ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Nguyễn Trung Kiên

PHÂN ĐOẠN TỪ TIẾNG VIỆT SỬ DỤNG MÔ HÌNH CRFs

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUI

Ngành: Công nghệ thông tin

HÀ NỘI - 2006

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Nguyễn Trung kiên

PHÂN ĐOẠN TỪ TIẾNG VIỆT SỬ DỤNG MÔ HÌNH CRFs

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUI
Ngành: Công nghệ thông tin

Cán bộ hướng dẫn: TS. Hà Quang Thụy
Cán bộ đồng hướng dẫn: TS. Nguyễn Lê Minh

HÀ NỘI - 2006

Lời cảm ơn

Trước tiên, em muốn gửi lời cảm ơn sâu sắc nhất đến thầy giáo, TS. Hà Quang Thụy, TS. Nguyễn Lê Minh, ThS. Phan Xuân Hiếu và CN. Nguyễn Cẩm Tú, CN. Nguyễn Việt Cường, những người đã tận tình hướng dẫn em trong suốt quá trình nghiên cứu Khoa học và làm khóa luận tốt nghiệp.

Em xin bày tỏ lời cảm ơn sâu sắc đến những thầy cô giáo đã giảng dạy em trong bốn năm qua, những kiến thức mà em nhận được trên giảng đường đại học sẽ là hành trang giúp em vững bước trong tương lai.

Em cũng muốn gửi lời cảm ơn đến các anh chị và các thầy cô trong nhóm seminar về "Khai phá dữ liệu" đã cho em những lời khuyên bổ ích về chuyên môn trong quá trình nghiên cứu.

Cuối cùng, em muốn gửi lời cảm ơn sâu sắc đến tất cả bạn bè, và đặc biệt là cha mẹ và chị gái, những người luôn kịp thời động viên và giúp đỡ em vượt qua những khó khăn trong cuộc sống.

Sinh viên

Nguyễn Trung Kiên

Tóm tắt

Phân đoạn từ là một bước cơ bản trong trích chọn thông tin từ văn bản và xử lý ngôn ngữ tự nhiên. Trong tiếng Việt, bài toán phân đoạn từ có thể được dùng cho các máy tìm kiếm tiếng Việt, dịch tự động, kiểm tra chính tả tiếng Việt…Hiện nay bài toán phân đoạn từ tiếng Việt đang được nghiên cứu, triển khai bởi rất nhiều cá nhân, tổ chức trong và ngoài nước.

Trong khóa luận này, em xin trình bày về một giải pháp cho bài toán phân đoạn từ tiếng Việt. Sau khi tìm hiểu về đặc điểm từ vựng tiếng Việt, xem xét các phương pháp phân đoạn từ tiếng Việt hiện nay, em đã chọn phương pháp tiếp cận học máy bằng cách xây dựng một hệ thống phân đoạn từ tiếng Việt dựa trên mô hình Conditional random fields (CRFs - Laferty, 2001). Ưu điểm của mô hình này là nó rất mạnh trong xử lý dữ liệu dạng chuỗi, với khả năng tính hợp rất nhiều các đặc điểm khác nhau rút ra từ tập dữ liệu, hỗ trợ rất tốt cho bài toán phân đoạn từ. Kết quả thử nghiệm trên các văn

Mục lục

Bảng từ viết tắt

Từ hoặc cụm từ	Viết tắt
Conditional Random Field	CRF
Mô hình Markov cực đại hóa entropy	MEMM
Limited-memory Broyden-Fletcher-Goldfarb-Shanno	L-BFGS

Lời nói đầu

Trong những năm gần đây, cùng với sự bùng nổ thông tin toàn cầu, thì lượng thông tin trên văn bản và web tiếng Việt cũng tăng lên nhanh chóng. Đây quả thực là một nguồn thông tin đầy tiềm năng cần được khai thác. Nếu chúng ta có thể sử dụng chúng để xây dựng một cơ sơ tri thức tiếng Việt thì ta sẽ có một cơ sở tri thức rất có giá trị. Song việc đó tới nay vẫn còn là một thách thức.

Trong nỗ lực xây dựng một cơ sở tri thức tiếng Việt thì việc hiểu các văn bản tiếng Việt, tóm tắt văn bản tiếng Việt, hay phân loại văn bản tiếng Việt…là những công việc không thể thiếu được. Chính vì lý do đó, Bộ Khoa học - Công nghệ đã phê duyệt đề tài cấp nhà nước với tên gọi "Nghiên cứu phát triển một số sản phẩm thiết yếu về xử lý tiếng nói và văn bản tiếng Việt" năm 2006. Một dạng điển hình về kết quả của đề tài là các công cụ cơ bản dùng để xử lý văn bản (tiếng Việt) như kiểm lỗi chính tả, phân tách từ, xác định loại từ, phân tích cú pháp... Công việc cơ bản đầu tiên có tính tiên quyết là phân đoạn từ tiếng Việt.

Ý thức được những lợi ích của việc xây dựng cơ sở tri thức tiếng Việt nói chung và bài toán phân đoạn từ tiếng Việt nói riêng, em đã chọn hướng nghiên cứu trong khóa luận của mình là xây dựng một hệ thống phân đoạn từ tiếng Việt

Bài toán phân đoạn từ tiếng Việt

Ta có thể hiểu đơn giản bài toán phân đoạn từ tiếng Việt là cho trước một văn bản tiếng Việt, ta cần xác định trong văn bản đó ranh giới giữa các từ trong câu. Nhưng khác với một số tiếng nước ngoài như tiếng Anh, thì trong tiếng Việt ranh giới giữa các từ nhiều trường hợp không phải là dấu cách trống. Ví dụ, trong câu nói "phân đoạn từ tiếng Việt là một bài toán quan trọng", chúng ta có thể thấy dấu cách trống không phải là dấu hiệu để nhận ra ranh giới của các từ.

Mục tiêu của khóa luận

Trong khóa luận này, mục tiêu chính là đưa ra được một hệ thống phân đoạn từ với độ chính xác cao. Hệ thống phải thể hiện được những ưu điểm so với các phương pháp đã có hiện nay và có thể đưa vào ứng dụng được, nhằm vào mục tiêu xây dựng cơ sở tri thức tiếng Việt.

Để làm được điều đó, trước hết ta cần xây dựng được bộ convert dữ liệu về dạng chuẩn phục vụ việc học máy. Đó là một chuỗi các quá trình xử lý dữ liệu: từ việc ghi lại từ internet và các nguồn khác, trính rút nội dung chính, phân đoạn từ bán tự động, đến việc chuyển dữ liệu đã xử lý về dạng chuẩn **iob2**.

Tiếp theo mục tiêu của khóa luận là phải đưa ra được các lựa chọn thuộc tính tốt nhất cho học máy. Đó là việc áp dụng mô hình CRFs với những đặc điểm riêng của tiếng Việt, và nó hoàn toàn khác với các mô hình đã có trong tiếng Anh, tiếng Trung, Thái Lan…

Ý nghĩa và đóng góp của khóa luận

Trong khóa luận này đã đưa ra một hướng tiếp cận mới cho bài toán phân đoạn từ tiếng Việt. Và đây sẽ là nên tảng cho các phương pháp sau này. Ta có thể tiếp tục phát triển, cải tiến những kết quả của khóa luận. Ngoài ra kết quả của khóa luận này có thể được dùng để so sánh với các phương pháp khác để thấy được tính vượt trội của mỗi phương pháp.

Cũng trong khóa luận này, em đã xây dựng một bộ dữ liệu chuẩn khá phong phú. Dữ liệu này không chỉ được dùng trong khóa luận mà nó có thể được các nhóm nghiên cứu khác tận dụng nhằm tăng đán kể lượng dữ liệu dùng cho học máy.

Hơn nữa, khi ta xây dựng được một hệ thống phân đoạn tiếng Việt tốt thì nó có thể được để hỗ trợ trong nhiều lĩnh vực khác như

- Hỗ trợ máy tìm kiếm tiếng Việt: các máy tìm kiếm thường phải xác định các từ quan trọng trong một văn bản. Việc phân đoạn đúng một từ tiếng Việt sẽ giúp máy tìm kiểm trả lại các kết quả chính xác cho người dùng.

- Xử lý ngôn ngữ tự nhiên, ví dụ như dịch tự động. Chúng ta đã biết từ là một đơn vị cơ bản trong xử lý ngôn ngữ tự nhiên, thế nên việc phân đoạn từ luôn là bước đầu tiên trong xử lý. Trong dịch tự động, chúng ta cần phải xác định ranh giới các từ trong văn bản cần dịch, từ đó mới có thể tiến hành các xử lý cần thiết để dịch sang ngôn ngữ khác.

- Kiểm tra chính tả tiếng Việt: việc kiểm tra chính tả phải bắt đầu bằng việc xác định giới hạn đâu là một từ để có thể đưa ra những đánh giá chính xác một từ là đúng hay sai chính tả trong văn cảnh cụ thể.

Cấu trúc của khóa luận

Trong khóa luận, em trình những tìm hiểu của mình về bài toán này và đưa ra một phương pháp để xây dựng hệ thống phân đoạn từ tiếng Việt

Chương 1. Phân đoạn từ tiếng Việt : trình bày những đặc điểm riêng của tiếng Việt khác với các ngôn ngữ khác. Các phương pháp phân đoạn từ hiện nay sẽ được trình bày và đánh giá, từ đó chọn ra một hướng tiếp cận của khóa luận

Chương 2. Conditional Random Fields : trình bày cơ bản về mô hình Conditional Random Field, một mô hình học máy rất mạnh trong việc phân đoạn và gán nhãn dữ liệu dạng chuỗi.

Chương 3. Phân đoạn từ tiếng Việt với CRFs: Trong chương này, bài toán phân đoạn từ tiếng Việt sẽ được mô tả chi tiếng theo hướng áp dụng mô hình CRFs. Việc lựa chọn thuộc tính cũng sẽ được trình bày cụ thể và đề cập tới cách đánh giá mô hình.

Chương 4. Thử nghiệm và đánh giá: trình bày môi trường thực nghiệm và các kết quả đã đạt được. Các phân tính, đánh giá kết quả đó sẽ cũng sẽ được đưa ra trong chương này.

Phần kết luận tổng kết các công việc đã làm được trong khóa luận và phương hướng nghiên cứu trong tương lại của em

Chương 1. Phân đoạn từ tiếng Việt

Hiện nay có khá nhiều phương pháp khác nhau để tiếp cận bài toán phân đoạn từ tiếng Việt. trong chương này sẽ giới thiệu một số phương pháp như vậy cùng với nhưng đánh giá về ưu điểm và nhược điểm của chúng và lý do tại sao em lại chọn hướng tiếp cận dưa trên mô hình CRFs. Nhưng trước hết, em xin trình bày về những tìm hiểu về tiếng Việt, đó sẽ là cơ sở để tìm ra một phương pháp hợp lý nhất cho bài toán phân đoạn từ.

1.1 Từ vựng tiếng Việt

Việc chỉ ra định nghĩa chính xác suất thế nào là một từ không phải đơn giản, đòi hỏi công sức nghiên cứu của các nhà ngôn ngữ học. Chúng ta giới thiệu một định nghĩa sau làm ví dụ về định nghĩa từ:

"Từ là đơn vị nhỏ nhất có nghĩa, có kết cấu vỏ ngữ âm bền vững, hoàn chỉnh, có chức năng gọi tên, được vận dụng độc lập, tái hiện tự do trong lời nói để tạo câu".[1]

Nhưng xét trên góc độ ứng dụng, ta có thể hiểu một các đơn giản là "từ được cấu tạo bởi một hoặc nhiều tiếng". Chúng ta tìm hiểu về khái niệm "tiếng" trong mục nhỏ ngay tiếp theo.

1.1.1 Tiếng – đơn vị cấu tạo lên từ

1.1.1.1 Khái niệm

Tiếng là đơn vị cơ sở để cấu tạo lên từ tiếng Việt. Về mặt hình thức, tiếng là một đoạn phát âm của người nói, dù chúng ta có cố tình phát âm chậm đến mấy cũng không thể tách tiếng ra thành các đơn vị khác được. Tiếng được các nhà ngôn ngữ gọi là âm tiết (syllable). Về mặt nội dung, tiếng là đơn vị nhỏ nhất có nội dung được thể hiện, chí ít tiếng cũng có giá trị về mặt hình thái học (cấu tạo từ), đôi khi người ta gọi tiếng là hình tiết (morphemesyllable), tức là âm tiết có có giá trị về hình thái học.

1.1.1.2 Phân loại

Các tiếng không phải tất cả đều giống nhau, xét về mặt ý nghĩa, chúng ta có thể chia tiếng thành các loại sau

- Tiếng tự thân nó đã có ý nghĩa, thường được quy chiếu vào một đối tượng, khái niệm. Ví dụ: trời, đất, nước, cây, cỏ…
- Tiếng tự thân nó không có ý nghĩa, chúng không được quy chiếu vào đối tượng, khái niệm nào cả. Chúng thường đi cùng với một tiếng khác có nghĩa và làm thay đổi sắc thái của tiếng đó, ví dụ như: (xanh) lè, (đường) xá, (năng) nôi…
- Tiếng tự thân nó không có ý nghĩa nhưng lại đi với nhau để tạo thành từ. Những nếu tách rời tiếng này ra đứng riêng thì chúng không có nghĩa gì cả, nhưng lại có thể ghép lại thành từ có nghĩa. Ta thường xuyên gặp ở những từ mượn như phéc-mơ-tuya, a-pa-tít, mì-chính...

Trong tiếng Việt thì các tiếng thuộc nhóm đầu tiên chiếm đa số. Các tiếng thuộc hai nhóm sau thưởng chỉ chiếm số ít, đặc biệt là nhóm thứ 3, chúng thường được gọi là tiếng vô nghĩa. Việc nhóm đầu tiên chiếm đa số phản ánh thực tế là khi nói, người ta thường sử dụng các tiếng có nghĩa, hiếm khi lại nói ra toàn từ vô nghĩa.

1.1.1.3 Mô hình tiếng trong tiếng Việt và các thành tố của nó

Ta có thể biểu diễn cấu trúc của tiếng như bảng sau [4]:

Bảng 1: cấu trúc của tiếng trong tiếng Việt

Âm đầu	Thanh điệu		
	Vần		
	Âm đệm	Âm chính	Âm cuối

- Thanh điệu: mỗi tiếng đều có một thanh điệu là một trong 6 loại sau: sắc, huyền, hỏi, ngã, nặng, và thanh bằng. Chúng có tác dụng phân biệt tiếng về cao độ. Ví dụ : "việt" và "viết"
- Âm đầu: có tác dụng mở đầu âm tiết. Ví dụ: "nắng" và "mắng"
- Âm đệm: Có tác dụng biến đổi âm sắc của âm tiết sau lúc mở đầu. Ví dụ: t**o**án – tán
- Âm chính: là hạt nhân và mang âm sắc chủ đạo của tiếng. Ví dụ : "tú**y**" và "t**ú**i"

- Âm cuối: có tác dụng kết thúc tiếng với các âm sắc khác nhau, do đó có thể phân biệt các tiếng. Ví dụ: "bà**n**" và "bà**i**"
- Cụm gồm âm đệm, âm chính và âm cuối ta gọi là vần. Ví dụ: vần "ang", vần "oan"…

Đây là 5 thành tố của tiếng (vần không phải là một thành tố mà chỉ là cách gọi của cụm 3 âm đã nói ở trên), mà bất cứ tiếng nào trong tiếng Việt đều tuân theo cấu trúc như trên. Nhưng cũng có trường hợp một số âm trùng nhau, nhất là với những tiếng gồm 3 kí tự trở xuống.

1.1.2 Cấu tạo từ

Như đã đề cập ở trên, từ trong tiếng Việt được cấu tạo hoặc là bằng một tiếng hoặc là tổ hợp nhiều tiếng theo các cách khác nhau để tạo ra các loại từ. Dưới đây, em xin trình bày về hai loại từ tiếng Việt

1.1.2.1 Từ đơn

Từ đơn, hay còn gọi là từ đơn âm tiết, là các từ được cấu tạo bởi một tiếng duy nhất. Ví dụ: tôi, bạn, nhà, hoa, vườn…

1.1.2.2 Từ ghép

Từ ghép là các từ được tạo lên từ hai hoặc nhiều hơn các tiếng lại. Giữa các tiếng có mỗi quan hệ về nghĩa với nhau, vì thế ta cũng có các loại từ ghép khác nhau.

- Từ ghép đẳng lập: các thành phần cấu tạo từ có mối quan hệ bình đẳng với nhau về nghĩa. Ví dụ: ăn nói, bơi lội …
- Từ ghép chính phụ: các thành phần cấu tạo từ có mối quan hệ phụ thuộc với nhau về nghĩa. Thành phần phụ sẽ có vai trò làm chuyên biệt hóa, tạo sắc thái cho thành phần chính. Ví dụ: hoa hồng, đường sắt…

1.1.2.3 Từ láy

Một từ sẽ được coi là từ láy khi các yếu tố cấu tạo nên nó có thành phần ngữ âm được lặp lại; nhưng vừa có lặp (còn gọi là điệp) vừa có biến đổi (còn gọi là đối). Ví dụ: đo

đỏ, man mát… Nếu một từ chỉ có phần lặp mà không có sự biến đổi (chẳng hạn như từ nhà nhà, ngành ngành…) thì ta có dạng láy của từ, hoàn toàn không phải là từ láy.

Độ dài từ láy thay đổi từ 2 tiếng đến 4 tiếng. Nhưng trong tiếng Việt đa số là từ láy hai tiếng., chúng chia thành hai loại từ láy sau:

- Láy hoàn toàn: là cách láy mà tiếng sau lặp lại hoàn toàn tiếng trước. Gọi là hoàn toàn nhưng thực ra các tiếng không trùng khít nhau mà có những sai khác rất nhỏ mà ta có thể nhận ra ngay. Một số kiểu láy hoàn toàn ta hay gặp
    - Láy hoàn toàn đối nhau ở thanh điệu, ví dụ như: "sừng sững", "loang loáng"…
    - Láy hoàn toàn đối nhau ở âm cuối, ví dụ như: "khin khít", "ăm ắp"…
    - Láy hoàn toàn đối nhau ở trọng âm, tức là một tiếng được nói nhấn mạnh hoặc kéo dài hơn so với tiếng kia, ví dụ như: đùng đùng, đăm đăm…
- Láy bộ phận: là cách láy mà chỉ có điệp ở phần âm đầu của tiếng, hoặc điệp ở phần vần thì được gọi là láy bộ phận. Căn cứ vào đó ta chia ra từng kiểu láy sau
    - Từ láy điệp ở âm đầu và đối ở vần, ví dụ như "nhưng nhức", "thơ thẩn",…
    - Từ láy điệm ở vần và đối ở âm đầu, ví dụ "hấp tấp", "liểng xiểng",…

1.1.3 Nhập nhằng

Nếu ta dựa trên khái niệm "từ" của các nhà ngôn ngữ học để trực tiếp phân đoạn từ bằng tay thì khó có thể xảy ra việc nhập nhằng trong tiếng Việt. Song dưới góc độ ứng dụng máy tính, chúng ta coi một từ chỉ đơn giản là cấu tạo từ một hoặc nhiều tiếng, và việc này rất dễ gây ra sự nhập nhằng trong quá trình phân đoạn từ.

Sự nhập nhằng của tiếng Việt có thể chia thành 2 kiểu sau [21]:

- Nhập nhằng chồng chéo: chuỗi "abc" được gọi là nhặp nhằng chồng chéo nếu như từ "ab", "bc" đều xuất hiện trong từ điển tiếng Việt. Ví dụ như

trong câu "ông già đi nhanh quá" thì chuỗi "ông già đi" bị nhập nhằng chồng chéo vì các từ "ông già" và "già đi" đều có trong từ điển.

- Nhập nhằng kết hợp: chuỗi "abc" được gọi là nhập nhằng kết hợp nếu như từ "a","b","ab" đều xuất hiện trong từ điển tiếng Việt. Ví dụ như trong câu "Bàn là này còn rất mới" thì chuỗi "bàn là" bị nhập nhằng kết hợp, do các từ "bàn", "là", "bàn là" đều có trong từ điển.

1.2 Phân đoạn từ tiếng Việt bằng máy tính

Trước hết chúng ta cần làm rõ sự khác nhau giữa phân đoạn từ tiếng Việt bằng máy tính và bằng thủ công. Nếu chúng ta làm thủ công, thì độ chính xác rất cao, gần như tuyệt đối. Song như đã nói ở chương đầu, phân đoạn từ là một công đoạn đầu của rất nhiều quá trình xử lý ngôn ngữ tự nhiên bằng máy tính nên việc phân đoạn từ bằng máy tính là rất quan trọng. Hơn nữa, khi mà khối lượng dữ liệu rất lớn thì việc phân đoạn từ bằng máy tính gần như là lựa chọn duy nhất.

Hiện đã có nhiều công trình nghiên cứu xây dựng mô hình phân đoạn từ tiếng Việt bằng máy tính. Đa số là các mô hình mà đã được áp dụng thành công cho các ngôn ngữ khác như tiếng Anh, tiếng Trung, tiếng Nhật…và được cải tiến để phù hợp với đặc điểm của tiếng Việt. Vấn đề mà tất cả mô hình phân đoạn từ tiếng Việt gặp phải đó là

- Nhập nhằng
- Xác định từ các từ chưa biết trước (đối với máy tính) như các câu thành ngữ, từ láy, hoặc tên người, địa điểm, tên các tổ chức…

Việc giải quyết tốt hay không hai vấn đề trên có thể quyết định một mô hình phân đoạn nào đó là tốt hay không.

1.2.1 Phương pháp Maximum Matching

Phương pháp này còn được gọi là phương pháp khớp tối đa. Tư tưởng của phương pháp này là duyệt một câu từ trái qua phải và chọn từ có nhiều tiếng nhất mà có mặt trong từ điển tiếng Việt. Nôi dung thuật toán này dựa trên thuật toán đã được Chih-Hao Tsai[8] giới thiệu năm 1996. Thuật toán có 2 dạng sau:

Dạng đơn giản: Giả sử có một chuỗi các tiếng trong câu là t_1, t_2, ..t_N. Thuật toán sẽ kiểm tra xem t_1 có mặt trong từ điển hay không, sau đó kiểm tra tiếp t_1-t_2 có trong từ điển hay không. Tiếp tục như vậy cho đến khi tìm được từ có nhiều tiếng nhất có mặt trong từ điển, và đánh dấu từ đó. Sau đó tiếp tục quá trình trên với tất các các tiếng còn lại trong câu và trong toàn bộ văn bản. Dạng này khá đơn giản nhưng nó gặp phải rất nhiều nhập nhằng trong tiếng Việt, ví dụ nó sẽ gặp phải lỗi khi phân đoạn từ câu sau: "học sinh | học sinh | học", câu đúng phải là "học sinh| học| sinh học"

Dạng phức tạp: dạng này có thể tránh được một số nhập nhằng gặp phải trong dạng đơn giản. Đầu tiên thuật toán kiểm tra xem t_1 có mặt trong từ điển không, sau đó kiểm tra tiếp t_1-t_2 có mặt trong từ điển không. Nếu t_1-t_2 đều có mặt trong từ điển thì thuật toán thực hiện chiến thuật chọn 3-từ tốt nhất. Tiêu chuẩn 3-từ tốt nhất được Chen & Liu (1992) đưa ra như sau:

- Độ dài trung bình của 3 từ là lớn nhất. Ví dụ với chuỗi "cơ quan tài chính" sẽ được phân đoạn đúng thành "cơ quan | tài chính", tránh được việc phân đoạn sai thành "cơ | quan tài | chính" vì cách phân đúng phải có độ dài trung bình lớn nhất.
- Sự chênh lệch độ dài của 3 từ là ít nhất. Ví dụ với chuỗi "công nghiệp hóa chất phát triển" sẽ được phân đoạn đúng thành "công nghiệp | hóa chất | phát triển" thay vì phân đoạn sai thành "công nghiệp hóa | chất | phát triển". Cả 2 cách phần đoạn này đều có độ dài trung bình bằng nhau, nhưng cách phân đoạn đúng có sự chênh lệch độ dài 3 từ ít hơn.

Nhận xét:

Tuy hai tiêu chuẩn trên có thể hạn chế được một số nhập nhằng, nhưng không phải tất cả. Ví dụ với câu "ông già đi nhanh" thì cả 2 cách phân đoạn sau đều có cùng độ dài trung bình và độ chênh lệch giữa các từ: "ông | già đi| nhanh" và "ông già | đi | nhanh", do đó thuật toán không thể chỉ ra cách phân đúng được.

Ưu điểm của phương pháp trên có thể thấy rõ là đơn giản, dễ hiểu và chạy nhanh. Hơn nữa chúng ta chỉ cần một tập từ điển đầy đủ là có thể tiến hành phân đoạn các văn bản, hoàn toàn không phải trải qua huấn luyện như các phương pháp sẽ trình bày tiếp theo.

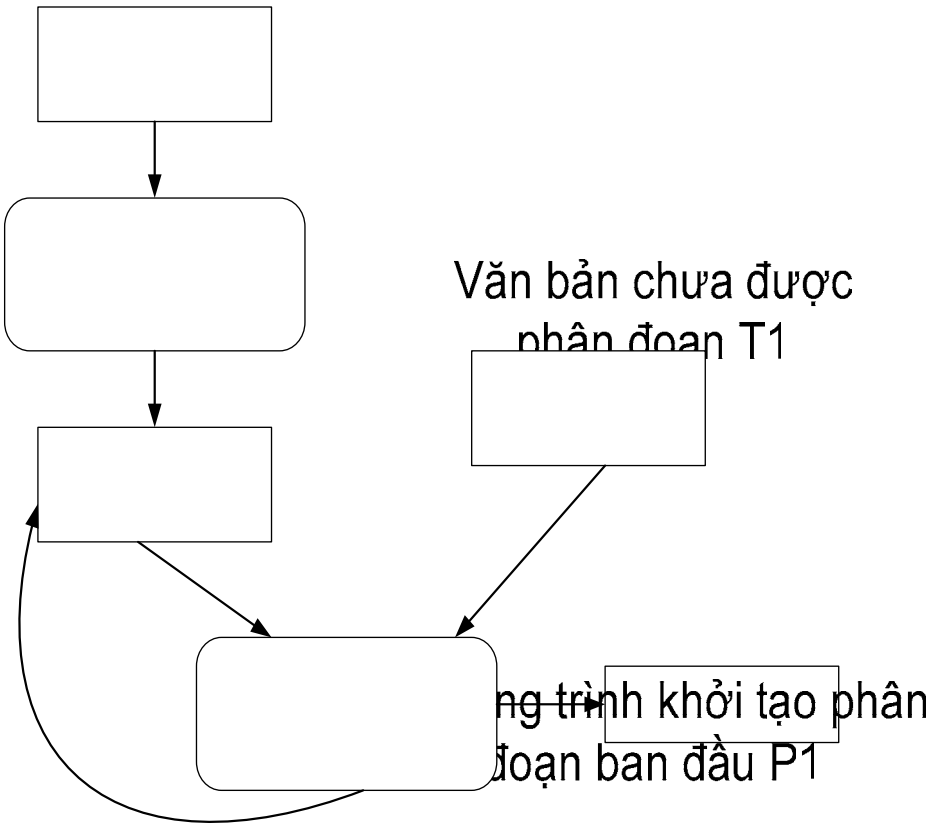
Nhược điểm của phương pháp này là nó không giải quyết được 2 vấn đề quan trọng nhất của bài toán phân đoạn từ tiếng Việt: thuật toán gặp phải nhiều nhập nhằng, hơn nữa nó hoàn toàn không có chiến lược gì với những từ chưa biết.

1.2.2 Phương pháp TBL

Phương pháp TBL (Transformation-Based Learning) còn gọi là phương pháp học cải tiến, được Eric Brill giới thiệu lần đầu vào năm 1992. Ý tưởng của phương pháp này áp dụng cho bài toán phân đoạn như sau

Đầu tiên văn bản chưa được phân đoạn T1 được phân tích thông qua chương trình khởi tạo phân đoạn ban đầu P1. Chương trình P1 có độ phức tạp tùy chọn, có thể chỉ là chương trình chú thích văn bản bằng cấu trúc ngẫu nhiên, hoặc phúc tạp hơn là phân đoạn văn bản một cách thủ công. Sau khi qua chương trình P1, ta được văn bản T2 đã được phân đoạn. Văn bản T2 được so sánh với văn bản đã được phân đoạn trước một cách chính xác là T3. Chương trình P2 sẽ thực hiện học từng phép chuyển đổi (transformation) để khi áp dụng thì T2 sẽ giống với văn bản chuẩn T3 hơn. Quá trình học được lặp đi lặp lại đến khi không còn phép chuyển đổi nào khi áp dụng làm cho T2 tốt hơn nữa. Kết quả ta thu được bộ luật R dùng cho phân đoạn.

Cách hoạt động của TBL có thể mô tả ở hình sau:



Hình 1: Mô hình hoạt động của TBL

Nhận xét

Phương pháp TBL có nhược điểm là mất rất nhiều thời gian học và tốn nhiều không gian nhớ do nó phải sinh ra các luật trung gian trong quá trình học. Vì để học được một bộ luật thì TBL chạy rất lâu và dùng tới nhiều bộ nhớ, thậm việc xây dựng được một bộ luật đầy đủ dùng cho phân đoạn từ là rất khó khăn. Vì thế khi áp dụng phương pháp này, sẽ có khá nhiều nhập nhằng.

Tuy nhiên sau khi có bộ luật thì TBL lại tiến hành phân đoạn khá nhanh. Hơn nữa, ý tưởng của phương pháp rút ra các quy luật từ ngôn ngữ và liên tục "sửa sai" cho luật thông qua quá trình lặp là phù hợp với bài toán xử lý ngôn ngữ tự nhiên.

1.2.3 Phương pháp WFST

Phương pháp WFST (Weighted Finite-State Transducer) [15] còn gọi là phương pháp chuyển dịch trạng thái hữu hạn có trọng số. Ý tưởng chính của phương pháp này áp dụng cho phân đoạn từ tiếng Việt là các từ sẽ được gán trọng số bằng xác suất xuất hiện

của từ đó trong dữ liệu. Sau đó duyệt qua các câu, cách duyệt có trọng số lớn nhất sẽ là cách dùng để phân đoạn từ. Hoạt động của WFST có thể chia thành ba bước sau:

- Xây dựng từ điển trọng số: từ điển trọng số D được xây dựng như là một đồ thị biến đổi trạng thái hữu hạn có trọng số. Giả sử
  - H là tập các tiếng trong tiếng Việt
  - P là tập các loại từ trong tiếng Việt.
  - Mỗi cung của D có thể là
    - Từ một phần tử của H tới môt phần tử của H
    - Từ phần tử ε (xâu rỗng) đến một phần tử của P
  - Mỗi từ trong D được biểu diễn bởi một chuỗi các cung bắt đầu bởi một cung tương ứng với một phần tử của H, kết thúc bởi một cung có trọng số tương ứng với một phần tử của $\varepsilon \times P$. Trọng số biểu diễn một chi phí ước lượng (estimated cost) cho bởi công thức
  - $C = -\log(\frac{f}{N})$ (1.1)

    Trong đó f: tần số xuất hiện của từ, N: kích thước tập mẫu
- Xây dựng các khả năng phân đoạn từ: bước này thống kê tất cả các khả năng phân đoạn của một câu. Giả sử câu có n tiếng, thì sẽ có 2^{n-1} cách phân đoạn khác nhau. Để giảm sự bùng nổ các cách phân đoạn, thuật toán sẽ loại bỏ ngay những nhánh phân đoạn mà chứa từ không xuất hiện trong từ điển.
- Lựa chọn khả năng phân đoạn tối ưu: sau khi liệt kê tất cả các khả năng phân đoạn từ, thuật toán sẽ chọn cách phân đoạn tốt nhất, đó là cách phân đoạn có trọng số bé nhất.

Ví dụ: câu "Tốc độ truyền thông tin sẽ tăng cao" (theo [9])

Từ điển trọng số:

"tốc độ"	8.68
"truyền"	12.31
"truyền thông"	1231

“thông tin” 7.24

“tin” 7.33

“sẽ” 6.09

“tăng” 7.43

“cao” 6.95

Trọng số theo mỗi cách phân đoạn được tính là

- “Tốc độ # truyền thông # tin # sẽ # tăng # cao.” = 8.68 +12.31 + 7.33 + 6.09 + 7.43 +6.95 = 48.79
- “Tốc độ # truyền # thông tin # sẽ # tăng # cao.”= 8.68 +12.31 + 7.24 + 6.09 + 7.43 +6.95 = 48.79

Do đó, ta có được phân đoạn tối ưu là cách phân đoạn sau “Tốc độ # truyền # thông tin # sẽ # tăng #cao.”

Nhận xét:

Nhược điểm chính của thuật toán là việc đánh trọng số dựa trên tần số xuất hiện của từ, nên khi tiến hành phân đoạn thì không tránh khỏi các nhập nhằng trong tiếng Việt. Hơn nữa với những văn bản dài thì phương pháp này còn gặp phải sự bùng nổ các khả năng phân đoạn của từng câu.

Ưu điểm của phương pháp này là sẽ cho độ chính xác cao nếu ta xây dựng được một dữ liệu học đầy đủ và chính xác. Nó còn có thể kết hợp với các phương pháp khử nhập nhằng (phương pháp mạng Neural) để cho kết quả phân đoạn rất cao.

1.3 Phương pháp tiếp cận của khóa luận

Sau khi tìm hiểu về ngôn ngữ tiếng Việt và một số phương pháp phân đoạn từ tiếng Việt bằng máy tính hiện nay, em nhận thấy một mô hình phân đoạn từ tiếng Việt tốt phải giải quyết được hai vấn đề chính đó là giải quyết nhập nhằng trong tiếng Việt và có khả năng phát hiện từ mới. Xuất phát từ đó, em chọn hướng tiếp cận sử dụng mô hình học máy CRF cho bài toán phân đoạn từ tiếng Việt. Đây là mô hình có khả năng tích hợp hàng triệu đặc điểm của dữ liệu huấn luyện cho quá trình học máy, nhờ đó có thể giảm thiểu nhập nhằng trong tiếng Việt. Hơn nữa ta có thể đưa vào rất nhiều đặc điểm cho học

máy để có khả năng phát hiện từ mới như tên riêng, từ láy…mà em sẽ trình bày cụ thể trong các chương tiếp theo.

1.4 Tổng kết chương

Chương này đã trình bày về từ vựng Tiếng Việt, chỉ ra những khó khăn đối với bài toán phân đoạn từ tiếng Việt và một số hướng tiếp cận giải quyết bài toán này cùng với những ưu và nhược điểm của chúng. Qua đó em chọn cách tiếp cận học máy sử dụng mô hình CRF. Trong chương tiếp theo, em sẽ trình bày cụ thể về mô hình CRF này.

Chương 2. Conditional Random Field

Trong khi giải quyết các vấn đề trên nhiều lĩnh vực khoa học, người ta thường bắt gặp các bài toán về phân đoạn và gán nhãn dữ liệu dạng chuỗi. Các mô hình xác suất phổ biến để giải quyết bài toán này là mô hình Markov ẩn (HMMs) và stochastic grammar. Trong sinh học, HMMs và stochastic grammars đã thành công trong việc sắp xếp các chuỗi sinh học, tìm kiếm chuỗi tương đồng với một quần thể tiến hóa cho trước, và phân tích cấu trúc RNA. Trong khoa học máy tính, HMMs và stochastic grammars được ứng dụng rộng rãi trong hàng loạt vấn đề về xử lý văn bản và tiếng nói, như là phân loại văn bản, trích chọn thông tin, phân loại từ [15].

HMMs và stochastic grammars là các mô hình sinh (generative models), tính toán xác suất joint trên cặp chuỗi quan sát và chuỗi trạng thái; các tham số thường được huấn luyện bằng cách làm cực đại độ đo likelihood của dữ liệu huấn luyện. Để tính được xác suất joint trên chuỗi quan sát và chuỗi trạng thái, các mô hình sinh cần phải liệt kê tất cả các trường hợp có thể có của chuỗi quan sát và chuỗi trạng thái. Nếu như chuỗi trạng thái là hữu hạn và có thể liệt kê được thì chuỗi quan sát trong nhiều trường hợp khó có thể liệt kê được bởi sự phong phú và đa dạng của nó. Để giải quyết vấn đề này, các mô hình sinh phải đưa ra giải thiết về sự độc lập giữa các dữ liệu quan sát, đó là dữ liệu quan sát tại thời điểm t chỉ phụ thuộc vào trạng thái tại thời điểm đó. Điều này hạn chế khá nhiều tính khả năng tích hợp các thuộc tính đa dạng của chuỗi quan sát. Hơn thế nữa, việc các mô hình sinh sử dụng các xác suất đồng thời để mô hình hóa bài toán có tính điều kiện là không thích hợp [15]. Với các bài toán này sẽ là hợp lý hơn nếu ta dùng một mô hình điều kiện để tính trực tiếp xác suất điều kiện thay vì xác suất đồng thời.

Mô hình Markov cực đại hóa entropy (Maximum entropy Markov models – MEMMs) [5] là một mô hình xác suất điều kiện được McCallum đưa ra năm 2000 như là đáp án cho những vấn đề của mô hình Markov truyền thống. Mô hình MEMMs định nghĩa hàm xác suất trên từng trạng thái, với đầu vào là thuộc tính quan sát, đầu ra là xác suất chuyển tới trạng thái tiếp theo. Như vậy mô hình MEMMs quan niệm rằng, dữ liệu quan sát đã được cho trước, điều ta quan tâm là xác suất chuyển trạng thái. So sánh với các mô hình trước đó, MEMMs có ưu điểm là loại bỏ giả thuyết độc lập dữ liệu, theo đó xác suất chuyển trạng thái có thể phụ thuộc vào các thuộc tính đa dạng của chuỗi dữ liệu

quan sát. Hơn nữa, xác suất chuyển trạng thái không chỉ phụ thuộc vào vào quan sát hiện tại mà còn cả quan sát trước đó và có thể cả quan sát sau này nữa.

Tuy nhiên, MEMMs cũng như các mô hình định nghĩa một phân phối xác suất cho mỗi trạng thái đều gặp phải một vấn đề gọi là "label bias"[14][15]: sự chuyển trạng thái từ một trạng thái cho trước tới trạng thái tiếp theo chỉ xem xét xác suất dịch chuyển giữa chúng, chứ không xem xét các xác suất dịch chuyển khác trong mô hình.

CRFs được giới thiệu gần đây như là một mô hình thừa kế các điểm mạnh của MEMMs nhưng lại giải quyết được vấn đề "label bias". CRFs làm tốt hơn cả MEMMs và HMMs trong rất nhiều các bài toán thực về gán nhãn dữ liệu dạng chuỗi [11,12,15]. Điểm khác nhau cơ bản giữa MEMMs và CRFs đó là MEMM định nghĩa phân phối xác suất trên từng trạng thái với điều kiện biết trạng thái trước đó và quan sát hiện tại, trong khi CRF định nghĩa phân phối xác suất trên toàn bộ chuỗi trạng thái với điều kiện biết chuỗi quan sát cho trước. Về mặt lý thuyết, có thể coi mô hình CRF như là một mô hình hữu hạn trạng thái với phân phối xác suất chuyển không chuẩn hóa. Bản chất không chuẩn hóa của xác suất chuyển trạng thái cho phép các bước chuyển trạng thái có thể nhận các giá trị quan trọng khác nhau. Vì thể bất cứ một trạng thái nào cũng có thể làm tăng, giảm xác suất được truyền cho các trạng thái sau đó, mà vẫn đảm bảo xác suất cuối cùng được gán cho toàn bộ chuỗi trạng thái thỏa mãn định nghĩa về xác suất nhờ thừa số chuẩn hóa toàn cục.

Mục ngay tiếp theo trình bày về định nghĩa CRFs, nguyên lý cực đại hóa Entropy với việc xác định hàm tiềm năng cho CRFs. Sau đó là phương pháp huấn luyện mô hình CRFs và thuật toán Viterbi dùng để suy diễn trong CRFs.

2.1 Định nghĩa CRF

Kí hiệu X là biến ngẫu nhiên có tương ứng với chuỗi dữ liệu cần gán nhãn và Y là biến ngẫu nhiên tương ứng với chuỗi nhãn. Mỗi thành phần Y_i của Y là một biến ngẫu nhiên nhận trá trị trong tập hữu hạn các trạng thái S. Ví dụ trong bài toán phân đoạn từ, X nhận giá trị là các câu trong ngôn ngữ tự nhiên, còn Y là chuỗi nhãn tương ứng với các câu này. Mỗi thành phần Y_i của Y là một nhãn xác định phạm vi của một từ trong câu (bắt đầu một từ, ở trong một từ và kết thúc một từ).

Cho một đồ thị vô hướng không có chu trình G = (V,E), trong đó E là tập các cạnh vô hướng của đồ thị, V là tập các đỉnh của đồ thị sao cho Y = { $Y_v \mid v \in V$}. Nói cách khác là tồn tại ánh xạ một – một giữa một đỉnh đồ thị và một thành phần Y_v của Y. Nếu mỗi biến ngẫu nhiên Y_v tuân theo tính chất Markov đối với đồ thị G – tức là xác suất của biến ngẫu nhiên Y_v cho bởi X và tất cả các biến ngẫu nhiên khác $Y_{\{u|u \neq v,\ \{u,v\}\ \in V\}}$:

$p(Y_v \mid X, Y_u, u \neq v, \{u,v\} \in V)$

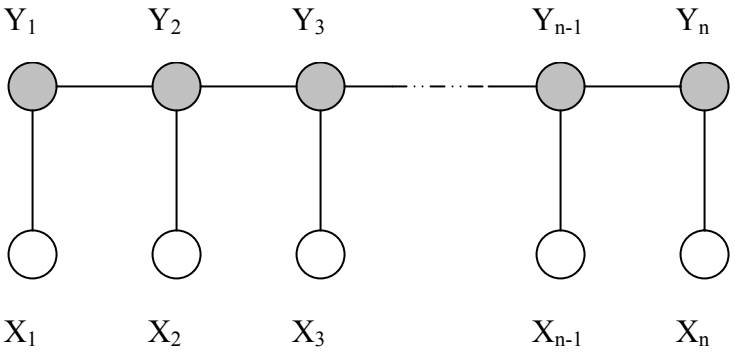
bằng xác suất của biến ngẫu nhiên Y_v cho bởi X và các biến ngẫu nhiên khác tương ứng với các đỉnh kề với đỉnh v trong đồ thị:

$p(Y_v \mid X, Y_{u,} (u,v) \in E)$,

thì ta gọi (X,Y) là một trường ngẫu nhiên điều kiện (Conditional Random Field)

Như vậy, một CRF là một trường ngẫu nhiên phụ thuộc toàn cục vào chuỗi quan sát X. Trong bài toán phân đoạn từ nói riêng và các bài toán xử lý dữ liệu dạng chuỗi nói chung, thì đồ thì G đơn giản chỉ là dạng chuỗi, V= {1, 2, … m}, E= {(i, i+1)}

Kí hiệu X= (X_1, X_2, ... X_n) và Y = (Y_1, Y_2, …Y_n) thì mô hình đồ thị G có dạng sau



Hình 2: đồ thị vô hướng mô tả CRF

Gọi C là tập các đồ thị con đầy đủ của G . Vì G có dạng chuỗi nên đồ thị con đầy đủ thực ra chỉ là một đỉnh hoặc một cạnh của đồ thị G. Áp dụng kết quả của Hammerley-Clifford[13] cho các trường ngẫu nhiên Markov thì phân phối của chuỗi nhãn Y với chuỗi quan sát X cho trước có dạng

$$P(\mathbf{y} \mid \mathbf{x}) = \prod_{A \in C} \psi_A(A \mid \mathbf{x}) \qquad (3.1)$$

Trong đó Ψ_A gọi là hàm tiềm năng, nhận giá trị thực- dương.

Lafferty xác định hàm tiềm năng này dựa trên nguyên lý cực đại entropy. Việc xác định một phân phối theo nguyên lý cực đại entropy có thể hiểu là ta phải xác định một phân phối sao cho "phân phối đó tuân theo mọi giải thiết suy ra từ thực nghiệm, ngoài ra không đưa thêm bất kì giả thiết nào khác" và gần nhất với phân phối đều.

Entropy là độ đo thể hiện tính không chắc chắn, hay độ không đồng đều của phân phối xác suất. Độ đo entropy điều kiện H(Y|X) được cho bởi công thức

$$H(Y \mid X) = -\sum_{x,y} \tilde{p}(x,y) \log q(y \mid x) \qquad (3.2)$$

Với $\tilde{p}(x,y)$ là phân phối thực nghiệm của dữ liệu.

Theo cách trên, Lafferty đã chỉ ra hàm tiềm năng của mô hình CRFs có dạng

$$\psi_A(A \mid \mathbf{x}) = \exp \sum_k \lambda_k f_k(A \mid \mathbf{x}) \qquad (3.3)$$

Trong đó λ_k là thừa số lagrangian ứng với thuộc tính f_k. Ta cũng có thể xem như λ_k là trọng số xác định độ quan trọng của thuộc tính f_k trong chuỗi dữ liệu. Có hai loại thuộc tính là thuộc tính chuyển (kí hiệu là f) và thuộc tính trạng thái (kí hiệu là g) tùy thuộc vào A là một đỉnh hay một cạnh của đồ thị. Thay công thức hàm tiềm năng vào công thức (3.1) và thêm thừa số chuẩn hóa để đảm bảo thỏa mãn điều kiện xác suất ta được

$$p(y \mid x) = \frac{1}{Z(x)} \exp\left(\sum_i \sum_k \lambda_k f_k(y_{i-1}, y_i, x) + \sum_i \sum_k \mu_k g_k(y_i, x) \right) \qquad (3.4)$$

Ở đây, x là chuỗi dữ liệu, y là chuỗi trạng thái tương ứng. $f_k\ (y_{i-1}, y_i, x)$ là thuộc tính của chuỗi quan sát ứng và các trạng thái ứng với vị trí thứ i và i-1 trong chuỗi trạng thái. $g_k(y_i, x)$ là thuộc tính của chuỗi quan sát và trạng thái ứng với trí thứ i trong chuỗi trạng thái.

Các thuộc tính này được rút ra từ tập dữ liệu và có giá trị cố định. Ví dụ:

$$f_i = \begin{cases} 1 \text{ nếu } x_{i-1} = \text{"Học"}, x_i = \text{"sinh"} \text{và } y_{i-1} = B\_W, y_i = I\_W \\ 0 \text{ nếu ngược lại} \end{cases}$$

$$g_i = \begin{cases} 1 \text{ nếu } x_i = \text{"Học"} \text{ và } y_i = \text{B\_W} \\ 0 \text{ nếu ngược lại} \end{cases}$$

Vấn đề của ta bây giờ là phải ước lượng được các tham số $(\lambda_1, \lambda_2, \ldots; \mu_1, \mu_2, \ldots)$ từ tập dữ liệu huấn luyện.

2.2 Huấn luyện CRF

Việc huấn luyện mô hình CRF thực chất là đi tìm tập tham số của mô hình. Kĩ thuật được sử dụng là làm cực đại độ đo likelihood giữa phân phối mô hình và phân phối thực nghiệm. Vì thế việc huấn luyện mô hình CRFs trở thành bài toán tìm cực đại của hàm logarit của hàm likelihood.

Giả sử dữ liệu huấn luyện gồm một tập N cặp, mỗi cặp gồm một chuỗi quan sát và một chuỗi trạng thái tương ứng, D=$\{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}$ $\forall i = 1 \ldots N$. Hàm log-likelihood có dạng sau

$$l(\theta) = \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x}, \mathbf{y}) \log\big(p(\mathbf{y} \mid \mathbf{x}, \theta)\big) \qquad (3.5)$$

Ở đây $\theta(\lambda_1, \lambda_2, ..., \mu_1, \mu_2 ..)$ là các tham số của mô hình và $\tilde{p}(\mathbf{x}, \mathbf{y})$ là phân phối thực nghiệm đồng thời của **x**,**y** trong tập huấn luyện.

Thay p(y|x) của CRFs trong công thức (3.4) vào trên ta được:

$$l(\theta) = \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x}, \mathbf{y}) \left[\sum_{i=1}^{n+1} \lambda f + \sum_{i=1}^{n} \mu g \right] - \sum_{\mathbf{x}} \tilde{p}(\mathbf{x}) \log Z \qquad (3.6)$$

Ở đây, $\lambda(\lambda_1, \lambda_2, .. \lambda_n)$ và $\mu(\mu_1, \mu_2, ..., \mu_m)$ là các vector tham số của mô hình, **f** là vector các thuộc tính chuyển, **g** là vector các thuộc tính trạng thái.

Người ta đã chứng minh được hàm log-likelihood là một hàm lõm và liên tục trong toàn bộ không gian của tham số. Vì vậy ta có thể tìm cực đại hàm log-likelihood bằng phương pháp vector gradient. Mỗi thành phần trong vector gradient sẽ được gán bằng 0:

$$\frac{\partial l(\theta)}{\partial \lambda_k} = E_{\tilde{p}(\mathbf{x},\mathbf{y})}[f_k] - E_{p(\mathbf{y}|\mathbf{x},\theta)}[f_k] \qquad (3.7)$$

Việc thiết lập phương trình trên bằng 0 tương đương với việc đưa ra ràng buộc với mô hình là : giá trị kì vọng của thuộc tính f_k đối với phân phối mô hình phải bằng giá trị kì vọng của thuộc tính f_k đới với phân phối thực nghiệm.

Hiện nay có khá nhiều phương pháp để giải quyết bài toán cực đại hàm log-likelihood, ví dụ như các phương pháp lặp (IIS và GIS), các phương pháp tối ưu số (Conjugate Gradient, phương pháp Newton…). Theo đánh giá của Malouf (2002) thì phương pháp được coi là hiệu quả nhất hiện nay đó là phương pháp tối ưu số bậc hai L-BFGS (limited memory BFGS)

Dưới đây em xin trình bày tư tưởng chính của phương pháp L-BFGS dùng để ước lượng tham số cho mô hình CRFs

L-BFGS là phương pháp tối ưu số bậc hai, ngoài tính toán giá trị của vector gradient, L-BFGS còn xem xét đếm yếu tố về đường cong hàm log-likelihood. Theo công thức khai triển Taylor tới bậc hai của $l(\theta + \Delta)$ ta có:

$$l(\theta + \Delta) \approx l(\theta) + \Delta^T G(\theta) + \frac{1}{2}\Delta^T H(\theta)\Delta \qquad (3.8)$$

Trong đó $G(\theta)$ là vector gradient còn $H(\theta)$ là đạo hàm bậc hai của hàm log-likelihood, gọi là ma trận Hessian. Thiết lập đạo hàm của xấp xỉ trong (3.8) bằng 0 ta tìm được gia số để cập nhật tham số mô hình như sau:

$$\Delta^{(k)} = H^{-1}(\theta^{(k)})G(\theta^{(k)}) \qquad (3.9)$$

Ở đây, k là chỉ số bước lặp. Ma trận Hessian thường có kích thước rất lớn, đặc biết với bài toán ước lượng tham số của mô hình CRFs, vì vậy việc tính trực tiếp nghịch đảo của nó là không thực tế. Phương pháp L-BFGS thay vì tính toán trực tiếp với ma trận

Hessian nó chỉ tính toán sự thay đổi độ cong của vector gradient so với bước trước đó và cập nhật lại.

Công thức (3.9) có thể viết lại là

$$\Delta^{(k)} = B^{-1}(\theta^{(k)})G(\theta^{(k)}) \qquad (3.10)$$

Trong đó ma trận $B^{-1}(\theta)$ phản ánh sự thay đổi độ cong qua từng bước lặp của thuật toán. Yếu tố "giới hạn bộ nhớ" (limited memory) của thuật toán thể hiện ở mỗi bước lặp, các tham số dùng để tính toán $B^{-1}(\theta)$ sẽ được lưu riêng biệt nhau và khi bộ nhớ bị sử dụng hết thì những tham số cũ sẽ được xóa đi để thay vào đó là các tham số mới [10].

Việc xấp xỉ ma trận Hessian theo $B(\theta)$ cho phép phương pháp L-BFGS hội tụ nhanh dù lượng dữ liệu rất lớn. Những thực nghiệm gần đây đã chứng minh rằng phương pháp L-BFGS đạt kết quả vượt trội so với các phương pháp khác[17].

2.3 Suy diễn CRF

Sau khi tìm được mô hình CRFs từ tập dữ liệu huấn luyện, nhiệm vụ của ta lúc này là làm sao dựa vào mô hình đó để gán nhãn cho chuỗi dữ liệu quan sát, điều này tương đương với việc làm cực đại phân phối xác suất giữa chuỗi trạng thái y và dữ liệu quan sát x. Chuỗi trạng thái y* mô tả tốt nhất chuỗi dữ liệu quan sát x sẽ là nghiệm của phương trình

$$y^* = \text{argmax}\{p(y \mid x)\}$$

Chuỗi y* có thể xác định được bằng thuật toán Viterbi.

Gọi S là tập tất cả trạng thái có thể, ta có $|S| = m$. Xét một tập hợp các ma trận cỡ $m \times m$ kí hiệu $\{ M_i(x) \mid i= 0,2\ldots n-1\}$ được định nghĩa trên từng cặp trạng thái $y, y' \in S$ như sau

$$M_i(y', y \mid x) = \exp\left(\sum_k \lambda_k f_k(y', y, x) + \sum_k \mu_k g_k(y, x)\right) \qquad (3.5)$$

Bằng việc đưa thêm hai trạng thái y_{-1} và y_n vào trước và sau chuỗi trạng thái. Coi như chúng ứng với trạng thái "start" và "end", phân phối xác suất có thể viết là

$$p(y \mid x, \lambda) = \frac{1}{Z(x)} \prod_{i=0}^{n} M_i(y', y \mid x) \qquad (3.6)$$

Ở đây Z(x) là thừa số chuẩn hóa được đưa thêm vào và có thể tính được dựa vào các M_i, nhưng vấn đề ta quan tâm là cực đại hóa p(y|x) nên không cần thiết phải tính Z(x). Như vậy ta chỉ cần cực đại hóa tích n+1 phần tử trên. Tư tưởng chính của thuật toán Viterbi là tăng dần chuỗi trạng thái tối ưu bằng việc quét các ma trận từ vị trí 0 cho đến vị trí n. Tại mỗi bước i ghi lại tất cả các chuỗi tối ưu kết thúc bởi trạng thái y với $\forall y \in S$ (ta kí hiệu là $y_i^*(y)$) và tích tương ứng $P_i(y)$:

Bước 1: $P_0(y) = M_0(start, y \mid x)$ và $y_0^*(y) = y$

Bước lặp: Cho i chạy từ 1 đến n tính:

$P_i(y) = \max_{y' \in S} P_{i-1}(y) \times M_i(y', y \mid x)$

$y_i^*(y) = y_{i-1}^*(\hat{y}).(y)$, trong đó $\hat{y} = \arg\max_{y' \in S} P_{i-1}(y) \times M_i(y', y \mid x)$ và "." là toán tử cộng chuỗi

Chuỗi $y_{n-1}^*(y)$ chính là chuỗi có xác suất p(y*|x) lớn nhất, đó cũng chính là chuỗi nhãn phù hợp nhất với chuỗi dữ liệu quan sát x cho trước.

2.4 Tổng kết chương

Chương này đã giới thiệu những vấn đề cơ bản về CRF: định nghĩa CRF, cách ước lượng tham số cho CRF và các suy diễn trong CRF để gán nhãn cho dữ liệu chưa được gán nhãn. Trong chương tiếp theo, em xin trình bày về bài toán phân đoạn tiếng Việt theo hướng áp dụng mô hình CRF

Chương 3. Phân đoạn từ tiếng Việt với mô hình CRF

Trước đây, chúng ta đã đề cập tới bài toán phân đoạn từ trong tiếng Việt và ở chương này, bài toán sẽ được mô tả cụ thể hơn theo hướng áp dụng mô hình CRF, bao gồm việc mô tả việc gán nhãn, chuẩn bị dữ liệu, cách chọn thuộc tính và cách đánh giá mô hình. Như đã được đề cập, phân đoạn tiếng Việt là một trong những nội dung quan trọng trong việc xây dựng các công cụ xử lý tiếng Việt của Việt Nam.

3.1 Mô tả bài toán phận đoạn từ tiếng Việt

Ta có thể quy bài toán phân đoạn từ tiếng Việt thành bài toán gán nhãn cho các âm tiết tiếng Việt. Dựa vào các nhãn đó ta có thể xác định được ranh giới của từng từ trong văn bản tiếng Việt. Các nhãn được sử dụng ở đây là

- B_W: nhãn đánh dấu bắt đầu một từ
- I_W: nhãn đánh dấu ở trong một từ
- O: nhãn đánh dấu ở ngoài tất cả các từ

Như vậy bài toán phân đoạn từ tiếng Việt có thể phát biểu là:

"Hãy xây dựng một mô hình để gán nhãn {B_W, I_W,O} cho các âm tiết của văn bản tiếng Việt chưa được phân đoạn".

Để có thể xây dựng được một mô hình tốt, trước hết ta phải chuẩn bị được một tập dữ liệu huấn luyện đầy đủ và chính xác.

3.1.1 Thu thập dữ liệu

Dữ liệu dùng cho huấn luyện được thu thập từ rất nhiều nguồn khác nhau trên mạng internet như báo điện tử Vnexpress, báo Người lao động, báo VietNamNet, báo Tuổi trẻ. Các bài báo thuộc nhiều lĩnh vực, em xin liệt kê cụ thể dưới đây

Bảng 2: Thống kê dữ liệu sử dụng trong các lĩnh vực

STT	Lĩnh vực	Số lượng bài	Số lượng câu
1	Kinh tế	90 bài	
2	Công nghệ thông tin	59 bài	
3	Giáo dục	38 bài	

4	Ô tô – xe máy	35 bài	
5	Thể thao	28 bài	
6	Pháp luật	31 bài	
7	Văn hóa- xã hội	24 bài	
	Tổng cộng	305 bài	

Để tăng độ chính xác của mô hình, em còn thu thập dữ liệu về tên riêng của người, tổ chức nhằm hỗ trợ cho việc phát hiện từ mới.

- Khoảng 20672 tên người lấy từ internet và từ trang http://www.vietnamgiapha.com
- Khoảng 707 tên địa danh Việt Nam lấy từ http://vi.wikipedia.org

3.1.2 Chuẩn bị dữ liệu

Các dữ liệu sau khi thu thập từ trên internet sẽ được lọc lấy nội dung chính . Sau đó dữ liệu được xử lý bán tự động qua 2 giai đoạn, nhằm đảm bảo độ chính xác của dữ liệu dùng cho huấn luyện mô hình.

Giai đoạn 1: Sử dụng phần mềm tách từ tự động WordMatching của CN. Nguyễn Cẩm Tú. Đây là một phần mềm tách tự dựa trên phương pháp Maximum Matching. Dữ liệu từ điển được dùng cho việc phân đoạn là từ điển Lạc-Việt, đây là từ điển khá phong phú về lượng từ, vì thế sau khi phân từ tự động, phần lớn dữ liệu đã được phân chính xác. Tuy nhiên, ta sẽ gặp những khó khăn do đặc điển của tiếng Việt, cũng như đặc điểm của dữ liệu lấy từ internet như sau

- Nhập nhằng trong tiếng Việt
- Từ mới không có trong từ điển, tiêu biểu là các từ tiếng nước ngoài
- Từ sai chính tả

Giai đoạn 2: kiểm tra thủ công. Việc kiểm tra được thực hiện bởi 2 người theo phương thức kiểm tra chéo, tức là dữ liệu sau khi được người thứ nhất kiểm tra sẽ được người thứ 2 kiểm tra lại. Sau đó cả 2 người thống nhất các nhập nhằng của việc phân đoạn trước khi dữ liệu chính thức được dùng cho việc huấn luyện mô hình.

3.1.3 Đầu vào và đầu ra của mô hình CRFs

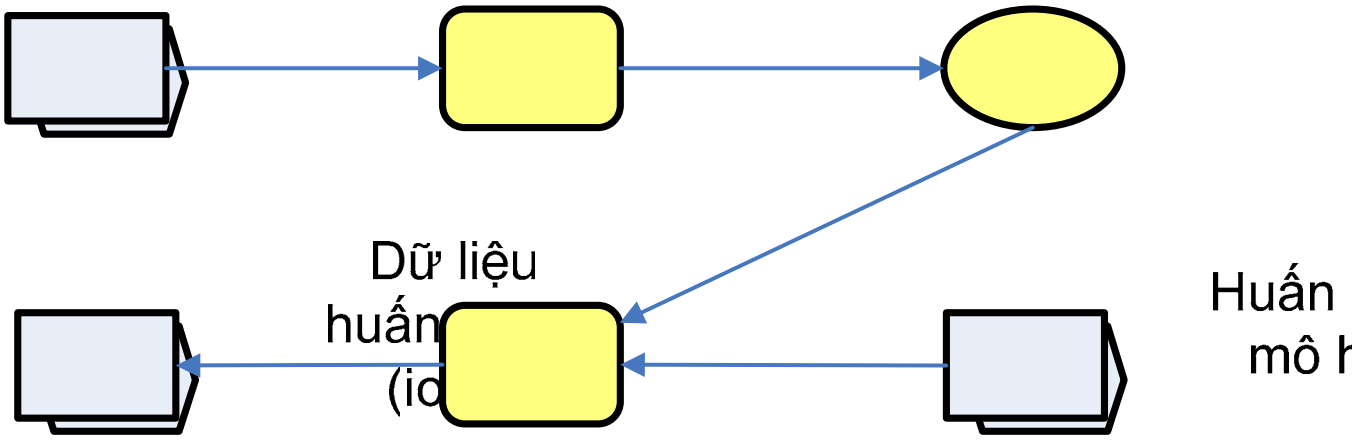
Dữ liệu sau khi được thu thập và phân đoạn sẽ được chuyển đổi về dạng iob2, là dạng dữ liệu đầu vào cho mô hình CRFs. Cấu trúc của định dang iob2 được thể hiện ở bảng dưới đây

Bảng 3: Ví dụ về dữ liệu định dạng chuẩn iob2

Kỹ	B_W
thuật	I_W
môi	B_W
trường	I_W
,	O
khoa	B_W
học	I_W
môi	B_W
trường	I_W
,	O
công	B_W
nghệ	I_W
môi	B_W
trường	I_W
có	B_W
phải	I_W
là	B_W
một	B_W
ngành	B_W
.	O

Trong định dạng iob2 trên, cột đầu được gọi là cột dữ liệu quan sát, cột tiếp theo là chuỗi trạng thái. Mỗi âm tiết sẽ được ghi trên một dòng kèm theo nhãn của chúng. Các câu sẽ được cách nhau bởi một dòng trắng

Dữ liệu định dang iob2 sẽ được huấn luyện để đưa ra một mô hình CRFs, mô hình này được dùng để phân đoạn các văn bản mới. Ta có thể mô tả quá trình này như sau:



Hình 3: quá trình phân đoạn sử dụng mô hình CRF

Dữ liệu định dạng iob2 được dùng để sinh ra các thuộc tính phục vụ việc huấn luyện mô hình. Thuộc tính được lựa chọn thế nào phụ thuộc vào từng bài toán cụ thể. Việc chọn thuộc tính tốt hay không sẽ ảnh hưởng rất nhiều đến kết quả của chương trình.

3.2 Lựa chọn thuộc tính

Lựa chọn các thuộc tính từ tập dữ liệu huấn luyện là nhiệm vụ quan trọng nhất, giữ vai trò quyết định đối với chất lượng của toàn bộ hệ thống. Các thuộc tính được chọn càng tinh tế, có ý nghĩa thì độ chính xác của hệ thống càng cao. Do đó việc tìm hiểu từ vựng tiếng Việt như đã trình bày ở chương 3 là rất có ích.

Văn bản đã được phân đoạn

Phân

Các thuộc tính tại vị trí i trong chuỗi dữ liệu quan sát gồm hai phần, một là thông tin ngữ cảnh tai vị trí i của chuỗi dữ liệu quan sát, một là phần thông tin về nhãn tương ứng. Công việc lựa chọn các thuộc tính thực chất là chọn ra các mẫu vị từ ngữ cảnh (context predicate template), các mẫu này thể hiện những các thông tin đáng quan tâm tại một vị trí bất kì trong chuỗi dữ liệu quan sát. Áp dụng các mẫu ngữ cảnh này tại môt vị trí trong chuỗi dữ liệu quan sát cho ta các thông tin ngữ cảnh (context predicate) tại vị trí đó. Mỗi thông tin ngữ cảnh tại i khi kết hợp với thông tin nhãn tương ứng tại vị trí đó sẽ cho ta một thuộc tính của chuỗi dữ liệu quan sát tại i. Như vậy một khi đã có các mẫu ngữ cảnh, ta có thể rút ra được hàng nghìn thuộc tính một cách tự động từ tập dữ liệu huấn luyện.

3.2.1 Mẫu ngữ cảnh từ điển

Mẫu ngữ cảnh từ điển cho ta các thuộc tính cho phép xác định từ A có ở trong danh sách các từ đã biết không, ví dụ ta có thuộc tính "âm tiết đi liền trước và âm tiết hiện tại kết hợp thành một từ có trong danh sách tên người Việt Nam"

Trong bài toán này, em sử dụng dữ liệu từ điển lạc việt và thu thập các danh sách thông tin khác từ internet. Cụ thể như sau:

Bảng 4: Mẫu ngữ cảnh dạng từ điển

Mẫu ngữ cảnh	Ý nghĩa
in_lacviet_dict	Có mặt trong từ điển lạc Việt không
family_name	Họ trong tiếng Việt
middle_name	Tên đệm trong tiếng Việt
last_name	Tên trong tiếng Việt
vnlocation	Địa danh tiếng Việt

3.2.2 Mẫu ngữ cảnh từ vựng

Bảng 5: Mẫu ngữ cảnh từ vựng

Mẫu ngữ cảnh	Ý nghĩa
S_{-2}	Âm tiết quan sát tại vị trí -2 so với vị trị hiện tại
S_{-1}	Âm tiết quan sát tại vị trí liền trước so với vị trị hiện tại
S_1	Âm tiết quan sát tại vị trí liền sau so với vị trí hiện tại
S_2	Âm tiết quan sát tại vị trí +2 so với vị trị hiện tại
S_0S_1	Âm tiết quan sát tại vị trí hiện tại và vị trí liền sau
$S_{-1}S_0$	Âm tiết quan sát tại vị trí liền trước và vị trí hiện tại
$S_{-2}S_{-1}$	Âm tiết quan sát tại vị trí -2 và dữ vị trí liền trước
$S_{-1}S_0S_1$	Âm tiết quan sát tại vị trí liền trước, hiện tại và liền sau

Ở đây ta chọn kích thước của sổ trượt (sliding window) bằng 5, vì đa số các từ trong tiếng Việt có độ dài nhỏ hơn 3 âm tiết, các từ nhiều hơn 3 âm tiết chỉ chiếm khoảng 3,1%.

3.2.3 Mẫu ngữ cảnh phát hiện tên thực thể.

Các tên thực thể thường được viết hoa kí tự đầu tiên, vì thế ta có thể thêm thuộc tính viết hoa vào mô hình. Nếu tất cả kí tự đều viết hoa thì khả năng từ đó là tên viết tắt của tổ chức. Tuy nhiên nếu một âm tiết được viết hoa mà nó đứng ở đầu câu thì thông tin viết hoa không có ý nghĩa nữa.

Bảng 6: Mẫu ngữ cảnh phát hiện tên thực thể

Mẫu ngữ cảnh	Ý nghĩa
InitialCap	Âm tiết viết hoa
AllCap	Âm tiết viết in
FirstObsr	Từ đầu tiên của câu
Mark	Dâu câu (ví dụ: chấm, phẩy, chấm phẩy...)

3.2.4 Mẫu ngữ cảnh phát hiện từ láy.

Phát hiện các từ láy trong tiếng Việt, bao gồm từ láy bộ phần và từ láy toàn bộ như đã trình bày ở trước.

Bảng 7: Mẫu ngữ cảnh phát hiên từ láy toàn bộ đối thanh điệu

Mẫu ngữ cảnh	Ý nghĩa
Full_Dup	Có phải láy toàn bộ không
Part_Dup	Có phải láy bộ phận không

3.2.5 Mẫu ngữ cảnh âm tiết tiếng Việt.

Dựa vào cấu trúc âm tiết đã trình bày ở trước, ta xây dựng mẫu ngữ cảnh xác định một âm tiết có phải là âm tiết tiếng Việt không

Bảng 8: Mẫu ngữ cảnh phát hiên từ láy toàn bộ đối thanh điệu

Mẫu ngữ cảnh	Ý nghĩa
not_valid_vnsyll	Âm tiết không có trong tiếng Việt (ví dụ: hard,soft..)

3.2.6 Mẫu ngữ cảnh dạng regular expression

Bảng 9: Mẫu ngữ cảnh phát hiên từ láy toàn bộ đối thanh điệu

Mẫu ngữ cảnh	Ví dụ	Ý nghĩa
^\d+[\.|,]{0,1}\d+$	94,19 ; 94.19	Số
^\d+[/|-]\d+[/|-]\d+$	25/05/2006 ; 25-05-06	Ngày tháng
^\d+[/|-]\d+$	25/05 ; 25-05	Ngày dạng ngắn
^\d+%$	100%	Phần trăm

3.3 Cách đánh giá

3.3.1 Phương pháp đánh giá

Phương pháp đánh giá được dùng trong luận văn là phương pháp ước lượng chéo trên k tập con, với dữ liệu của bài toán em chọn k=5. Quá trình huấn luyện được thực hiện 5 lần. Tại mỗi lần huấn luyện, dữ liệu huấn luyện được chia thành 5 phần bằng nhau, trong đó 1 phần để kiểm tra, 4 phần được trộn lại để huấn luyện. Ở mỗi bước lặp của quá trình huấn luyện, hệ thống tiến hành đo các chỉ số đánh giá độ chính xác : độ chính xác (precision), độ hồi tưởng (recall), độ đo F1.

Sau các bước huấn luyện ta sẽ chọn ra bước lặp có chỉ số F1 cao nhất, vì độ lớn của chỉ số F1 sẽ phản ánh chất lượng của mô hình.

3.3.2 Các đại lượng đo độ chính xác

Việc đánh giá độ chính xác của mô hình phân đoạn từ của chúng ta là rất quan trong. Nó cho phép ta so sánh độ chính xác của mô hình giữa các tập dữ liệu huấn luyện, hơn nữa, có thể so sánh độ chính xác của mô hình do ta xây dựng với những mô hình phân đoạn từ đã có hiện nay. Có nhiều cách để đánh giá độ chính xác của mô hình phân đoạn từ, nhưng cách phổ biến nhất hiện này là sử dụng các đô đo như độ chính xác (precision), độ hồi tưởng (recall), độ đo F1. Độ đo F1 là một chỉ số cân bằng giữa độ chính xác và độ hồi tưởng. Nếu độ chính xác và độ hồi tưởng cao và cân bằng thì độ đo F1 lớn, còn độ chính xác và hồi tưởng nhỏ và không cân bằng thì độ đo F1 nhỏ. Mục tiêu của ta là xây dựng mô hình phân đoạn từ có chỉ số F1 cao.

Độ đo dựa theo từ được tính theo các công thức sau:

$$\text{Recall} = \frac{c}{N} \quad (4.1)$$

$$\text{Precision} = \frac{\text{c}}{\text{n}} \quad (4.2)$$

$$\text{F} = \frac{2 \text{ x Recall x Precision}}{\text{Recall + Precision}} \quad (4.3)$$

Trong đó

- Kí hiệu c là số lượng từ được hệ thống phân đoạn đúng
- Kí hiệu N là số lượng từ trong văn bản
- Kí hiệu n là số lượng từ được hệ thống phân đoạn

Sau khi có các độ đo, ta tính kết quả trung bình cho từng độ đo của bước lặp tương ứng. Có 2 loại kết quả trung bình là Avg1 và Avg2:

- Kết quả trung bình loại Avg1 cho một độ đo là một số được tính bằng trung bình cộng độ đo tương ứng .
- Kết quả trung bình loại Avg2 là kết quả được tính trên kết quả tổng thể. Trong tường hợp các độ đo tính dựa trên từ thì kết quả trung bình loại Avg2 bằng kết quả trung bình loại Avg1.

Ví dụ ta cần phân đoạn một văn bản có 100 từ, hệ thống phân đoạn được 102 từ trong đó có 90 từ là phân đoạn đúng thì các độ đo được tính là:

$$\text{Recall} = \frac{90}{100} = 90\%$$

$$\text{Precision} = \frac{90}{102} = 88\%$$

$$\text{F} = \frac{2 \times 90\% \times 88\%}{90\% + 88\%} = 88{,}98\%$$

Trong bài toán phân đoạn từ của ta có thể đánh giá độ chính xác dựa trên nhãn hoặc dựa trên từ. Độ chính xác dựa trên nhãn chỉ tính đến độ chính xác của việc gán nhãn cho các âm tiết. Độ chính xác dựa trên từ đánh giá tính chính xác của hệ thống trong việc phân đoạn từ, vì thế chỉ số dựa trên từ có ý nghĩa hơn trong bài toán của phân đoạn. Ví

dụ: nếu từ "bộ giáo dục" được gán nhãn là "B_W B_W O" trong khi nhãn đúng phải là "B_W B_W I_W" thì độ chính xác tính theo nhãn sẽ là 2/3, độ chính xác theo từ là 1/2.

3.4 Tổng kết chương

Chương này đã trình bày quá trình chuẩn bị dữ liệu và việc xây dựng ngữ cảnh lựa chọn thuộc tính cho mô hình CRF, đồng thời đưa ra cách đánh giá mô hình. Chương tiếp theo trình bày về kết quả của việc áp dụng mô hình CRF vào bài toán phân đoạn từ tiếng Việt

Chương 4. Thử nghiệm và đánh giá

Việc xây dựng được một hệ thống phân đoạn từ tiếng Việt sẽ góp phần quan trọng vào việc xây dựng cơ sở tri thức tiếng Việt. Tuy rằng bài toán phân đoạn từ là một bài toán rất cơ bản trong xử lý ngôn ngữ tự nhiên, nhưng đối với tiếng Việt thì lại là một bài toán không hề đơn giản. Mặc dù những khó khăn do đặc thù tiếng Việt, những thử nghiệm ban đầu của em cho tiếng Việt cũng đạt được một số kết quả đáng khích lệ

4.1 Môi trường thử nghiệm

4.1.1 Phần cứng

Máy tính IBM, chip Intel Pentium 4 CPU 2.40GHz, RAM 382 MB

4.1.2 Phần mềm

FlexCRFs là một CRF Framework cho các bài toán gán nhãn dữ liệu dữ liệu dạng chuỗi như POS tagger, Noun Phrase Chunking, Word Segmentation... Đây là một công cụ mã nguồn mở được phát triển bởi ThS. Phan Xuân Hiếu và TS. Nguyễn Lê Minh (Viện JAIST-Nhật Bản).

WordMatching là một phần mềm phân đoạn từ tiếng Việt sử dụng phương pháp Maximum Matching với từ điển. Phần mềm được phát triển bởi CN. Nguyễn Cẩm Tú (ĐH Công Nghệ, ĐH Quốc Gia HN).

4.2 Mô tả thử nghiệm

4.2.1 Thiết lập tham số

Các tham số tùy chọn dùng trong FlexCRFs Framework được thiết lập như sau

Bảng 10: Các tham số huấn luyện dùng tron FlexCRFs

Tham số	Giá trị	Ý nghĩa
init_lamda_val	0	Giá trị khởi tạo cho các tham số trong mô hình
num_iterations	150	Số bước lặp huấn luyện

f_rare_threshold	1	Chỉ có các thuộc tính có tần số xuất hiện lớn hơn giá trị này thì mới được tích hợp vào mô hình CRF
cp_rare_threshold	1	Chỉ có các mẫu vị từ ngữ cảnh có tần số xuất hiện lớn hơn giá trị này mới được tích hợp vào mô hình CRF
eps_log_likelihood	0.01	Giá trị này cho ta điều kiện dừng của vòng lặp huấn luyện, nếu như |log_likelihood(t)-log_likelihood(t-1)|<0.01 thì dừng quá trình huấn luyện .Ở đây t và t-1 là bước lặp thứ t và t-1.

4.2.2 Mô tả thử nghiệm

Để đánh giá phương pháp CRF với bài toán phân đoạn từ tiếng Việt và tìm ra một cách lựa chọn thuộc tính tốt nhất, em đã tiến hành năm thử nghiệm . Trước hết để tiện cho việc mô tả thử nghiệm, em xin phân nhóm các thuộc tính được xây dựng từ các mẫu ngữ cảnh được trình bày ở trước như sau

- Nhóm 1- Syllable Conjuction: các cách kết hợp âm tiết với kích thước cửa số trượt là 5
- Nhóm 2 - Regex: là các thuộc tính về ngày, số, phần trăm…
- Nhóm 3 - Lexicon: xác định các từ liệu có mặt trong từ điển Lạc Việt không, hay có phải là tên riêng tiếng Việt, tên địa danh tiếng Việt không
- Nhóm 4- Vietnamese Syllable: xác định một âm tiết liệu có mặt trong tiếng Việt không
- Nhóm 5- Reduplicate: xác định một từ có phải từ láy hay không.

Trong các thử nghiệm, em tiến hành đánh giá một số thuộc tính thuộc nhóm Lexicon, nhóm Reduplicate và nhóm Vietnamese Syllable.

Thử nghiệm 1: Đây là thí nghiệm khá đặc biệt, vì em tiến phân đoạn từ tiếng Việt sử dụng phương pháp Maximum Matching với từ điển Lạc – Việt. Mục đích của thử nghiệm này là để đánh giá so sánh với kết quả của các thí nghiệm tiếp theo.

Thử nghiệm 2: phân đoạn từ sử dụng mô hình CRF và chỉ sử dụng các thuộc tính nhóm 1 và nhóm 2.

Thử nghiệm 3: phân đoạn từ sử dụng mô hình CRF , các thuộc tính thuộc nhóm 1 và nhóm 2, đồng thời đưa vào thuộc tính in_lacviet_dict (có mặt trong từ điển Lạc Việt) ở trong nhóm 3 vào thử nghiệm. Mục đích thử nghiệm này đánh giá độ quan trọng của từ điển đối với mô hình.

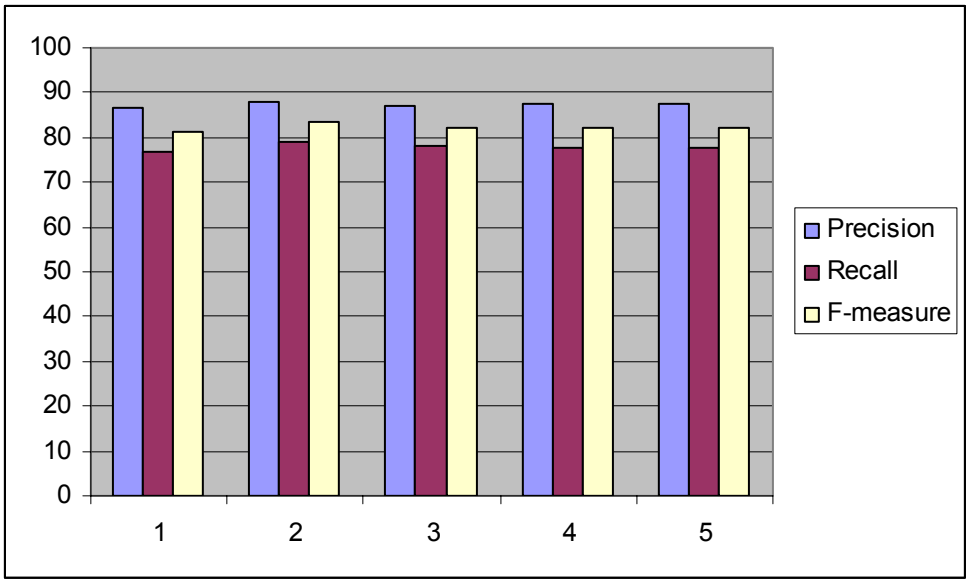
Thử nghiệm 4: phân đoạn từ sử dụng mô hình CRF, đưa vào các thuộc tính thuộc nhóm 1, nhóm 2, nhóm 3 và nhóm 4. Việc đưa thêm các phát hiện tên người và địa danh tiếng Việt nhằm mang lại kết quả cao hơn các thử nghiệm trước

Thử nghiệm 5: phân đoạn từ sử dụng mô hình CRF với tất cả các thuộc tính thuộc năm nhóm trên. Đây là một thử nghiệm đầy đủ với số lượng thuộc tính rất lớn với việc phát hiện từ láy.

4.3 Kết quả thử nghiệm

4.3.1 Thử nghiệm 1

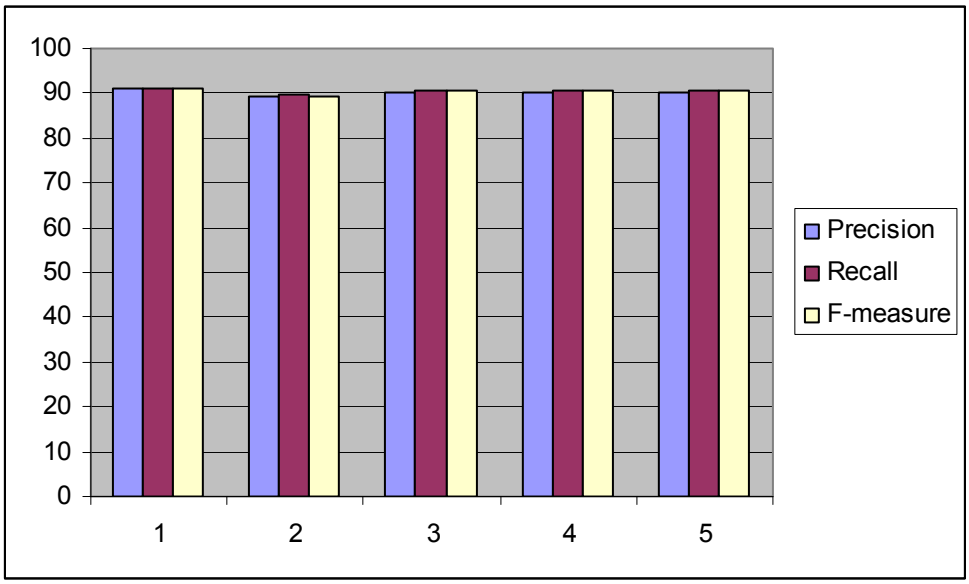
Kết quả sau 5 lần thử nghiệm với maximum matching như sau



Hình 4: kết quả 3 độ đo với thử nghiệm 1 qua 5 lần thử nghiệm

4.3.2 Thử nghiệm 2

4.3.2.1 Kết quả 5 lần thử nghiệm



Hình 5: kết quả 3 độ đo thử nghiệm 2 qua 5 lần thử nghiệm

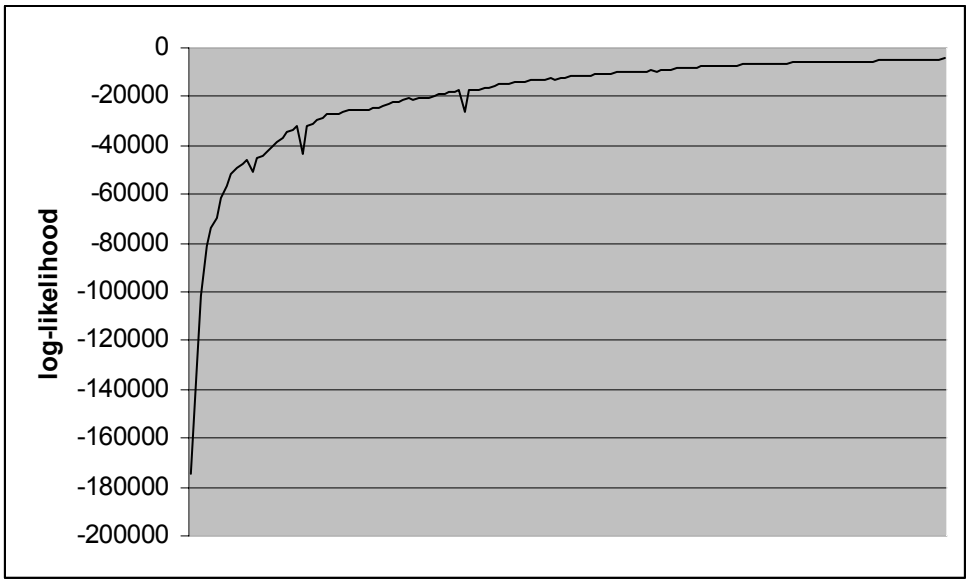
4.3.2.2 Lần thử nghiệm cho kết quả tốt nhất

Bảng 11: Đánh giá mức độ nhãn – lần thử nghiệm cho kết quả tốt nhất

Label	Manual	Model	Match	Pre. (%)	Rec. (%)	F-Measure(%)
B_W	26189	26240	25338	96.56	96.75	96.66
I_W	10000	9967	9115	91.45	91.15	91.30
O	4561	4543	4505	99.16	98.77	98.97
AVG1.				95.73	95.56	95.64
AVG2.	40750	40750	38958	95.60	95.60	95.60

Bảng 12: Đánh giá mức độ từ – lần thử nghiệm cho kết quả tốt nhất

Label	Manual	Model	Match	Pre. (%)	Rec. (%)	F-Measure(%)
Word	26189	26240	25338	90.85	91.03	90.94
AVG1.				90.85	91.03	90.94
AVG2.	40750	40750	38958	90.85	91.03	90.94



Hình 6: Quá trình tăng likelihood qua 150 bước lặp

4.3.2.3 Trung bình 5 lần thực nghiệm

Bảng 131: Đánh giá mức nhãn- Trung bình 5 lần thử nghiệm

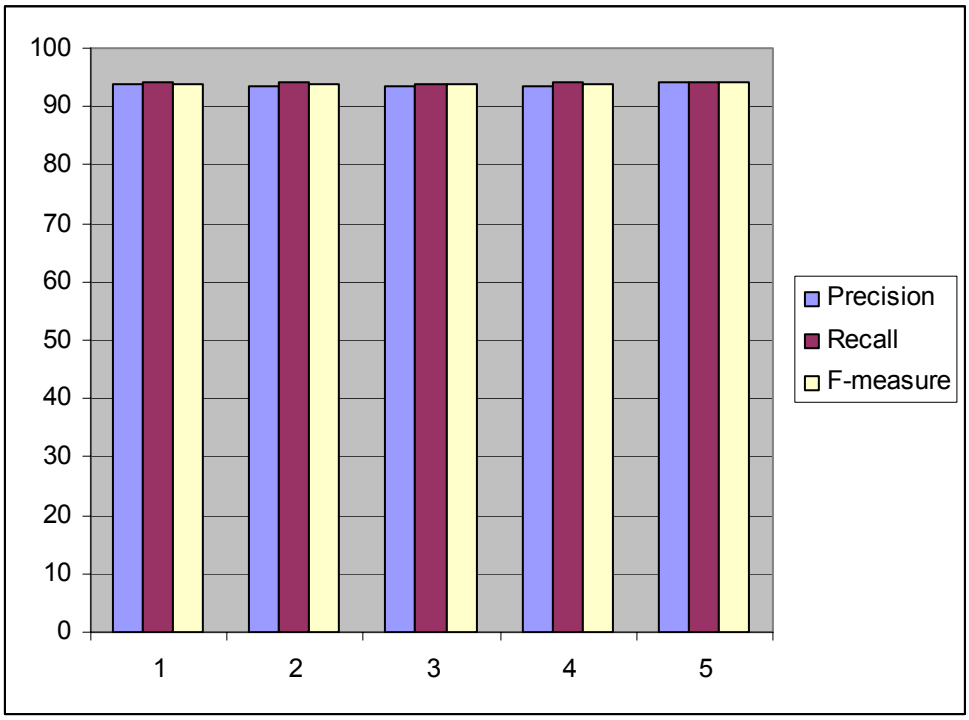
Độ đo	Giá trị (%)
Precision	95.342
Recall	95.342
F-measure	95.342

Bảng 142: Đánh giá ở mức từ – trung bình 5 lần thử nghiệm

Độ đo	Giá trị (%)
Precision	90.20
Recall	90.536
F-measure	90.368

4.3.3 Thử nghiệm 3

4.3.2.1 Kết quả 5 lần thử nghiệm



Hình 7: kết quả 3 độ đo thử nghiệm 3 qua 5 lần thử nghiệm

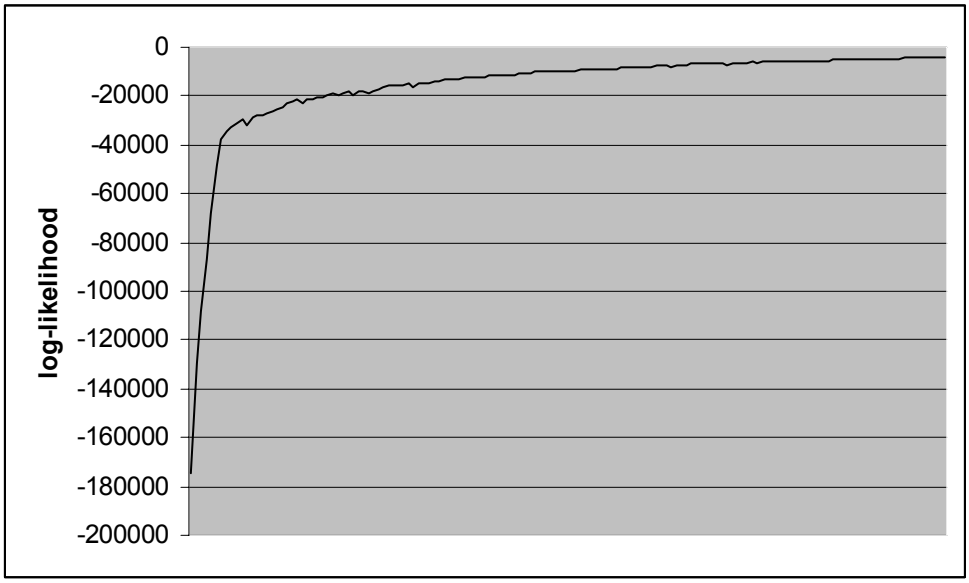
4.3.2.2 Lần thử nghiệm cho kết quả tốt nhất

Bảng 14: Đánh giá mức độ nhãn – lần thử nghiệm cho kết quả tốt nhất

Label	Manual	Model	Match	Pre. (%)	Rec. (%)	F-Measure(%)
B_W	25910	25974	25396	97.77	98.02	97.9
I_W	9929	9890	9369	94.73	94.36	94.55
O	4724	4699	4664	99.26	98.73	98.99
AVG1.				97.25	97.04	97.14
AVG2.	40563	40563	39429	97.2	97.2	97.2

Bảng 16: Đánh giá mức độ từ – lần thử nghiệm cho kết quả tốt nhất

Label	Manual	Model	Match	Pre. (%)	Rec. (%)	F-Measure(%)
Word	25910	25974	24435	94.07	94.31	94.19
AVG1.				94.07	94.31	94.19
AVG2.	25910	25974	24435	94.07	94.31	94.19



Hình 8: Quá trình tăng likelihood qua 150 bước lặp

4.3.2.3 Trung bình 5 lần thực nghiệm

Bảng 173: Đánh giá mức nhãn- Trung bình 5 lần thử nghiệm

Độ đo	Giá trị (%)
Precision	97.098
Recall	97.098
F-measure	97.098

Bảng 184: Đánh giá ở mức từ – trung bình 5 lần thử nghiệm

Độ đo	Giá trị (%)
Precision	93.65
Recall	93.65
F-measure	93.65

4.3.4 Thử nghiệm 4

4.3.2.1 Kết quả 5 lần thử nghiệm

4.3.2.2 Lần thử nghiệm cho kết quả tốt nhất

4.3.2.3 Trung bình 5 lần thực nghiệm

4.3.5 Thử nghiệm 5

4.3.2.1 Kết quả 5 lần thử nghiệm

4.3.2.2 Lần thử nghiệm cho kết quả tốt nhất

4.3.2.3 Trung bình 5 lần thực nghiệm

4.4 Phân tích và thảo luận kết quả thử nghiệm

4.5 Tổng kết chương

Phần kết luận

Tổng kết công việc đã làm và đóng góp của luận văn

Khóa luận đã hệ thống hóa một số vấn đề về phân đoạn từ tiếng Việt bao gồm tìm hiểu về từ vựng tiếng Việt, các hướng tiếp cận bài toán phần đoạn từ tiếng Việt kèm theo đánh giá nhận xét. Đồng thời đề xuất phương án phân đoạn từ tiếng Việt bằng học máy sử dụng mô hình CRF, thực nghiệm trên dữ liệu tiếng Việt cho kết quả rất khả quan. Sau đây là tóm lược một số ý chính luận văn đã đề cập tới

- Đã trình bày hệ thống về mô hình CRF, gồm định nghĩa, các huấn luyện mô hình và cách suy diễn mô hình. Chương này cũng cho thấy mô hình CRF tốt hơn so với các phương pháp trước đó như MEMM…

- Đã mô tả chi tiết các phương pháp phân đoạn tiếng Việt theo hướng thi hành phương pháp áp dụng mô hình CRF. Quá trình thu thập và xử lý dữ liệu đã mô tả chi tiết. Đã đề xuất một số mẫu ngữ cảnh với các đặc điểm riêng của tiếng Việt. Chương này cũng đã đưa ra cách đánh giá độ chính xác của mô hình theo ước lượng chéo trên k tập con, với ba độ đo là độ chính xác, độ hồi tưởng, và độ đo F1.

Kết quả thực nghiệm và các đánh giá được trình bày chi tiết trong chương 4. Nhiều thử nghiệm đã được tiến hành để so sánh và tìm ra mô hình tốt nhất cho bài toán, và luận văn cũng đạt được những kết quả khả quan.

Hướng nghiên cứu tiếp theo

Mặc dù kết quả thu được của luận văn là đáng khích lệ nhưng trong thời gian có hạn, em chưa thể thu thập dữ liệu lớn hơn và tiến hành thêm nhiều thử nghiệm khác nhau. Trong thời gian tới, em sẽ tiến hành thu thập thêm các dữ liệu sách báo, truyện tiếng Việt, các bài văn cổ như truyện Kiều,…với lượng dữ liệu phong phú nhiều lĩnh vực em hi vọng sẽ đạt được kết quả cao hơn nữa.

Cũng trên cơ sở kết quả đạt được của luận văn, em dự định xây dựng một phần mềm hoàn chỉnh cho phép phân đoạn các văn bản tiếng Việt với độ chính xác cao, tiện dụng và đem lại hiệu quả thiết thực trong xử lý văn bản tiếng Việt.

Phân đoạn từ tiếng Việt là mới chỉ là bước đầu trong xử lý văn bản tiếng Việt, thời gian tới em sẽ tiếp tục tìm hiểu thêm các lĩnh vực tiếp khác như phân loại văn bản…

Tài liệu tham khảo

[1] Mai Ngọc Chừ; Vũ Đức Nghiệu & Hoàng Trọng Phiến. Cơ sở ngôn ngữ học và tiếng Việt. Nxb Giáo dục, H., 1997, trang 142–152.

[2] Nguyễn Việt Cường. Bài toán lọc và phân lớp nội dung Web tiếng Việt với hướng tiếp cận Entropy cực đại. Luận văn tốt nghiệp ĐHCN 2005

[3] Nguyễn Cẩm Tú. Nhận biết các loại thực thể trong văn bản tiếng Việt nhằm hỗ trợ Web ngữ nghĩa và tìm kiếm hướng thực thể. Luận văn tốt nghiệp ĐHCN 2005

[4] Website: http://ngonngu.net/

[5] A. McCallum, D. Freitag, and F. Pereia. Maximum entropy markov models for information extraction and segmentation. In Proc. Interational Conference on Machine Learning, 2000

[6] Andrew McCallum. Efficiently Inducing Features of Conditional Random Fields. Computer Science Department. University of Massachusetts

[7] Andrew McCallum, Khashayar Rohanimanesh, and Charles Sutton. Dynamic Conditional Random Fields for Jointly Labeling Multiple Sequences. Department of Computer Science, University of Massachusetts

[8] Chih-Hao Tsai. MMSEG: A Word Identification System for Mandarin Chinese Text Based on Two Variants of the Maximum Matching Algorithm, 1996.

[9] Dinh Dien, Hoang Kiem, Nguyen Van Toan. Vietnamese Word Segmentation.. The sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan, 11/2001. pp. 749 -756

[10] Dong C.Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. Mathematical Programming 45 (1989), pp 503-528

[11] F. Sha and F.Pereia. Shallow parsing with conditional random fields. Proceedings of Human Language Technology, NAACL 2003, 2003

[12] H. M. Wallach. Efficient training of conditional random fields. Master's thesis, University of Edinburgh, 2002

[13] Hammersley, J., & Clifford. P. Markov fields on finite graphs and lattices. Unpublished manuscript ,1971.

[14] Hana Wallach. Efficient Training of Conditional Random Fields. M.Sc. thesis, Division of Informatics, University of Edinburgh, 2002.

[15] J. Lafferty, A. McCallum, and F. Pereia. Conditional ramdom fields: probabilistic models for segmenting and labeling sequence data. In International Conference on Machine Learning, 2001

[16] Mehryar Mohri, AT&T Labs – Research. Weighted Finite-State Transducer Algorithms An Overview.

[17] Robert Malouf. 2002. "A comparison of algorithms for maximum entropy parameter estimation." In Proceedings of the Sixth Conference on Natural Language Learning (CoNLL-2002). Pages 49–55.

[18] Ronald Schoenberg. Optimization with the Quasi-Newton Method, September 5, 2001.

[19] Sunita Sarawagi, William W. Cohen. Semi-Markov Conditional Random Fields for Information Extraction.

[20] Trausti Kristjansson, Aron Cullota, Paul viola, Adrew McCallum. Interactive Information Extraction with Constrained Conditionial Random Fields.

[21] Hoang Cong Duy Vu, Nguyen Le Nguyen, Dinh Dien, Nguyen Quoc Hung. A Vietnamese word segmentation approach using maximum matching algorithms and support vector machines