiện tại
33	WDT	Wh-determiner	Định từ bắt đầu bắt Wh-
34	WP	Wh-pronoun	Đại từ bắt đầu bắt Wh-
35	WP$	Possessive wh-pronoun	Đại từ sở hữu bắt đầu bắt Wh-
36	WRB	Wh-adverb	Trạng từ bắt đầu bắt Wh-

Phụ lục B: Bộ nhãn từ loại tiếng Việt.

STT	Nhãn từ loại	Ý nghĩa
1	CC	Liên từ
2	CD	Số từ
3	DT	Định từ
4	FW	Từ nước ngoài
5	IN	Giới từ
6	A	Tính từ)
7	LS	Dấu liệt kê
8	MD	Từ tình thái
9	N	Danh từ
10	POS	Sở hữu cách
11	P	Đại từ nhân xưng
12	P$	Đại từ sở hữu
13	R	Trạng từ
14	RP	Tiểu từ
15	SYM	Ký hiệu
16	UH	Thán từ
17	V	Động từ

Phụ lục C: Bảng ánh xạ từ loại từ tiếng Anh sang tiếng Việt.

STT	Nhãn từ loại tiếng Anh	Nhãn từ loại tiếng Việt
1	CC	CC
2	CD	CD
3	DT	DT
4	EX	V
5	FW	FW
6	IN	IN
7	JJ	A
8	JJR	A
9	JJS	A
10	LS	LS
11	MD	MD
12	NN	N
13	NNS	N
14	NNP	N
15	NNPS	N
16	PDT	DT
17	POS	POS
18	PRP	P
19	PRP$	P$
20	RB	R
21	RBR	R
22	RBS	R
23	RP	RP
24	SYM	SYM

25	TO	-
26	UH	UH
27	VB	V
28	VBD	V
29	VBG	V
30	VBN	V
31	VBP	V
32	VBZ	V
33	WDT	P
34	WP	P
35	WP$	P$
36	WRB	R

Phụ lục D: Một số luật chuyển đổi.

1.GOOD:169 BAD:15 SCORE:154 RULE: pos_0=VBZ pos_-1=NNS pos_-2=NN => pos=VBP

2.GOOD:149 BAD:12 SCORE:137 RULE: pos_0=VBP pos:[-3,-1]=MD => pos=VB

3.GOOD:120 BAD:0 SCORE:120 RULE: pos_0=AUX pos:[1,3]=DT => pos=VBZ

4.GOOD:123 BAD:3 SCORE:120 RULE: pos_0=VBZ pos:[-3,-1]=MD => pos=VB

5.GOOD:98 BAD:9 SCORE:89 RULE: pos_0=NNS pos_1=CD => pos=NNP

6.GOOD:81 BAD:5 SCORE:76 RULE: pos_0=NNP pos_-1=VB pos_1=CD => pos=NN

7.GOOD:42 BAD:0 SCORE:42 RULE: pos_0=AUX pos:[-3,-1]=NNP => pos=VBZ

8.GOOD:39 BAD:0 SCORE:39 RULE: pos_0=AUX pos:[1,3]=NNS => pos=VBP

9.GOOD:28 BAD:0 SCORE:28 RULE: pos_0=VBZ pos_-1=TO => pos=VB

10.GOOD:24 BAD:0 SCORE:24 RULE: pos_0=AUXG pos_1=VBN => pos=VBG

11.GOOD:49 BAD:25 SCORE:24 RULE: pos_0=NNP pos_-1=NN => pos=NN

12.GOOD:23 BAD:0 SCORE:23 RULE: pos_0=AUX pos:[1,3]=VBN => pos=VB

13.GOOD:23 BAD:2 SCORE:21 RULE: pos_0=VBZ pos_-1=VBZ => pos=VBN

14.GOOD:28 BAD:8 SCORE:20 RULE: pos_0=VBP pos_-1=TO => pos=VB

15.GOOD:16 BAD:0 SCORE:16 RULE: pos_0=AUX pos:[1,3]=NN => pos=VBZ

16.GOOD:14 BAD:0 SCORE:14 RULE: pos_0=AUX pos_1=PRP => pos=VBP

17.GOOD:14 BAD:0 SCORE:14 RULE: pos_0=AUXG pos:[-3,-1]=IN => pos=VBG

18.GOOD:14 BAD:0 SCORE:14 RULE: pos_0=AUX pos:[1,3]=JJ => pos=VB

19.GOOD:13 BAD:0 SCORE:13 RULE: pos_0=RB pos_-1=NNS pos_1=NN => pos=NNP

20.GOOD:26 BAD:1 SCORE:25 RULE: pos_0=NNS pos_1=NNP => pos=NNP

21.GOOD:13 BAD:1 SCORE:12 RULE: pos_0=NN pos_-1=, pos_1=DT => pos=VB

22.GOOD:10 BAD:0 SCORE:10 RULE: pos_0=JJ pos_-1=ZZZ pos_1=DT => pos=VB

23.GOOD:10 BAD:1 SCORE:9 RULE: pos_0=CD pos_-1=CD pos_1=CC => pos=NN

24.GOOD:9 BAD:0 SCORE:9 RULE: pos_0=VBZ pos_-1=MD => pos=VB

25.GOOD:8 BAD:0 SCORE:8 RULE: pos_0=VBZ pos_-1=NNS pos_-2=, => pos=VBP

26.GOOD:10 BAD:2 SCORE:8 RULE: pos_0=NN pos_-1=RB pos_1=NN => pos=VB

27.GOOD:15 BAD:8 SCORE:7 RULE: pos_0=VBZ pos_-1=NN pos_-2=CC => pos=VBP

28.GOOD:11 BAD:4 SCORE:7 RULE: pos_0=NNP pos_-1=: pos_-2=NN => pos=NN

29.GOOD:7 BAD:0 SCORE:7 RULE: pos_0=AUX pos_-1=DT => pos=VBZ

30.GOOD:9 BAD:3 SCORE:6 RULE: pos_0=VBZ pos_-1=PRP pos_1=VBG => pos=VBP

31.GOOD:6 BAD:0 SCORE:6 RULE: pos_0=NN pos_-1=VBN pos_1=VB => pos=SYM

32.GOOD:10 BAD:4 SCORE:6 RULE: pos_0=NNS pos_1=: pos_2=VBN => pos=NNP

33.GOOD:6 BAD:0 SCORE:6 RULE: pos_0=FW pos_-1=NNP => pos=NNP

34.GOOD:5 BAD:0 SCORE:5 RULE: pos_0=VBP pos_-1=NNP pos_-2=, => pos=VBZ

35.GOOD:5 BAD:0 SCORE:5 RULE: pos_0=VBZ pos_-1=NNS pos_-2=POS => pos=VBP

36.GOOD:6 BAD:1 SCORE:5 RULE: pos_0=NN pos_-1=RB pos_1=DT => pos=VB

37.GOOD:5 BAD:0 SCORE:5 RULE: pos_0=NN pos_-1=, pos_1=VB => pos=VB

38.GOOD:5 BAD:0 SCORE:5 RULE: pos_0=VBZ pos_-1=RB pos_-2=VBZ => pos=VB

40.GOOD:7 BAD:2 SCORE:5 RULE: pos_0=NNS pos_-1=IN pos_1=NNS => pos=NNP

Phụ lục E: Kết quả gán nhãn từ loại trong mô hình kết hợp không dùng thông tin tiếng Việt

Kết quả gán nhãn cơ sở	**Kết quả gán nhãn từ loại bằng mô hình kết hợp**
Most/JJS computers/NNS from/IN the/DT biggest/JJS to/TO the/DT smallest/JJS **operate/VBP** on/IN the/DT same/JJ fundamental/JJ principles/NNS ./.	Most/JJS computers/NNS from/IN the/DT biggest/JJS to/TO the/DT smallest/JJS operate/NN on/IN the/DT same/JJ fundamental/JJ principles/NNS ./.
They/PRP are/VBP all/DT fabricated/VBN from/IN the/DT same/JJ basic/JJ types/NNS of/IN components/NNS ,/, and/CC they/PRP **all/DT need/VB** instructions/NNS to/TO make/VB them/PRP run/VB ./.	They/PRP are/VBP all/DT fabricated/VBN from/IN the/DT same/JJ basic/JJ types/NNS of/IN components/NNS ,/, and/CC they/PRP all/RB need/VBP instructions/NNS to/TO make/VB them/PRP run/VB ./.
Any/DT computer/NN -/: regardless/RB of/IN its/PRP$ type/NN -/: is/VBZ controlled/VBN by/IN programmed/JJ instructions/NNS ,/, which/WDT **give/VB** the/DT machine/NN a/DT purpose/NN and/CC **tell/VB** it/PRP what/WP to/TO do/VB ./.	Any/DT computer/NN -/: regardless/RB of/IN its/PRP$ type/NN -/: is/VBZ controlled/VBN by/IN programmed/JJ instructions/NNS ,/, which/WDT give/VBP the/DT machine/NN a/DT purpose/NN and/CC tell/VBP it/PRP what/WP to/TO do/VB ./.
Other/JJ types/NNS of/IN programs/NNS exist/VBP primarily/RB for/IN the/DT user/NN and/CC **enable/VB** the/DT computer/NN to/TO perform/VB tasks/NNS ,/, such/JJ as/IN creating/VBG documents/NNS or/CC **drawing/NN** pictures/NNS ./.	Other/JJ types/NNS of/IN programs/NNS exist/VBP primarily/RB for/IN the/DT user/NN and/CC enable/VBP the/DT computer/NN to/TO perform/VB tasks/NNS ,/, such/JJ as/IN creating/VBG documents/NNS or/CC drawing/VBG pictures/NNS ./.
People/NNS are/VBP the/DT computer/NN operators/NNS ,/, also/RB known/VBN as/IN users/NNS ./.	People/NNS are/VBP the/DT computer/NN operators/NNS ,/, also/RB known/VBN as/IN users/NNS ./.
The/DT microprocessor/NN is/VBZ **plugged/VBD** into/IN the/DT computer/NN 's/POS motherboard/NN ./.	The/DT microprocessor/NN is/VBZ plugged/VBN into/IN the/DT computer/NN 's/POS motherboard/NN ./.
When/WRB you/PRP launch/VB a/DT program/NN ,/, it/PRP is/VBZ	When/WRB you/PRP launch/VBP a/DT program/NN ,/, it/PRP is/VBZ

loaded/VBN into/IN and/CC run/VBN from/IN memory/NN ./.	loaded/VBN into/IN and/CC run/VBN from/IN memory/NN ./.
Data/NNS is/VBZ both/DT written/VBN to/TO and/CC **read/VB** from/IN this/DT memory/NN ./.	Data/NNS is/VBZ both/DT written/VBN to/TO and/CC read/VBN from/IN this/DT memory/NN ./.
Input/NN devices/NNS **accept/VB** data/NNS and/CC instructions/NNS from/IN the/DT user/NN or/CC from/IN another/DT computer/NN system/NN (/(such/JJ as/IN a/DT computer/NN on/IN the/DT Internet/NNP)/) ./.	Input/NN devices/NNS accept/VBP data/NNS and/CC instructions/NNS from/IN the/DT user/NN or/CC from/IN another/DT computer/NN system/NN (/(such/JJ as/IN a/DT computer/NN on/IN the/DT Internet/NNP)/) ./.
A/DT microphone/NN or/CC **CD/NN** player/NN **attached/VBN** to/TO the/DT computer/NN enables/VBZ you/PRP to/TO add/VB the/DT sound/NN of/IN a/DT voice/NN or/CC a/DT music/NN selection/NN ./.	A/DT microphone/NN or/CC CD/NNP player/NN attached/VBD to/TO the/DT computer/NN enables/VBZ you/PRP to/TO add/VB the/DT sound/NN of/IN a/DT voice/NN or/CC a/DT music/NN selection/NN ./.
In/IN addition/NN ,/, there/EX is/VBZ also/RB a/DT **diskette/JJ** drive/NN ,/, which/WDT allows/VBZ you/PRP to/TO use/VB removable/JJ diskettes/NNS ./.	In/IN addition/NN ,/, there/EX is/VBZ also/RB a/DT diskette/NN drive/NN ,/, which/WDT allows/VBZ you/PRP to/TO use/VB removable/JJ diskettes/NNS ./.
The/DT name/NN implies/VBZ that/IN you/PRP can/MD not/RB change/VB the/DT information/NN on/IN the/DT disk/NN ,/, just/RB as/IN you/PRP can/MD not/RB record/VB over/IN an/DT **audio/JJ CD/NN** ./.	The/DT name/NN implies/VBZ that/IN you/PRP can/MD not/RB change/VB the/DT information/NN on/IN the/DT disk/NN ,/, just/RB as/IN you/PRP can/MD not/RB record/VB over/IN an/DT audio/NN CD/NNP ./.
Next/JJ ,/, the/DT computer/NN looks/VBZ for/IN an/DT operating/NN system/NN ,/, which/WDT is/VBZ usually/RB stored/VBN on/IN the/DT hard/JJ disk/NN ./.	Next/RB ,/, the/DT computer/NN looks/VBZ for/IN an/DT operating/NN system/NN ,/, which/WDT is/VBZ usually/RB stored/VBN on/IN the/DT hard/JJ disk/NN ./.

Phụ lục F: Kết quả gán nhãn từ loại trong mô hình kết hợp có dùng thông tin tiếng Việt

Kết quả gán nhãn từ loại không dùng thông tin tiếngViệt	Kết quả gán nhãn từ loại có dùng thông tin tiếng Việt
Note/NN ,/, however/RB ,/, that/WDT in/IN newer/JJR personal/JJ computers/NNS ,/, some/DT devices/NNS are/VBP built/VBN directly/RB onto/IN the/DT motherboard/NN instead/RB of/IN attaching/VBG to/TO it/PRP as/IN a/DT separate/JJ circuit/NN board/NN ./.	Note/VB ,/, however/RB ,/, that/WDT in/IN newer/JJR personal/JJ computers/NNS ,/, some/DT devices/NNS are/VBP built/VBN directly/RB onto/IN the/DT motherboard/NN instead/RB of/IN attaching/VBG to/TO it/PRP as/IN a/DT separate/JJ circuit/NN board/NN ./.
Even/RB if/IN a/DT computer/NN can/MD do/VBP its/PRP$ job/NN without/IN a/DT person/NN sitting/VBG in/IN front/NN of/IN it/PRP ,/, people/NNS still/RB design/VBP ,/, build/VBP ,/, **program/NN** ,/, and/CC **repair/NN** computer/NN systems/NNS ./.	Even/RB if/IN a/DT computer/NN can/MD do/VBP its/PRP$ job/NN without/IN a/DT person/NN sitting/VBG in/IN front/NN of/IN it/PRP ,/, people/NNS still/RB design/VBP ,/, build/VBP ,/, program/VBP ,/, and/CC repair/VBP computer/NN systems/NNS ./.
For/IN example/NN ,/, a/DT computer/NN document/NN can/MD be/VB a/DT text/NN file/NN (/(such/JJ as/IN a/DT letter/NN)/) ,/, a/DT group/NN of/IN numbers/NNS (/(such/JJ as/IN a/DT budget/NN)/) ,/, a/DT video/NN clip/NN (/(which/WDT includes/VBZ images/NNS and/CC **sounds/VBZ**)/) ,/, or/CC any/DT combination/NN of/IN these/DT items/NNS ./.	For/IN example/NN ,/, a/DT computer/NN document/NN can/MD be/VB a/DT text/NN file/NN (/(such/JJ as/IN a/DT letter/NN)/) ,/, a/DT group/NN of/IN numbers/NNS (/(such/JJ as/IN a/DT budget/NN)/) ,/, a/DT video/NN clip/NN (/(which/WDT includes/VBZ images/NNS and/CC sounds/NNS)/) ,/, or/CC any/DT combination/NN of/IN these/DT items/NNS ./.
A/DT scanner/NN can/MD copy/VB a/DT printed/VBN page/NN of/IN text/NN or/CC a/DT **graphic/JJ** into/IN the/DT computer/NN 's/POS memory/NN ,/, eliminating/VBG the/DT time/NN -/: consuming/NN step/NN of/IN typing/VBG input/NN or/CC creating/VBG an/DT image/NN	A/DT scanner/NN can/MD copy/VB a/DT printed/VBN page/NN of/IN text/NN or/CC a/DT graphic/NN into/IN the/DT computer/NN 's/POS memory/NN ,/, eliminating/VBG the/DT time/NN -/: consuming/NN step/NN of/IN typing/VBG input/NN or/CC creating/VBG an/DT image/NN

from/IN scratch/NN ./.	from/IN scratch/NN ./.
This/DT can/MD also/RB lead/VB to/TO **career/NN** advancement/NNP opportunities/NNS ./.	This/DT can/MD also/RB lead/VB to/TO career/VB advancement/NNP opportunities/NNS ./.
Name/NN and/CC describe/VB three/CD types/NNS of/IN storage/NN devices/NNS ./.	Name/VB and/CC describe/VB three/CD types/NNS of/IN storage/NN devices/NNS ./.
Any/DT computer/NN -/: regardless/RB of/IN its/PRP$ type/NN -/: is/VBZ controlled/VBN by/IN programmed/JJ instructions/NNS ,/, which/WDT **give/VB** the/DT machine/NN a/DT purpose/NN and/CC **tell/VB** it/PRP what/WP to/TO do/VB ./.	Any/DT computer/NN -/: regardless/RB of/IN its/PRP$ type/NN -/: is/VBZ controlled/VBN by/IN programmed/JJ instructions/NNS ,/, which/WDT **give/VB** the/DT machine/NN a/DT purpose/NN and/CC **tell/VB** it/PRP what/WP to/TO do/VB ./.
Some/DT programs/NNS exist/VBP primarily/RB for/IN the/DT computer/NN 's/POS use/NN and/CC help/VB the/DT computer/NN perform/VB and/CC manage/VB its/PRP$ own/JJ tasks/NNS ./.	Some/DT programs/NNS exist/VBP primarily/RB for/IN the/DT computer/NN 's/POS use/NN and/CC help/VB the/DT computer/NN perform/VB and/CC manage/VB its/PRP$ own/JJ tasks/NNS ./.
The/DT computer/NN manipulates/VBZ data/NNS according/VBG to/TO the/DT instructions/NNS contained/VBN in/IN the/DT software/NN and/CC then/RB forwards/RB it/PRP for/IN **use/VB** by/IN people/NNS or/CC another/DT computer/NN ./.	The/DT computer/NN manipulates/VBZ data/NNS according/VBG to/TO the/DT instructions/NNS contained/VBN in/IN the/DT software/NN and/CC then/RB forwards/RB it/PRP for/IN use/NN by/IN people/NNS or/CC another/DT computer/NN ./.
Early/JJ PC/NN microprocessors/NNS were/VBD not/RB **much/JJ** larger/JJR than/IN a/DT thumbnail/NN ./.	Early/JJ PC/NN microprocessors/NNS were/VBD not/RB much/RB larger/JJR than/IN a/DT thumbnail/NN ./.
Perhaps/RB the/DT most/RBS important/JJ thing/NN to/TO remember/VB about/IN RAM/NNP is/VBZ that/IN it/PRP is/VBZ volatile/JJ ,/, so/CC it/PRP needs/VBZ a/DT constant/JJ **supply/VB** of/IN power/NN ./.	Perhaps/RB the/DT most/RBS important/JJ thing/NN to/TO remember/VB about/IN RAM/NNP is/VBZ that/IN it/PRP is/VBZ volatile/JJ ,/, so/CC it/PRP needs/VBZ a/DT constant/JJ supply/NN of/IN power/NN ./.
They/PRP could/MD not/RB receive/VB instructions/NNS or/CC deliver/VB the/DT results/NNS of/IN	They/PRP could/MD not/RB receive/VB instructions/NNS or/CC deliver/VB the/DT results/NNS of/IN

their/PRP$ **work/VB** ./.	their/PRP$ work/NN ./.
One/CD example/NN is/VBZ the/DT touch/NN screen/NN ,/, a/DT type/NN of/IN monitor/NN that/WDT displays/VBZ text/NN or/CC **icons/VBZ** you/PRP can/MD touch/VB ./.	One/CD example/NN is/VBZ the/DT touch/NN screen/NN ,/, a/DT type/NN of/IN monitor/NN that/WDT displays/VBZ text/NN or/CC icons/NNS you/PRP can/MD touch/VB ./.
If/IN you/PRP **make/VB** **changes/VBZ** to/TO data/NNS while/IN working/VBG on/IN it/PRP ,/, the/DT changed/JJ data/NNS replaces/VBZ the/DT original/JJ data/NNS in/IN the/DT file/NN cabinet/NN (/(unless/IN you/PRP put/VBP it/PRP in/IN a/DT different/JJ place/NN in/IN storage/NN)/) ./.	If/IN you/PRP **make/VB** changes/NNS to/TO data/NNS while/IN working/VBG on/IN it/PRP ,/, the/DT changed/JJ data/NNS replaces/VBZ the/DT original/JJ data/NNS in/IN the/DT file/NN cabinet/NN (/(unless/IN you/PRP put/VBP it/PRP in/IN a/DT different/JJ place/NN in/IN storage/NN)/) ./.

Tài liệu tham khảo.

[1]. Eric Brill (1993). *A Corpus-based approach to Language Learning.* Luận án tiến sĩ, Đại học Pennsylvania, Hoa Kỳ.

[2]. Radu Floarian, Grace Ngai (2001). *Fast Transformation-based Learning Toolkit*. Đại học Johns Hopkins, 9/2001.

[3]. Radu Florian, Grace Ngai (2001). *Transformation-based learning in the fast lane.* Proceedings of North American ACL-2001.

[4]. Samuel K. (1998). *Lazy Transformation-based learning.* Proceedings of the 11th International Floarida AI Research Symposium Conference, Florida, Hoa Kỳ.

[5]. Helmut Schmid(1993). *Part of Speech Tagging with Nearal Networks*, Proceedings of the International Conference on Computaional Linguistics,Kyoto, Japan,8/994.

[6]. Đinh Điền, Nguyễn Văn Toàn, Diệp Chí Cường, *"Gán nhãn từ loại tiếng Việt tự động"*, Kỷ yếu hội nghị Khoa học lần 3, ĐH Khoa học Tự nhiên – ĐHQG-TPHCM.

[7]. Adwait Ratnapark (1996), A Maximum Entropy model for POS Tagging,

[8]. Helmut Schmid (1993), *Probabilistics POS Tagging using Decision Trees*,

[9]. Đinh Điền (2002*). Bước đầu xây dựng kho ngữ liệu song ngữ Anh-Việt điện tử.* Luận văn thạc sĩ ngôn ngữ học so sánh, ĐH Khoa học Xã hội & Nhân văn, ĐH Quốc Gia TP.HCM

[10]. Đinh Điền (2003). *Mô hình học luật chuyển đổi từ ngữ liệu song ngữ cho Hệ dịch tự động Anh-Việt*. Luận án Tiến sĩ Tin học, Đại học Quốc gia Tp.HCM

[11]. Sampson (1995) *English for computer: The SUSANNE Corpus and Analytic Scheme,*Claredon Press (Oxford University Press).

[12]. Dien Dinh and Kiem Hoang (2002), "Bilingual corpus and word sense disambiguation in the English-to-Vietnamese Machine Translation", *Proceedings of APIS-02*, Bangkok, Thailand.

[13]. Hans van Halteren,WaterDaelemans and Jakub Zavrel(2001), *Improving Accuracy in World Class Tagging through the Combination of Machine Systems Association for Computational Linguistic,Netherlands.*